Diff-Reg: Diffusion Model in Doubly Stochastic Matrix Space for Registration Problem Supplementary Material

Qianliang Wu[1], Haobo Jiang[5], Lei Luo[1], Jun Li[1], Yaqing Ding[4], Jin Xie[2,3]✉, and Jian Yang[1]✉

[1] PCA Lab, Key Lab of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, and Jiangsu Key Lab of Image and Video Understanding for Social Security, School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China
[2] State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China
[3] School of Intelligence Science and Technology, Nanjing University, Suzhou, China
[4] Visual Recognition Group, Faculty of Electrical Engineering, Czech Technical University in Prague, Prague, Czech Republic
[5] National University of Singapore, Singapore
`wuqianliang@njust.edu.cn`

This supplementary material provides additional information on matching matrix space (Sec. 1), more details of network design, experiments, and analysis about 3D registration (Sec. 2), 2D-3D registration (Sec. 3), the derivation of the simplified version loss L_{simple} (Sec. 4) for denoising module g_θ, and limitations (Sec. 5).

1 Revisiting Doubly Stochastic Matrix.

We can represent the point clouds $\mathbf{P}$ and $\mathbf{Q}$ as two graphs, denoted as $\mathcal{G}_1 = \left\{\mathbf{P}, \mathbf{E}^{\mathbf{P}}\right\}$ and $\mathcal{G}_2 = \left\{\mathbf{Q}, \mathbf{E}^{\mathbf{Q}}\right\}$, where $\mathbf{E}^{\mathbf{P}}$ and $\mathbf{E}^{\mathbf{Q}}$ are respective edge sets. The matching matrix between these two graphs is a one-to-one mapping $\mathbf{E} \in \{0,1\}^{N\times M}$. In cases where $N \neq M (e.g., N > M)$, we can introduce $N - M$ dummy points in $\mathbf{Q}$ to make a square matching matrix, also known as a permutation matrix $\mathcal{M} = \left\{A : A1_N = 1_N, A^T 1_N = 1_N, A \geq 0\right\}$. Then, we further employ sinkhorn iterations [5] to convert this non-negative real matrix into a "doubly stochastic" matrix, which has uniform row sum M and column sum N [2]. In paticular, in our method, we use the focal loss function to approximate the predicted full doubly stochastic matrix to a non-full ground truth matching matrix; the matching scores of inlier correspondences are prone to be higher, while those of outlier correspondences are prone to be lower. Consequently, we can safely select the correspondences with the top-k highest matching scores as our inlier correspondences.

✉ Corresponding Authors

These two aspects serve as a significant motivation for our approach:

$$\min_{\mathbf{E}\in\mathcal{M}}\sum_{i=1}^{N}\sum_{j=1}^{M}\mathbf{E}_{ij}*||W^{\mathbf{E}}(\mathbf{p}_i)-\mathbf{q}_j||_2 \le \min_{W\in SE(3)/\mathbb{R}^{3N}/\mathbb{R}^{2N}}\sum_{i=1}^{N}\sum_{j=1}^{M}\hat{\mathbf{E}}^W_{ij}*||W(\mathbf{p}_i)-\mathbf{q}_j||_2$$

where $W^{\mathbf{E}}$ is a warping generated by $\mathbf{E}$ and $\hat{\mathbf{E}}^W$ is a matching matrix computed by the warping W.

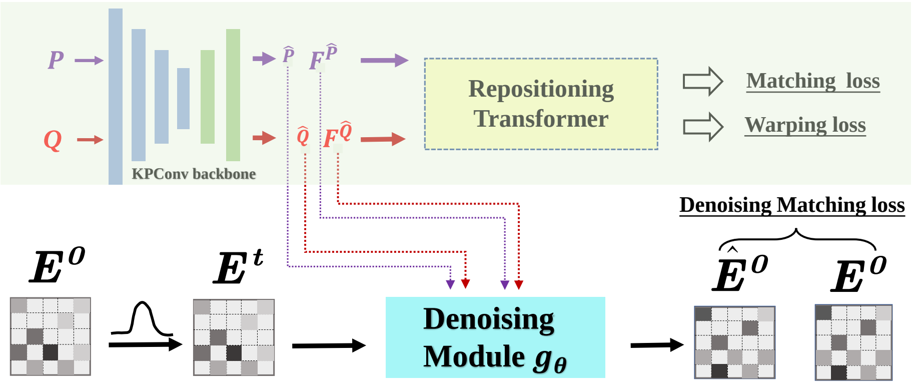


Fig. 1: Overview of our diffusion matching model in the 3D registration task. The top half is the **single-pass prediction head** with the feature backbone, while the bottom is our denoising module.

Algorithm 1 Training Diff-Reg in 3D Registration Task

```
Require: Point clouds $\hat{\mathbf{P}}$, $\hat{\mathbf{Q}} \in \mathbb{R}^3$ and associated point features $\mathbf{F}^{\hat{\mathbf{P}}}, \mathbf{F}^{\hat{\mathbf{Q}}}$.
while not converged do
    Sample $\mathbf{E}^0 \sim q(\mathbf{E}^0)$
    $N \times M \leftarrow \mathbf{E}^0$.shape
    Sample $t \sim$ Uniform$(1, ..., T)$
    $\epsilon \sim \mathcal{N}(0,1)^{N\times M}$
    if Rigid then
        $\mathbf{E}^t \leftarrow \sqrt{\bar{\alpha}_t}\mathbf{E}^0 + \sqrt{1-\bar{\alpha}_t}f_\epsilon(\epsilon)$,   $\tilde{\mathbf{E}}^t = \mathbf{E}^t - \mathrm{Min}(\mathbf{E}^{\mathrm{t}})$
    else
        $\mathbf{E}^t = \sqrt{\bar{\alpha}_t}\mathbf{E}^0 + \sqrt{1-\bar{\alpha}_t}\epsilon_0$,   $\tilde{\mathbf{E}}^t = \mathrm{Sigmoid}(\mathbf{E}^{\mathrm{t}})$
    end if
    $\hat{\mathbf{E}}_0 \leftarrow g_\theta(\tilde{\mathbf{E}}^t, \hat{\mathbf{P}}, \hat{\mathbf{Q}}, \mathbf{F}^{\hat{\mathbf{P}}}, \mathbf{F}^{\hat{\mathbf{Q}}})$
    Optimize $L_t =$ Focal_loss$(\hat{\mathbf{E}}_0, \mathbf{E}^0)$
end while
```

2 3D Registration Task

2.1 Implemantation Details

The framework is trained and tested with PyTorch on one NVIDIA RTX 3090 GPU.

The overview of our network architecture is shown in Fig. 1. We utilize the joint training strategy for the single-pass prediction head and our denoising module. As shown in Fig. 1, the denoising matching loss is for supervising our denoising module, while the matching loss and warping loss are for supervising the single-pass head. During inference, we use the feature backbone to generate the superpoint features $\mathbf{F}^{\hat{\mathbf{P}}}$ and $\mathbf{F}^{\hat{\mathbf{Q}}}$ and treat them as fixed inputs in the reverse denoising sampling process. The training and inference of denoising module g_θ are in Algorithm. 1 and Algorithm. 2.

