NEURAL INFORMATION PROCESSING SYSTEMS

Learning 3D Equivariant Implicit Function with Patch-Level Pose-Invariant Representation

NeurIPS 2024

Xin Hu，Xiaole Tang, Ruixuan Yu, Jian Sun

NEURAL INFORMATION PROCESSING SYSTEMS

Definition of Neural Vector Fields (NVF)

Given a sparse point cloud $X \in \mathbb{R}^{N_x \times 3}$ sampled on a shape $\mathcal{X}$, and a query set $Q \in \mathbb{R}^{N_q \times 3}$ sampled near the surface of $\mathcal{X}$, where N_x and N_q represent the number of the input points and query points respectively.

A shape $\mathcal{X}$ is defined as the zero displacement of the implicit function $\mathcal{F}$

$$\mathcal{X} = \{x \in \mathbb{R}^3 | \mathcal{F}(x) = \vec{0}\},$$

where x is a point in point cloud X, containing its spatial coordinate. $\vec{0}$ represents the zero displacement of point x.

Problem Statement for Equivariant Neural Vector Fields (NVF)

For a query point $q \in \mathbb{R}^3$, the implicit function $\mathcal{F}$ is formulated by

$$\mathcal{F}(q) = \Delta q = \hat{x} - q, \qquad where\ \hat{x} = \mathrm{argmin}_{x \in \mathcal{X}} \|x - q\|,$$

and $\hat{x}$ is the nearest point of query q on the $\mathcal{X}$.

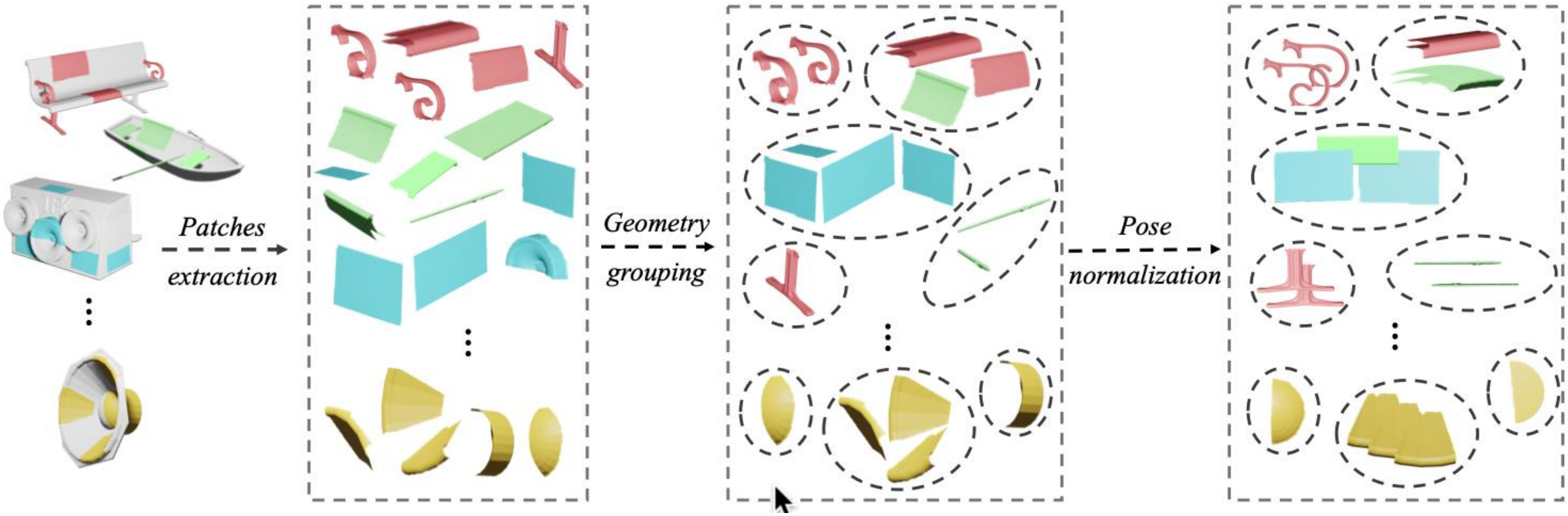
Definition 1 (Equivariant Implicit Function). Given an abstract group G, the implicit function $\mathcal{F}$ based on NVF is equivariant with regard to G, if

$$\mathcal{F}(\zeta \circ q) = \zeta \circ \mathcal{F}(q) = \Delta q, \qquad \forall \zeta \in G,$$

where q is a query point near or on the surface of $\mathcal{X}$. In this work, the group G is *SE(3)*.

