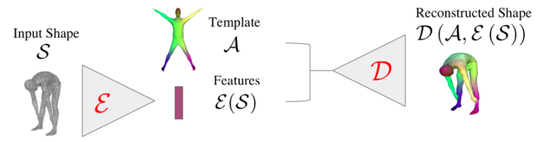
Ordered Diffusion for 3D Human Registration

Mattia Masiero, Ilya A. Petrov, Daniel Cremers, Gerard Pons-Moll, Riccardo Marin

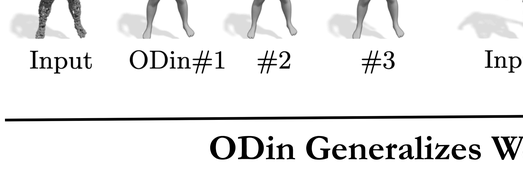
Eberhard Karls Universität Tübingen · Tübingen AI Center · Munich Center for Machine Learning · Max Planck Institute for Informatics

3D non-rigid registration is generally tackled as a reconstruction task



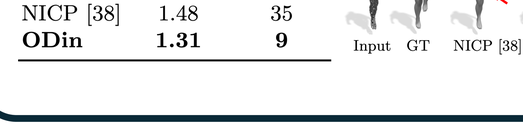
3D-CODED : 3D Correspondences by Deep Deformation, Groueix et al., 2018

But in the real world, it is inherently ambiguous



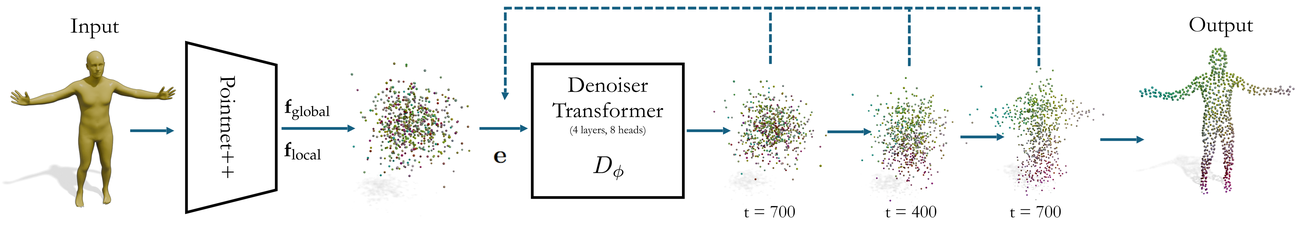
NICP: Neural ICP for 3D Human Registration at Scale, Marin et al., 2024

Point diffusion often relies on **images** and does **not natively support point order**



PC 2 : Projection-Conditioned Point Cloud Diffusion for Single-Image 3D Reconstruction, Melas-Kyriazi, 2023

We Diffuse Geometry in Correspondence with a Template



a) Given the input point cloud, we extract **local and global features**

b) During the denoising process, we **add the local features to every point** via nearest neighbors

c) We train a denoiser transformer that outputs the **ordered position w.r.t. the SMPL template** (690 vertices)

Tokenization

$$\mathbf{e} = \left[\mathbf{v}^1, \mathbf{v}^2, \ldots, \mathbf{v}^{689}, \hat{\mathbf{f}}_{\text{global}}, \hat{\mathbf{t}}_{\text{emb}}\right]^\top \in \mathbb{R}^{691 \times d}$$

With: $\mathbf{v}^i = \hat{\mathbf{x}}_t^i + \hat{\mathbf{f}}_{\text{local}}^i + \text{PE}(i)$

The positional encoding is crucial to inform the denoiser about point order.

$$\text{PE}(i, 2k) = \sin\left(\frac{i}{10000^{2k/d}}\right),$$

$$\text{PE}(i, 2k+1) = \cos\left(\frac{i}{10000^{2k/d}}\right),$$

$$k = 0, 1, \ldots, \frac{d}{2} - 1.$$

Training

1. **GT mesh → ordered point cloud**

$$\mathbf{x}_0 \in \mathbb{R}^{689 \times 3}$$

2. **Sample t and add Gaussian noise**

$$\mathbf{x}_0 \rightarrow \mathbf{x}_t$$

3. **Extract scan features**

$$\mathbf{f}_{global},\ \mathbf{f}_{local}$$

$$L_{\text{diff}} = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,\mathbf{I})}\left[\left\|\epsilon - D_\phi(\mathbf{x}_t, \mathbf{f}_{global}, \mathbf{f}_{local}, t)\right\|_2^2\right]$$

Train denoiser and feature extractor **end-to-end**.

Body Model Recovery

Diffusion yields 689 SMPL vertices. To recover the full SMPL instance, we run two further optimizations.