Algorithm 2 Sampling by Diff-Reg in 3D Registration Task

```
Require: Initial matching matrix $E^T$ from backbone or white noise; Point clouds $\hat{\mathbf{P}}, \hat{\mathbf{Q}} \in \mathbb{R}^3$ and associated point features $\mathbf{F}^{\hat{\mathbf{P}}}, \mathbf{F}^{\hat{\mathbf{Q}}}$.
Ensure: Target matching matrix $\tilde{\mathbf{E}}^0$.
$N \times M \leftarrow \mathbf{E}^T$.shape
for $t = T, ..., 1$ do
    $\mathbf{z}_t \sim N(0,1)^{N \times M}$ if $t > 1$ else $\mathbf{z}_t \leftarrow \mathbf{0}^{N \times M}$
    if Rigid then
        $\tilde{\mathbf{E}}^t = \mathbf{E}^t - \mathrm{Min}(\mathbf{E}^{\mathrm{t}})$
    end if
    $\hat{\mathbf{E}}_0 \leftarrow g_\theta(\hat{\mathbf{E}}^t, \hat{\mathbf{P}}, \hat{\mathbf{Q}}, \mathbf{F}^{\hat{\mathbf{P}}}, \mathbf{F}^{\hat{\mathbf{Q}}})$
    $\epsilon_t \leftarrow \frac{\hat{\mathbf{E}}_0}{\sqrt{1-\bar{\alpha}_t}} - \frac{\sqrt{\bar{\alpha}_t}}{\sqrt{1-\bar{\alpha}_t}} \hat{\mathbf{E}}^t$
    $\sigma_t \leftarrow \sqrt{\frac{(1-\alpha_{t-1})}{1-\alpha_t}} \sqrt{1 - \frac{\alpha_t}{\alpha_{t-1}}}$
    $\hat{\mathbf{E}}^{t-1} \leftarrow \sqrt{\alpha_{t-1}} \hat{\mathbf{E}}_0 + \sqrt{1 - \alpha_{t-1} - \sigma_t^2} \epsilon_t + \sigma_t \mathbf{z}_t$
end for
if Rigid then
    $\tilde{\mathbf{E}}^0 = \tilde{\mathbf{E}}^0 - \mathrm{Min}(\tilde{\mathbf{E}}^0), \quad \tilde{\mathbf{E}}^0 = \mathbf{f}_{\mathrm{sinkhorn}}(\tilde{\mathbf{E}}^0)$
else
    $\tilde{\mathbf{E}}^0 = \mathrm{Sigmoid}(\tilde{\mathbf{E}}^0)$
end if
```

2.2 3DMatch/3DLoMatch Benchmark

Datasets. 3DMatch [26] is an indoor benchmark for 3D matching and registration. Following [8,13,16], we split it to 46/8/8 scenes for training/validation/testing. The overlap ratio between scan pairs in 3DMatch/3DLoMatch is about $> 30\%/10\%-30\%$.

Metrics. Following [8, 10, 16], we utilize three evaluation metrics to evaluate our method and other baselines: (1) Inlier Ratio (IR): The proportion of accurate correspondences in which the distance falls below a threshold (i.e., $0.1m$) based on the ground truth transformation. (2) Feature Matching Recall (FMR): The percentage of matched pairs that have an inlier ratio exceeding a specified threshold (i.e., 5%). (3) Registration Recall (RR): The fraction of successfully registered point cloud pairs with a predicted transformation error below a certain threshold (e.g., RMSE $<$ 0.2).

Results. We compared our method with some state-of-the-art feature matching based methods: FCGF [4], D3Feat [1], Predator [8], Lepard [13], GeoTr [16], and RoITr [24]. The notations **Diff-Reg(steps=1)** and **Diff-Reg(steps=20)** represent our denoising module with one or twenty reverse sampling steps, respectively. As demonstrated in Table 1, our method Diff-Reg(steps=20) achieves the highest registration recall on the 3DMatch benchmark. On the 3DLoMatch benchmark, our method increases by 3% compared to our referenced baseline Lepard [13].

Table 1: Quantitative results on the 3DMatch and 3DLoMatch benchmarks. The best results are highlighted in bold, and the second-best results are underlined.

Method	Reference	3DMatch			3DLoMatch		
		FMR(%)	IR(%)	RR(%)	FMR(%)	IR(%)	RR(%)
FCGF	ICCV2019 [4]	95.20	56.90	88.20	60.90	21.40	45.80
D3Feat	CVPR2020 [1]	95.80	39.00	85.80	69.30	13.20	40.20
Predator	CVPR2021 [8]	96.70	58.00	91.80	78.60	26.70	62.40
Lepard	CVPR2022 [13]	97.95	57.61	93.90	84.22	27.83	70.63
GeoTR	CVPR2022 [16]	**98.1**	72.7	92.3	88.7	44.7	75.4
RoITr	CVPR2023 [24]	98.0	**82.6**	91.9	**89.6**	**54.3**	74.8
PEAL-3D	CVPR2023 [25]	98.5	73.3	94.2	87.6	49.0	**79.0**
Diff-Reg(steps=1)		96.28	30.92	94.8	69.6	9.6	73.3
Diff-Reg(steps=20)		96.28	30.92	**95.0**	69.6	9.6	73.8

2.3 4DMatch/4DLoMatch Benchmark

Metrics. In this section, we give a detailed definition of the two metrics we utilize to evaluate the quality of predicted matches. (1) Inlier ratio (**IR**): This measure denotes the correct fraction in the correspondences prediction $\mathcal{K}_{pred}$:

$$IR = \frac{1}{|\mathcal{K}_{pred}|}\Sigma_{(\hat{\mathbf{p}},\hat{\mathbf{q}})\in\mathcal{K}_{pred}}[||W_{gt}(\hat{\mathbf{p}})-\hat{\mathbf{q}}||_2 < \sigma] \tag{1}$$

where $||\cdot||_2$ is the Euclidean norm, $W_{gt}(\cdot)$ is the ground truth warping function, $[\cdot]$ is the Inverse bracket, and $\sigma = 0.04m$. (2) Non-rigid Feature Matching Recall (**NFMR**): This measure is to compute the fraction of the ground truth correspondences $(u, v) \in \mathcal{K}_{gt}$ that can be successfully recovered from the predicted correspondences $\mathcal{K}_{pred}$. First, we construct the predicted correspondences $\mathcal{A} = \{\hat{\mathbf{p}}|(\hat{\mathbf{p}}, \hat{\mathbf{q}}) \in \mathcal{K}_{pred}\}$ and the associated sparse 3D flow fields $\mathcal{F} = \{\hat{\mathbf{q}} - \hat{\mathbf{p}}|(\hat{\mathbf{p}}, \hat{\mathbf{q}}) \in \mathcal{K}_{pred}\}$. Then, for any source point u in $\mathcal{K}_{gt}$, we can recover the flow field for u by inverse distance interpolation:

$$\mathit{\Gamma}(u, \mathcal{A}, \mathcal{F}) = \mathit{\Sigma}_{\mathcal{A}_i \in \mathcal{N}(u,\mathcal{A})} \frac{\mathcal{F}_i ||u - \mathcal{A}_i||_2^{-1}}{\mathit{\Sigma}_{\mathcal{A}_i \in \mathcal{N}(u,\mathcal{A})} ||u - \mathcal{A}_i||_2^{-1}} \tag{2}$$

where $\mathcal{N}(\cdot, \cdot)$ is k-nearest neighbors search with k = 3. After that, we define the $NFMR$ to measure the fraction of ground truth matches that we discovered from $\mathcal{K}_{pred}$:

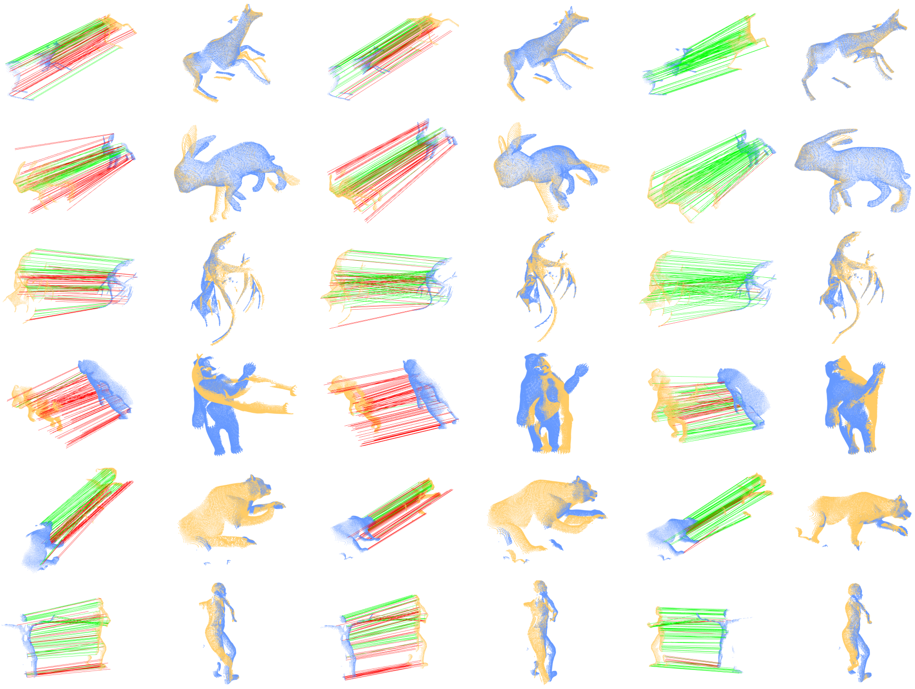
$$NFMR = \frac{1}{|\mathcal{K}_{gt}|} [||\mathit{\Gamma}(u, \mathcal{A}, \mathcal{F}) - (v - u)||_2 < \sigma] \tag{3}$$

Results. The definition of NFMR indicates that a higher NFMR value reflects a higher quality of $\mathcal{K}_{pred}$. We provide additional visualizations of the blended motion based on the predicted correspondences in Fig. 2. The top three lines are from 4DMatch, while the bottom three lines are from the 4DLoMatch dataset. The results in the bottom three lines demonstrate that our denoising module can effectively handle scenes with "large deformations + low overlapping," whereas the top three lines show that our method excels in registering asymmetric objects.

2.4 Ablation Study and Discussion of 3D Registration Task.

Capability of Escaping from Local Minima. The DDPM framework is specifically designed to remove noise from perturbed samples. Typically, we begin with a baseline method that provides an initial solution, which can then be further refined to achieve better performance. To demonstrate that our denoising network has indeed learned the posterior distribution for the denoising module, we conducted an ablative experiment. In this experiment, we initiated reverse sampling from the solution obtained by the single-pass prediction head or from Gaussian white noise. In Table 2, the $E^T_{Backbone}$ results demonstrate that our denoising network can overcome local minima produced by the single-pass prediction head.

Reverse Sampling Steps. Our approach considers the denoising module as an optimizer that searches for the optimal matching matrix. We argue that increasing the number of search iterations may lead to better solutions. To validate this hypothesis, we experimented to investigate the impact of iterative searching steps on the performance. We ran the reverse sampling step from 1 to 20 iterations. As shown in Table. 3, the registration recall of our diffusion matching model increases as the number of sampling steps grows. The results prove that the reverse sampling process can reach a better solution by increasing the number of search steps.



(a) GeoTR [16] **(b)** RoITr [24] **(c)** Diff-Reg(steps=20)

Fig. 2: More qualitative results of non-rigid registration in the 4DMatch/4DLoMatch benchmark. The blue and yellow colors denote the source and target point cloud, respectively. The green and red lines indicate whether the threshold accepts the predicted deformable flow from the source points. The deformable registration is built by GraphSCNet [17]. Zoom in for details.

Table 2: $E^T_{Backbone}$ denote the starting point where E^T is generated by the single-pass prediction head. $E^T_{Gaussian}$ denote the starting point where E^T is sampling from the Gaussian white noise $N(0,1)^{N\times M}$. $z_t = 0$ denotes deterministic sampling, while $z_t \neq 0$ denotes the random sampling.

	$z_t \neq 0$						$z_t = 0$					
	3DMatch			3DLoMatch			3DMatch			3DLoMatch		
	FMR	IR	RR	FMR	IR	RR	FMR	IR	RR	FMR	IR	RR
$E^T_{Backbone}$	96.12	31.07	94.8	69.79	9.71	73.8	96.30	30.92	94.9	69.71	9.51	73.4
$E^T_{Gaussian}$	96.14	31.07	94.6	69.73	9.72	73.8	96.28	30.93	94.8	69.62	9.56	73.3

	$z_t \neq 0$				$z_t = 0$			
	4DMatch		4DLoMatch		4DMatch		4DLoMatch	
	NFMR	IR	NFMR	IR	NFMR	IR	NFMR	IR
$E^T_{Backbone}$	88.38	86.38	75.94	67.64	88.34	86.36	76.22	67.82
$E^T_{Gaussian}$	88.40	86.40	76.09	67.73	88.72	86.72	76.48	68.16

Table 3: The ablation study of the iterative steps of reverse sampling.

Sampling Steps	RR		NFMR/IR	
	3DMatch	3DLoMatch	4DMatch	4DLoMatch
1	94.8	73.3	85.34/83.93	73.11/65.26
10	94.8	73.4	87.99/86.07	75.46/67.15
20	95.0	73.8	88.39/86.64	76.16/67.82

Table 4: Comparison of time cost in the reverse sampling step. The best results are highlighted in bold.

Method	3DMatch		3DLoMatch	
	RR	Time(sec.)	RR	Time(sec.)
GeoTR [16]	92.0	0.296	74.0	0.284
GeoTR. + PEAL [25] 1-step	93.7	0.663	77.8	0.642
KPConv. + Diff-Reg 1-step	**94.8**	**0.048**	73.2	**0.052**
GeoTR. + PEAL [25] 5-step	94.0	2.131	78.5	2.074
KPConv. + Diff-Reg 5-step	**94.8**	**0.182**	73.7	**0.194**

The Lightweight Design of Denoising Module. Due to the lightweight design, our denoising network showcases a notable speed enhancement compared to other concurrent diffusion-based methods for rigid point cloud registration. This efficiency improvement allows our approach to conduct more denoising iterations. We have included a detailed list displaying the time costs (refer to Table 4) associated with the reverse denoising steps and comparisons to recent studies. Our method achieves competitive results on the 3DMatch benchmark while maintaining fast processing speeds.