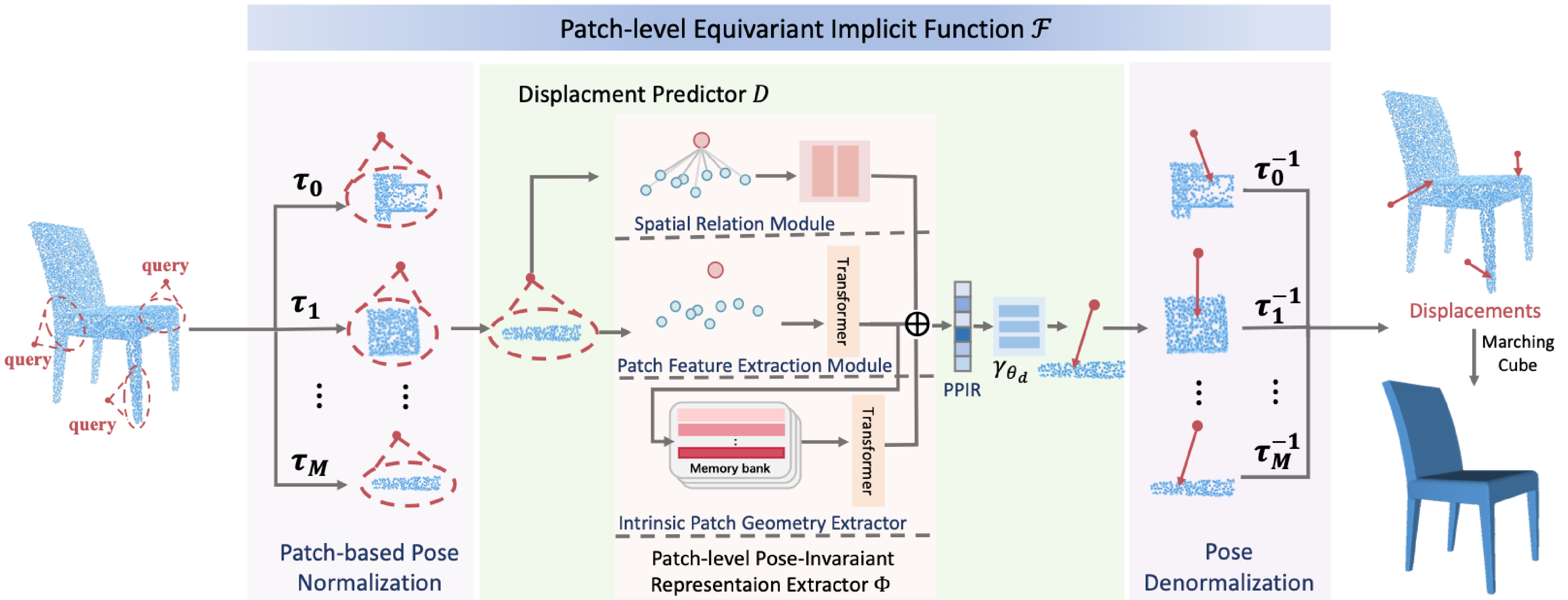
Motivation

Local 3D patches may exhibit geometric similarity, but with different poses. When the pose is removed, these local regions appear repeatedly.



Patch-level Equivariant Implicit Function (PEIF)

- Given a query point q, the corresponding patch for q on point cloud X is $P = \{p_i\}_{i=0}^{K}$, i.e., the KNN of q based on Euclidean distance.
- The patch P and query point q are normalized by patch-based pose normalization τ, obtaining *SE(3)*-invariant ones $\{\tau(P), \tau(q)\}$.
- Taking $\{\tau(P), \tau(q)\}$ as input, the displacement predictor D learns *SE(3)*-invariant representation and predicts the displacement $\Delta\bar{q}$.
- This predicted displacement is transformed back to the pose of P with τ^{-1}.


Patch-level Equivariant Implicit Function $\mathcal{F}$
query
query
query
τ_0
τ_1
τ_M
Patch-based Pose Normalization
Displacment Predictor D
Spatial Relation Module
Transformer
Patch Feature Extraction Module
Memory bank
Transformer
Intrinsic Patch Geometry Extractor
Patch-level Pose-Invaraiant Representaion Extractor Φ
PPIR
γ_{θ_d}
τ_0^{-1}
τ_1^{-1}
τ_M^{-1}
Pose Denormalization
Displacements
Marching Cube

➢ Pose Normalization

The patch P is first decentralized by subtracting the center μ of the points, then the rotation matrix U is obtained by computing the Singular Value Decomposition (SVD) over the covariance matrix $(P-\mu)^T(P-\mu)$. The pose-normalized patch $\bar{P}$ and query point $\bar{q}$ are obtained as follows:

$$\bar{P} \triangleq \tau(P) = (P-\mu)U, \qquad \bar{q} \triangleq \tau(q) = (q-\mu)U.$$

➢ Displacement Predictor Design

Taking the pose-normalized patch $\bar{P}$ and query $\bar{q}$ as input, the displacement predictor D is designed to predict the displacement $\Delta\bar{q}$.