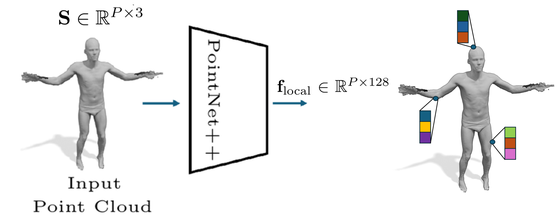
1) Fit SMPL to prediction via `smplfitter`
2) Refine SMPL via Chamfer loss to scan:

$$L = L_{\text{Chamfer}} + \text{reg}(\theta, \lambda) + \gamma\|\beta\|_2^2$$

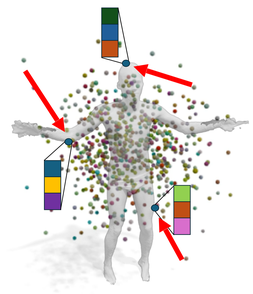
$$\text{reg}(\theta, \lambda) = \lambda \cdot \sum_{i=1}^{m} \|\text{VPoser}(\theta)_i\|_2^2$$

Local Feature Assignment

1 The co-trained PointNet++ yields a **feature vector** for **every input scan point**.



2 Each diffusion vertex is assigned the **local feature** of its **closest scan point**.



3 At diffusion steps $t \approx 1000$, these assignments are largely arbitrary, …

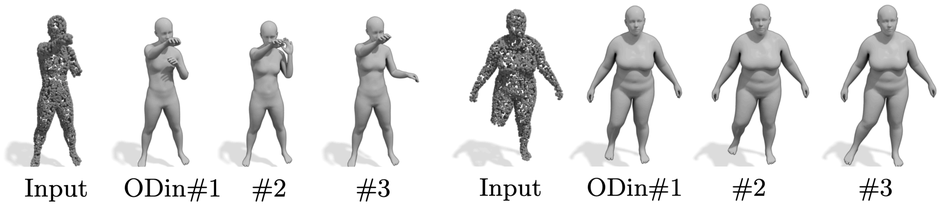


4 … but as t approaches 0, local features become a **highly informative conditioning** source.



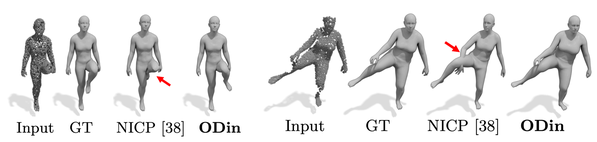
ODin Performs Strongly on Full and Partial Scans

	DFAUST err (cm) ↓	DFAUST$_{ml}$ err (cm) ↓
ODin	**1.15/0.79**	**4.80/3.87**
NICP	1.47/1.00	5.75/5.38
ODin$_{ml}$	**1.14/0.78**	**2.22/2.14**
NICP$_{ml}$	1.49/0.99	2.56/2.59



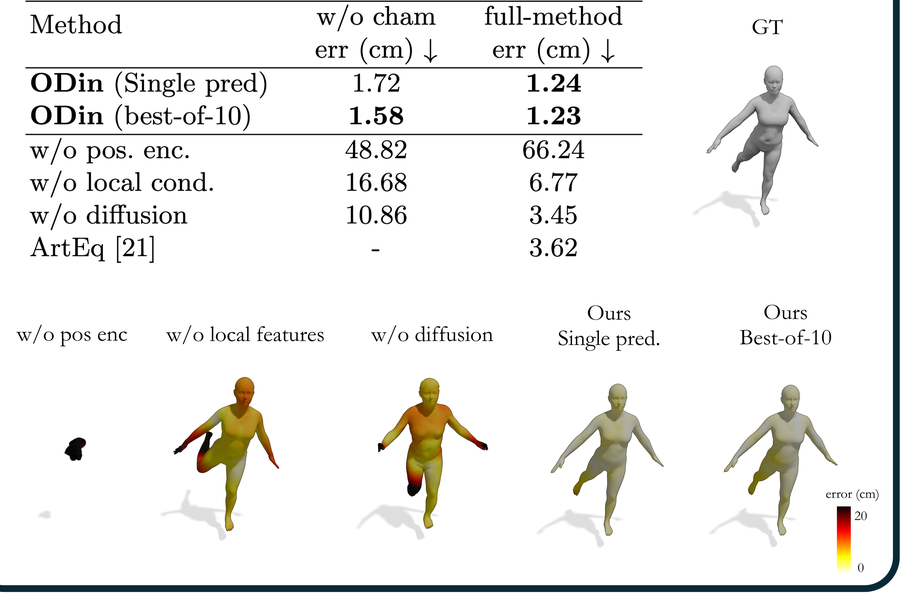
ODin Generalizes Well and Is Faster

FAUST	err (cm) ↓	time [s] ↓
NICP [38]	1.48	35
ODin	**1.31**	**9**



Ablation

Method	w/o cham err (cm) ↓	full-method err (cm) ↓
ODin (Single pred)	1.72	**1.24**
ODin (best-of-10)	**1.58**	**1.23**
w/o pos. enc.	48.82	66.24
w/o local cond.	16.68	6.77
w/o diffusion	10.86	3.45
ArtEq [21]	-	3.62



Open Directions

- Incorporate appearance / other conditioning sources
- More diffusion techniques
- Extend to Flow Matching