Robustness to Noise and Outliers. To assess the robustness of our method, we conducted noise resilience experiments on the 4DLoMatch dataset by introducing various levels of Gaussian noise [0.002, 0.005, 0.05, 0.1]. The corresponding IR/NFMR results [67.8%/76.2%, 67.7%/76.1%, 66.9%/75.4%, 63.4%/70.4%] demonstrate the resilience of our approach to noise. Additionally, we analyzed the performance of our method across different overlap ratios. Points situated in the non-overlapping region can be considered as outliers. The evaluation results illustrated in Fig.3 reveal that our methods display consistent inlier ratio (IR) and registration precision (NFMR) on both the 4DMatch and 4DLoMatch datasets across varying outlier ratios. By leveraging our diffusion-enhanced matching matrix, many salient super points can be effectively identified for robust registration. This is further supported by the high NFMR scores achieved on the 4DLoMatch dataset, even with high outlier ratios (i.e., low overlapping ratio).

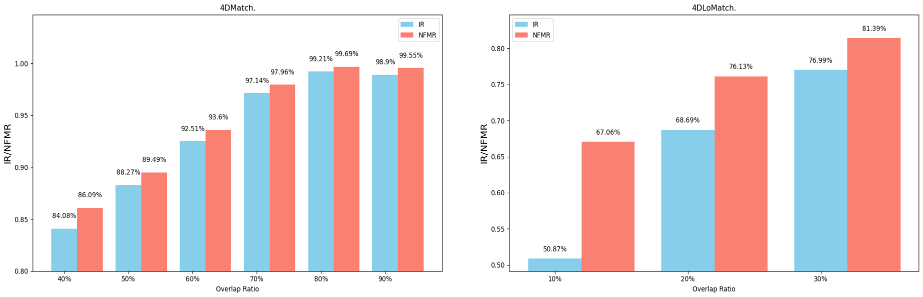


Fig. 3: Stability in the presence of outliers. **Left**: 4DMatch. **Right**: 4DLoMatch. Zoom in for details.

Trade-off between Quality and Efficiency. In indoor and outdoor scenes, the number of sensor scan points can reach several tens of thousands. To handle this large amount of data, we employ a down-sampling strategy to obtain coarse-level super points and exploit the point-to-node grouping [23] to maintain dense-to-coarse mapping relations. For the 3D registration task, we select the second to last layer of the KPConv backbone [18] as our super points. The number of coarse points in this layer can be as high as more than 1k, which is close to the GPU memory limit of the RTX3090 during the training stage. In the 2D-3D registration task, we take the coarsest points/pixels in the last down-sampling layer of KPConv and ResNet [7], where the coarsest super points or superpixels numbers are up to several hundred. We choose these two different coarse levels for a trade-off between quality and efficiency.

Algorithm 3 Training Diff-Reg in 2D-3D Registration Task

```
Require: Coarse level image and points $\hat{\mathbf{X}} \in \mathbb{R}^{\hat{H}\times\hat{W}\times 3}$, $\hat{\mathbf{Q}} \in \mathbb{R}^3$ and associated features $\mathbf{F}^{\hat{\mathbf{X}}}, \mathbf{F}^{\hat{\mathbf{X}}}_{dino}, \mathbf{F}^{\hat{\mathbf{Q}}}$.
while not converged do
    Sample $\mathbf{E}^0 \sim q(\mathbf{E}^0)$
    $N \times M \leftarrow \mathbf{E}^0$.shape
    Sample $t \sim \text{Uniform}(1, ..., T)$
    $\epsilon \sim \mathcal{N}(0,1)^{N\times M}$
    $\mathbf{E}^t \leftarrow \sqrt{\bar{\alpha}_t}\mathbf{E}^0 + \sqrt{1-\bar{\alpha}_t} f_\epsilon(\epsilon), \quad \tilde{\mathbf{E}}^t = \mathbf{E}^t - \text{Min}(\mathbf{E}^t)$
    $\hat{\mathbf{E}}_0 \leftarrow g_\theta(\tilde{\mathbf{E}}^t, \hat{\mathbf{X}}, \hat{\mathbf{Q}}, \mathbf{F}^{\hat{\mathbf{X}}}, \mathbf{F}^{\hat{\mathbf{X}}}_{dino}, \mathbf{F}^{\hat{\mathbf{Q}}})$
    Optimize $L_t = \text{Focal\_loss}(\hat{\mathbf{E}}_0, \mathbf{E}^0)$
end while
```

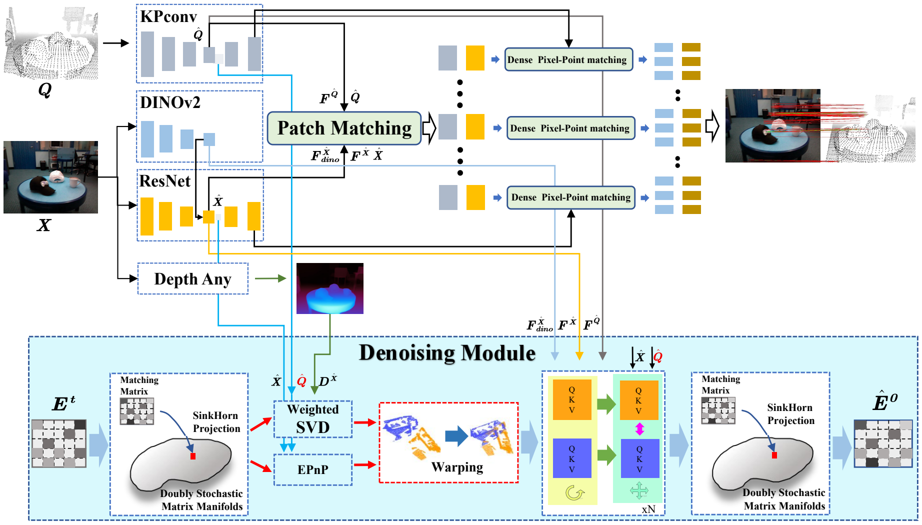

KPconv
Q
DINOv2
ResNet
X
Depth Any
Patch Matching
$F^{\hat{Q}}$ $\hat{Q}$
$F^{\hat{X}}_{dino}$ $F^{\hat{X}}$ $\hat{X}$
Dense Pixel-Point matching
Dense Pixel-Point matching
Dense Pixel-Point matching
$F^{\hat{X}}_{dino}$ $F^{\hat{X}}$ $F^{\hat{Q}}$
Denoising Module
E^t
Matching Matrix
SinkHorn Projection
Doubly Stochastic Matrix Manifolds
$\hat{X}$ $\hat{Q}$ $D^{\hat{X}}$
Weighted SVD
EPnP
Warping
Q K V
xN
$\hat{E}^0$

Fig. 4: Overview of Our diffusion matching model framework for 2D-3D Registration Task: The top half is the **single-pass prediction head** with the backbones, while the bottom is our denoising module. The transformer in our denoising module utilizes the $\mathbf{F}^{\hat{\mathbf{X}}}_{dino}$, $\mathbf{F}^{\hat{\mathbf{X}}}$, $\mathbf{F}^{\hat{\mathbf{Q}}}$, $\hat{\mathbf{X}}$, $\hat{\mathbf{Q}}$, and $\mathbf{D}^{\hat{\mathbf{X}}}$ as inputs, while "patch matching" module in the single-pass head takes $\mathbf{F}^{\hat{\mathbf{X}}}_{dino}$, $\mathbf{F}^{\hat{\mathbf{X}}}$, $\mathbf{F}^{\hat{\mathbf{Q}}}$, $\hat{\mathbf{X}}$, and $\hat{\mathbf{Q}}$ as inputs. Please zoom in for details.

Algorithm 4 Sampling by Diff-Reg in 2D-3D Registration Task

Require: Initial matching matrix $\mathbf{E}^T$ from backbone or white noise; Coarse level image and points $\hat{\mathbf{X}} \in \mathbb{R}^{\hat{H}\times\hat{W}\times 3}$, $\hat{\mathbf{Q}} \in \mathbb{R}^3$ and associated features $\mathbf{F}^{\hat{\mathbf{X}}}, \mathbf{F}^{\hat{\mathbf{X}}}_{dino}, \mathbf{F}^{\hat{\mathbf{Q}}}$.

Ensure: Target matching matrix $\tilde{\mathbf{E}}^0$).

```
$N \times M \leftarrow \mathbf{E}^T$.shape
for $t = T, ..., 1$ do
    $\mathbf{z}_t \sim \mathcal{N}(0,1)^{N\times M}$ if $t > 1$ else $\mathbf{z}_t \leftarrow \mathbf{0}^{N\times M}$
    $\hat{\mathbf{E}}_0 \leftarrow g_\theta(\tilde{\mathbf{E}}^t, \hat{\mathbf{X}}, \hat{\mathbf{Q}}, \mathbf{F}^{\hat{\mathbf{X}}}, \mathbf{F}^{\hat{\mathbf{X}}}_{dino}, \mathbf{F}^{\hat{\mathbf{Q}}})$
    $\epsilon_t \leftarrow \frac{\hat{\mathbf{E}}_0}{\sqrt{1-\bar{\alpha}_t}} - \frac{\sqrt{\bar{\alpha}_t}}{\sqrt{1-\bar{\alpha}_t}}\hat{\mathbf{E}}^t$
    $\sigma_t \leftarrow \sqrt{\frac{(1-\alpha_{t-1})}{1-\alpha_t}}\sqrt{1-\frac{\alpha_t}{\alpha_{t-1}}}$
    $\hat{\mathbf{E}}^{t-1} \leftarrow \sqrt{\alpha_{t-1}}\hat{\mathbf{E}}_0 + \sqrt{1-\alpha_{t-1}-\sigma_t^2}\epsilon_t + \sigma_t\mathbf{z}_t$
end for
$\tilde{\mathbf{E}}^0 = \mathbf{f}_{\text{sinkhorn}}(\tilde{\mathbf{E}}^0)$
```

3 2D-3D Registration Task

3.1 Implemantation Details

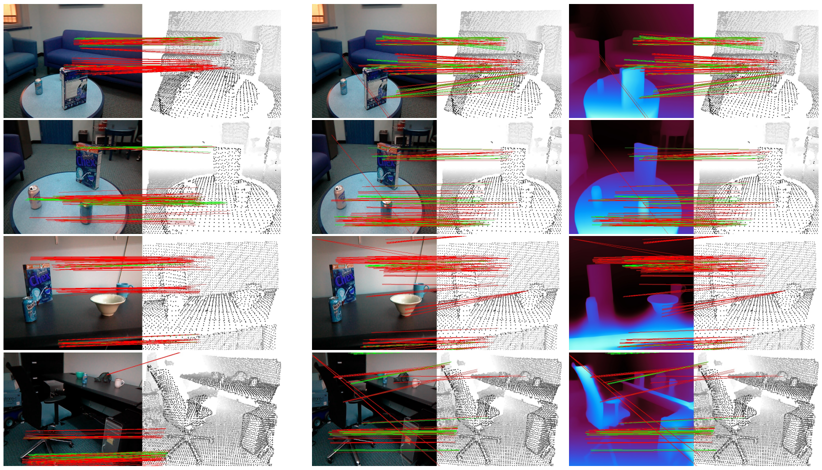
The overview of our framework for the 2D-3D registration task is illustrated in Fig.4. We use **Diff-Reg(dino)** to denote the single-pass prediction head trained solely. **Diff-Reg(dino/backbone)** refers to the single-pass prediction head that is jointly trained with our denoising module g_θ. **Diff-Reg(dino/steps=*)** denotes that the "Patch Matching" module in the framework is replaced with our denoising module (with * steps of reverse sampling). The superpixel features $\mathbf{F}^{\hat{\mathbf{X}}}$, $\mathbf{F}^{\hat{\mathbf{X}}}_{dino}$, and superpoint features $\mathbf{F}^{\hat{\mathbf{Q}}}$ are treated as fixed inputs of denoising module g_θ in each reverse sampling step. For all variants of Diff-Reg(*), we exclude the three scales "6 × 8, 12 × 16, 24 × 32" (refer to section 4.1 in [11]) after the coarsest level of ResNet and preserve only "24 × 32" resolution. Then the visual features from DINOv2 [15] are combined with the coarsest level features that are upsampled to the finest level in ResNet. The training and reverse sampling process are listed in Algorithm.3 and Algorithm.4

3.2 Metrics

In this section, following [11], we give a detailed definition of three evaluation protocols: (1) Inlier Ratio (IR), the ratio of pixel-point matches whose 3D distance is under a certain threshold (i.e., 5cm). (2) Feature Matching Recall (FMR), the ratio of image-to-point-cloud pairs whose inlier ratio is above a certain threshold (i.e., 10%). (3) Registration Recall (RR), the ratio of image-to-point-cloud pairs whose RMSE is under a certain threshold (i.e., 10cm).

3.3 Results

Following [11], we compare our diffusion matching model with FCGF [4], P2-Net [20], Predator [8], 2D3D-MATR [11], and FreeReg [21]. As demonstrated in Table 5, our method outperforms 2D3D-MATR [11] and FreeReg [21] significantly. The feature matching recall of Diff-Reg(dino/backbone) achieving the best performance proves that our denoising module's training brings implicit data augmentation. The best recall performance of our Diff-Reg(dino/steps=1) indicates that our denoising module seeks salient combinational correspondences crucial for successful registration. We provide more visualizations in Fig. 5 to illustrate our method's effectiveness.



(a) Diff-Reg(dino) **(b)** Diff-Reg(dino/steps=10)

Fig. 5: More qualitative results on the RGB-D Scenes V2 benchmark [9]. The green/red color indicates whether the matching score is accepted based on a threshold value. Zoom in for details.

Table 5: Evaluation results on RGB-D Scenes V2 [11]. The best results are highlighted in bold, and the second-best results are underlined.

Model	Scene-11	Scene-12	Scene-13	Scene-14	Mean
Mean depth (m)	1.74	1.66	1.18	1.39	1.49
Inlier Ratio↑					
FCGF-2D-3D [4]	6.8	8.5	11.8	5.4	8.1
P2-Net [4]	9.7	12.8	17.0	9.3	12.2
Predator-2D-3D [8]	17.7	19.4	17.2	8.4	15.7
2D3D-MATR [11]	32.8	34.4	39.2	23.3	32.4
FreeReg [21]	36.6	34.5	34.2	18.2	30.9
Diff-Reg(dino)	38.6	37.4	**45.4**	31.6	38.3
Diff-Reg(dino/backbone)	44.9	**49.5**	38.3	**33.1**	**41.4**
Diff-Reg(dino/steps=1)	**47.5**	48.9	32.8	22.4	37.9
Diff-Reg(dino/steps=10)	47.2	48.7	32.9	22.4	37.8
Feature Matching Recall↑					
FCGF-2D-3D [4]	11.10	30.40	51.50	15.50	27.10
P2-Net [4]	48.60	65.70	82.50	41.6	59.60
Predator-2D-3D [8]	86.10	89.20	63.90	24.30	65.90
2D3D-MATR [11]	98.60	98.00	88.70	77.90	90.80
FreeReg [21]	91.90	93.40	**93.10**	49.60	82.00
Diff-Reg(dino)	**100.**	**100.**	89.70	81.9	92.9
Diff-Reg(dino/backbone)	**100.**	**100.**	92.8	**91.2**	**96.0**
Diff-Reg(dino/steps=1)	**100.**	**100.**	88.7	76.5	91.3
Diff-Reg(dino/steps=10)	**100.**	**100.**	88.7	77.0	91.4
Registration Recall↑					
FCGF-2D-3D [4]	26.4	41.2	37.1	16.8	30.4
P2-Net [4]	40.3	40.2	41.2	31.9	38.4
Predator-2D-3D [8]	44.4	41.2	21.6	13.7	30.2
2D3D-MATR [11]	63.9	53.9	58.8	49.1	56.4
FreeReg+Kabsch [21]	38.7	51.6	30.7	15.5	34.1
FreeReg+PnP [21]	74.2	72.5	54.5	27.9	57.3
Diff-Reg(dino)	87.5	86.3	63.9	60.6	74.6
Diff-Reg(dino/backbone)	79.2	86.3	75.3	**71.2**	78.0
Diff-Reg(dino/steps=1)	**98.6**	**100.**	87.6	66.8	**88.3**
Diff-Reg(dino/steps=10)	**98.6**	96.1	83.5	63.7	85.5
Diff-Reg(dino/backbone$_{epnp}$)	95.8	96.1	**88.7**	69.0	87.4

3.4 Denoising Module $g_{\boldsymbol{\theta}}$

In this section, we elucidate the functionality of our denoising module. By leveraging the recent state-of-the-art depth estimation model DepthAnything [22], we consider this non-metric depth as a constant input for each image. **Diff-Reg(dino/backbone**$_{epnp}$**)** (in Table.5) utilizes the EPnP solver [12] to determine the transformation. Its high recall provides evidence that the implicit data augmentation from training our denoising module indeed enhances the image backbone and the point cloud backbone. However, the EPnP solver's vulnerability to outliers during inference hinders its direct application in the reverse sampling process. Integrating the more robust PnP solver [3] into our denoising module could potentially remove the need for the depth estimation model. This enhancement is an aspect we aim to explore in future research.

During inference, each reverse sampling step takes $\mathbf{E}^{t-1}$ from the previous step to provide initial correspondences. We then use either weighted SVD or the EPnP solver to calculate the transformation $\boldsymbol{\Gamma}^{t-1}$ that maps the point cloud $\hat{\mathbf{Q}}$ into the image plane space. Subsequently, we apply $\boldsymbol{\Gamma}^{t-1}$ to $\hat{\mathbf{Q}}$ to get warped $\hat{\mathbf{Q}}^{\boldsymbol{\Gamma}^{t-1}}$. Next, we use $\hat{\mathbf{Q}}^{\boldsymbol{\Gamma}^{t-1}}$, $\hat{\mathbf{X}}$, $\mathbf{F}^{\hat{\mathbf{X}}}_{dino}$, $\mathbf{F}^{\hat{\mathbf{X}}}$, and $\mathbf{F}^{\hat{\mathbf{Q}}}$ as inputs of the transformer in g_θ to compute the updated image features and point features.

4 Derivation of $\mathbf{L}_{\boldsymbol{simple}}$

For latent variable $\mathbf{E}^{1:T}$, the Evidence Lower Bound (ELBO) for $\mathbf{E}^0$ with distribution q formulates as:

$$
\begin{aligned}
\log(p_\theta(\mathbf{E}^0)) \geq \mathbb{E}_{q(\mathbf{E}^{1:T}|\mathbf{E}^0)}\left[\log\left(\frac{p_\theta(\mathbf{E}^{0:T})}{q(\mathbf{E}^{1:T}|\mathbf{E}^0)}\right)\right] &= L_{vb}(\mathbf{E}^0) \\
= \mathbb{E}_{q(\mathbf{E}^1|\mathbf{E}^0)}\left[\log(p_\theta(\mathbf{E}^0|\mathbf{E}^1))\right] &+ \mathbb{E}_{q(\mathbf{E}^T|\mathbf{E}^0)}\left[\log(\frac{p_\theta(\mathbf{E}^T)}{q(\mathbf{E}^T|\mathbf{E}^0)})\right] \\
+ \sum_{t=2}^{T} \mathbb{E}_{q(\mathbf{E}^t,\mathbf{E}^{t-1}|\mathbf{E}^0)}&\left[\frac{p_\theta(\mathbf{E}^{t-1}|\mathbf{E}^t)}{q(\mathbf{E}^{t-1}|\mathbf{E}^t,\mathbf{E}^0)}\right] \\
= C_\theta(\mathbf{E}^T,\mathbf{E}^1,\mathbf{E}^0) - \sum_{t=2}^{T} &\underbrace{\mathbb{E}_{q(\mathbf{E}^t|\mathbf{E}^0)}\left[D_{KL}\left(q(\mathbf{E}^{t-1}|\mathbf{E}^t,\mathbf{E}^0)||p_\theta(\mathbf{E}^{t-1}|\mathbf{E}^t)\right)\right]}_{\text{denoising matching term}}.
\end{aligned}
$$

The posterior distribution for $q(\mathbf{E}^{t-1}|\mathbf{E}^t,\mathbf{E}^0)$ is defined as:

$$
\begin{aligned}
q(\mathbf{E}^{t-1}|\mathbf{E}^t,\mathbf{E}^0) &= \frac{q(\mathbf{E}^t|\mathbf{E}^{t-1},\mathbf{E}^0)q(\mathbf{E}^{t-1}|\mathbf{E}^0)}{q(\mathbf{E}^t|\mathbf{E}^0)} \\
\propto N(\mathbf{E}^{t-1}; &\underbrace{\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})\mathbf{E}^t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)\mathbf{E}^0}{1-\bar{\alpha}_t}}_{\mu_q(\mathbf{E}^t,\mathbf{E}^0)}, \underbrace{\frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{I}}_{\varSigma_q(t)}).
\end{aligned}
\tag{4}
$$

We propose a lightweight denoising network $g_{\theta}(E^t)$ seek to predict E^0 from any noisy one E^t, the optimization object is simplified to:

$$\begin{aligned}
&\arg\min_{\theta}\, D_{KL}\left(q(\mathbf{E}^{t-1}|\mathbf{E}^{t},\mathbf{E}^{0})||p_{\theta}(\mathbf{E}^{t-1}|\mathbf{E}^{t})\right)\\
&=\arg\min_{\theta}\, D_{KL}\left(\mathcal{N}(\mathbf{E}^{t-1},\mu_q,\mathit{\Sigma}_q(t))||\mathcal{N}(\mathbf{E}^{t-1},\mu_{\theta},\mathit{\Sigma}_q(t))\right)\\
&=\arg\min_{\theta}\,\frac{1}{2\sigma_q^2(t)}\frac{\bar{\alpha}(1-\alpha_t)^2}{(1-\bar{\alpha})^2}\left[||g_{\theta}(\mathbf{E}^{t})-\mathbf{E}^{0}||_2^2\right]\\
&\equiv-\arg\min_{\theta}\,\mathbb{E}_{q(\mathbf{E}^{0})}\left[\mathbb{E}_{q(\mathbf{E}^{t}|\mathbf{E}^{0})}logp_{\theta}(\mathbf{E}^{0}|\mathbf{E}^{t})\right]
\end{aligned}\tag{5}$$

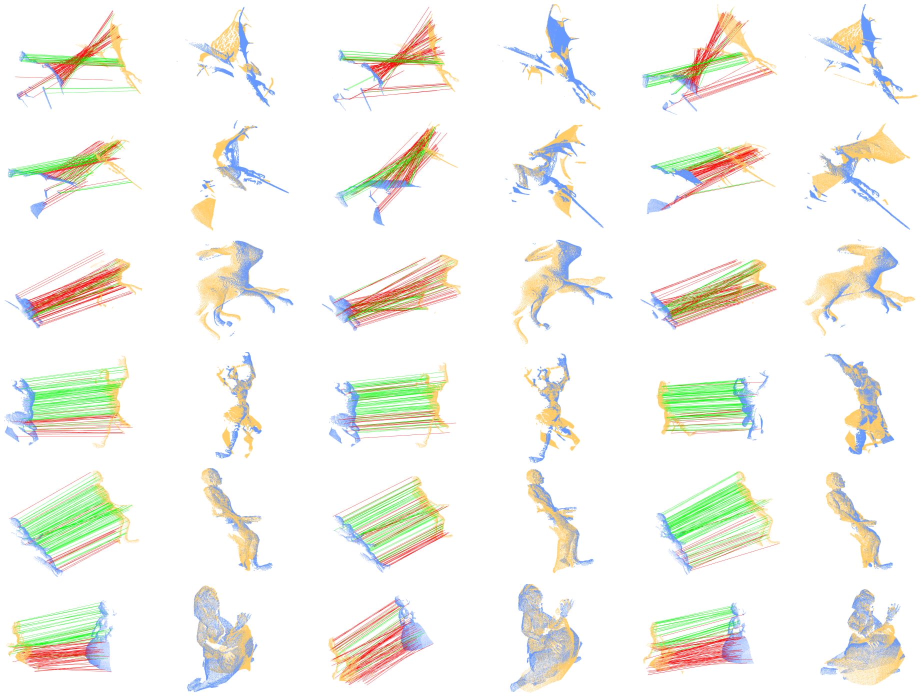
5 Challenges and Limitations

As illustrated in Fig. 6, deformable registration encounters various challenges, including large motion, low overlap, sparse overlapping regions, and local non-rigid motion (i.e., drastic non-isometric deformation, such as the dress worn by the dancer in the last line of Fig. 6). Although our diffusion matching model is intended to address both rigid and deformable registration tasks, it may have inherent limitations. The transformer in our denoising module and the associated KPConv and ResNet backbone follow a relatively generic design. To overcome these challenges, we acknowledge the importance of integrating more robust feature embedding techniques [16,24] into our feature backbone for enhancements. Furthermore, incorporating physics priors into our denoising module could prove beneficial in addressing specific issues such as non-isometrical deformation. More failed cases on the 4DMatch and 4DLoMatch can be seen in Fig.7 and Fig.8. These failed cases reveal that the non-rigid registration task is a very tough problem that deserves our further effort to improve.

This paper did not explore the difficult case of large variabilities in shapes (e.g., the human case). However, we conducted an experiment in which we directly generalized our method to non-isometric human cases (point cloud pair borrowed from [14]). As shown in Fig. 9, our method can potentially achieve reliable correspondence estimation on human cases with substantial non-isometric deformations, thanks to our effective correspondence denoising mechanism. We plan to extend our method to the human case in our future research.

The other challenging scenario involves an extremely low overlapping region, relying on discriminative local point features. The suboptimal performance of our method on the 3DLoMatch benchmark (10% to 30% overlap ratio) may be due to the absence of integrated geometric embeddings, as discussed in [6,16,19,24], within the feature backbone or transformer of our denoising module. We plan to complete this improvement in our future work.

Although we achieved successful 2D-3D registration on the indoor dataset RGB-D Scenes V2 [11], 3D registration heavily relies on the geometric features of point clouds. The DepthAnything model [22] may encounter challenges in accurately estimating depth at greater distances. Furthermore, the model's reliance on multiple outdoor datasets for pre-training limits its direct applicability in outdoor registration tasks due to potential leaks in ground truth depth labels. Our upcoming research will address the issue by replacing the weighted SVD with the EPnP solver. This allows us to conduct registration without relying on the DepthAnything model [22], using only point cloud and image pairs. Please pay attention to our subsequent related research work.



(a) GeoTR [16] **(b)** RoITr [24] **(c)** Diff-Reg(steps=20)

Fig. 6: Failure cases of our diffusion matching model on 4DMatch/4DLoMatch benchmark. The blue and yellow colors denote the source and target point cloud, respectively. The green and red lines indicate whether the threshold accepts the predicted deformable flow from the source points. The deformable registration is built by GraphSCNet [17]. Zoom in for details.

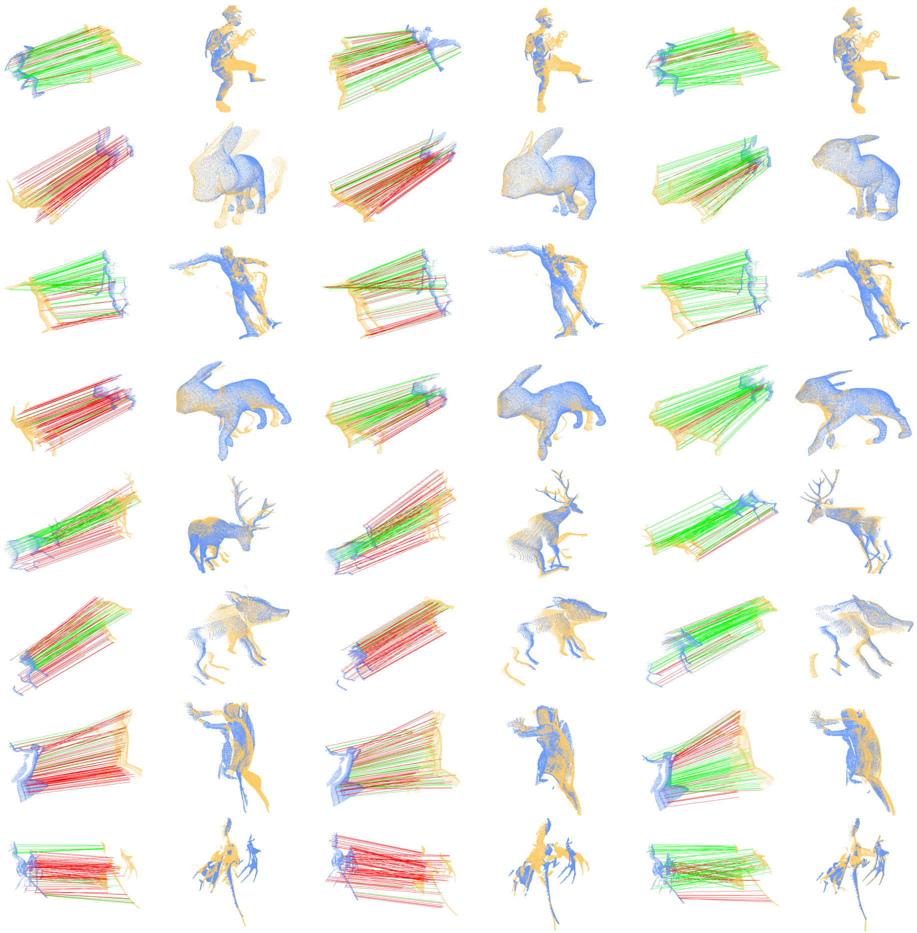


Fig. 7: More failure cases on the **4DMatch** benchmark. The blue and yellow colors denote the source and target point cloud, respectively. The green and red lines indicate whether the threshold accepts the predicted deformable flow from the source points. The deformable registration is built by GraphSCNet [17]. Zoom in for details.

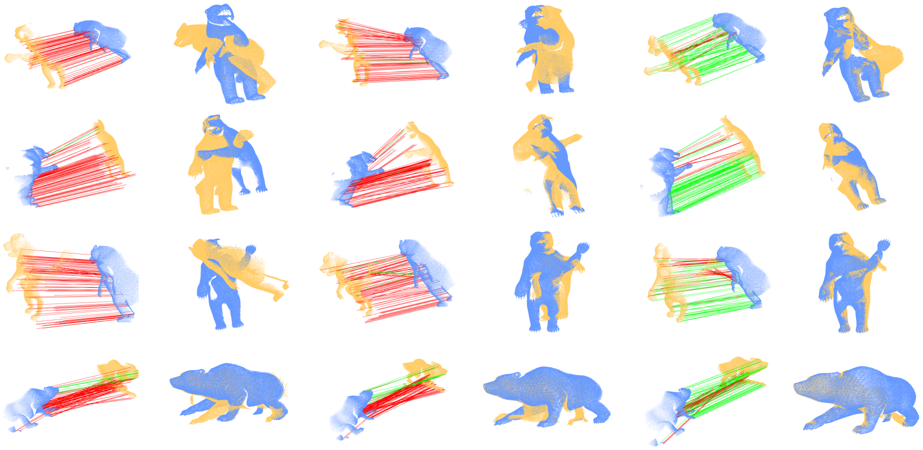


Fig. 8: More failure cases on the **4DLoMatch** benchmark. The blue and yellow colors denote the source and target point cloud, respectively. The green and red lines indicate whether the threshold accepts the predicted deformable flow from the source points. The deformable registration is built by GraphSCNet [17]. Zoom in for details.

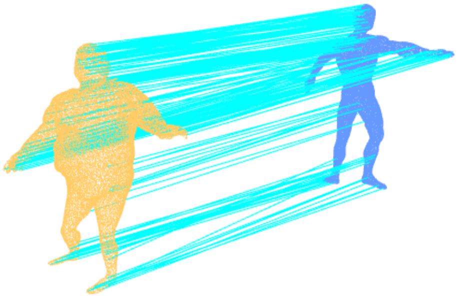


Fig. 9: The correspondences estimated by our model trained on the 4DMatch dataset.

References

1. Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., Tai, C.L.: D3feat: Joint learning of dense detection and description of 3d local features. In: CVPR (2020)
2. Caron, R.M., Li, X., Mikusiński, P., Sherwood, H., Taylor, M.D.: Nonsquare "doubly stochastic" matrices. Lecture Notes-Monograph Series **28**, 65–75 (1996), http://www.jstor.org/stable/4355884
3. Chen, H., Wang, P., Wang, F., Tian, W., Xiong, L., Li, H.: Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2781–2790 (2022)
4. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2019)
5. Cuturi, M.: Sinkhorn distances: Lightspeed computation of optimal transport. Advances in neural information processing systems (2013)
6. Deng, H., Birdal, T., Ilic, S.: Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In: Proceedings of the European Conference on Computer Vision (ECCV) (2018)
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016)
8. Huang, S., Gojcic, Z., Usvyatsov, M., Wieser, A., Schindler, K.: Predator: Registration of 3d point clouds with low overlap. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition (2021)
9. Lai, K., Bo, L., Fox, D.: Unsupervised feature learning for 3d scene labeling. In: IEEE International Conference on Robotics and Automation (ICRA) (2014)
10. Lee, J., Cho, M., Lee, K.M.: Hyper-graph matching via reweighted random walks. In: CVPR (2011)
11. Li, M., Qin, Z., Gao, Z., Yi, R., Zhu, C., Guo, Y., Xu, K.: 2d3d-matr: 2d-3d matching transformer for detection-free registration between images and point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2023)
12. Li, S., Xu, C., Xie, M.: A robust o (n) solution to the perspective-n-point problem. IEEE transactions on pattern analysis and machine intelligence (2012)
13. Li, Y., Harada, T.: Lepard: Learning partial point cloud matching in rigid and deformable scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
14. Li, Y., Harada, T.: Non-rigid point cloud registration with neural deformation pyramid. Advances in Neural Information Processing Systems (2022)
15. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
16. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K.: Geometric transformer for fast and robust point cloud registration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)
17. Qin, Z., Yu, H., Wang, C., Peng, Y., Xu, K.: Deep graph-based spatial consistency for robust non-rigid point cloud registration. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
18. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. In: Proceedings of the IEEE/CVF international conference on computer vision (2019)

19. Deng, H., Birdal, T., Ilic, S.: Ppfnet: Global context aware local features for robust 3d point matching. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2018)
20. Wang, B., Chen, C., Cui, Z., Qin, J., Lu, C.X., Yu, Z., Zhao, P., Dong, Z., Zhu, F., Trigoni, N., Markham, A.: P2-net: Joint description and detection of local features for pixel and point matching. 2021 IEEE/CVF International Conference on Computer Vision (ICCV) (2021)
21. Wang, H., Liu, Y., Wang, B., Sun, Y., Dong, Z., Wang, W., Yang, B.: Freereg: Image-to-point cloud registration leveraging pretrained diffusion models and monocular depth estimators. ArXiv (2023)
22. Yang, L., Kang, B., Huang, Z., Xu, X., Feng, J., Zhao, H.: Depth anything: Unleashing the power of large-scale unlabeled data. arXiv preprint arXiv:2401.10891 (2024)
23. Yu, H., Li, F., Saleh, M., Busam, B., Ilic, S.: Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. Advances in Neural Information Processing Systems (2021)
24. Yu, H., Qin, Z., Hou, J., Saleh, M., Li, D., Busam, B., Ilic, S.: Rotation-invariant transformer for point cloud matching. In: CVPR (2023)
25. Yu, J., Ren, L., Zhang, Y., Zhou, W., Lin, L., Dai, G.: Peal: Prior-embedded explicit attention learning for low-overlap point cloud registration. In: CVPR (2023)
26. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1802–1811 (2017)