Spatial Relation Module: $h_{\bar{q}} = \mathrm{SRM}(\bar{P}, \bar{q})$

Patch Feature Extraction Module: $f_{\bar{q}}, \{f_{p_i}\}, f_{\bar{P}} = \mathrm{PFEM}(\bar{P}, \bar{q})$

Intrinsic Patch Geometry Extractor: $g_{\bar{P}} = \mathrm{IPGE}(f_{\bar{P}})$

The displacement $\Delta\bar{q}$ for query point $\bar{q}$ can then be derived by

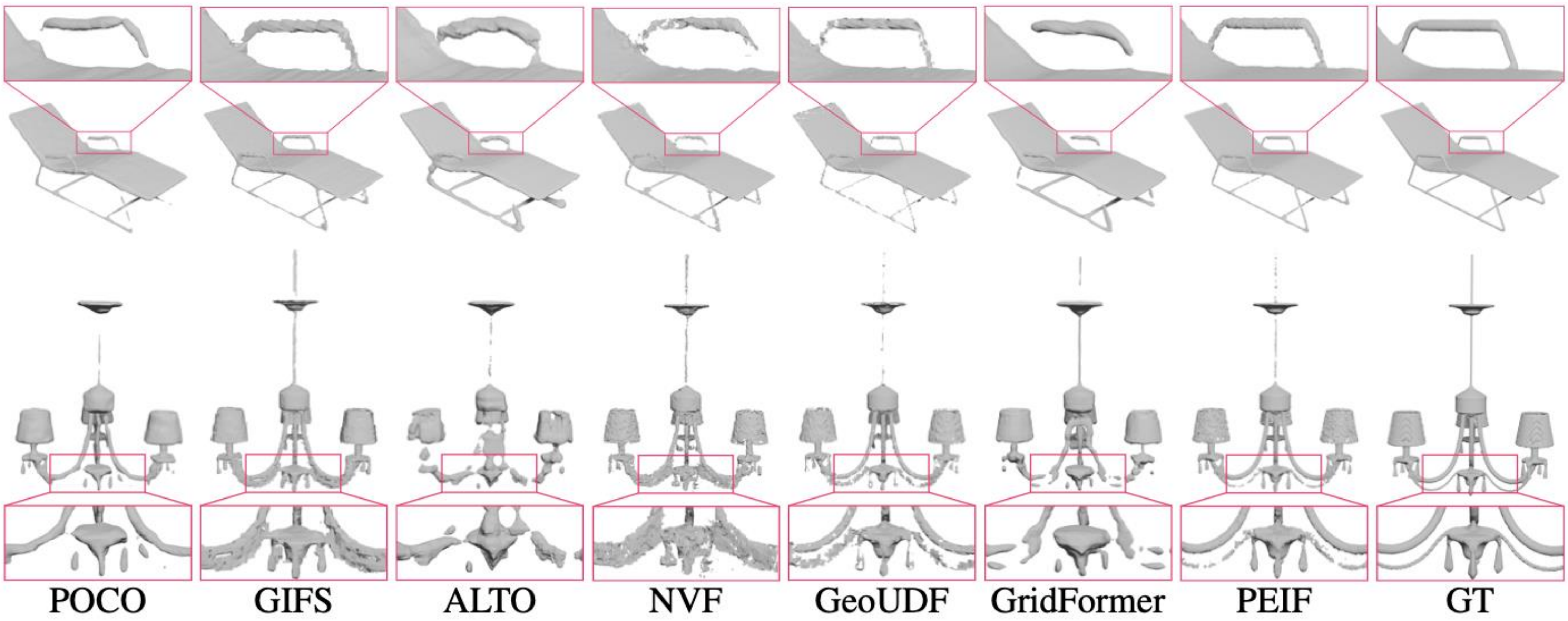
$$\Delta\bar{q} = \gamma_{\theta_d}\left(\gamma_{\theta_a}\left(h_{\bar{q}} \oplus f_{\bar{q}} \oplus \{f_{p_i}\}\right) \oplus g_{\bar{P}}\right).$$

➢ Pose Denormalization

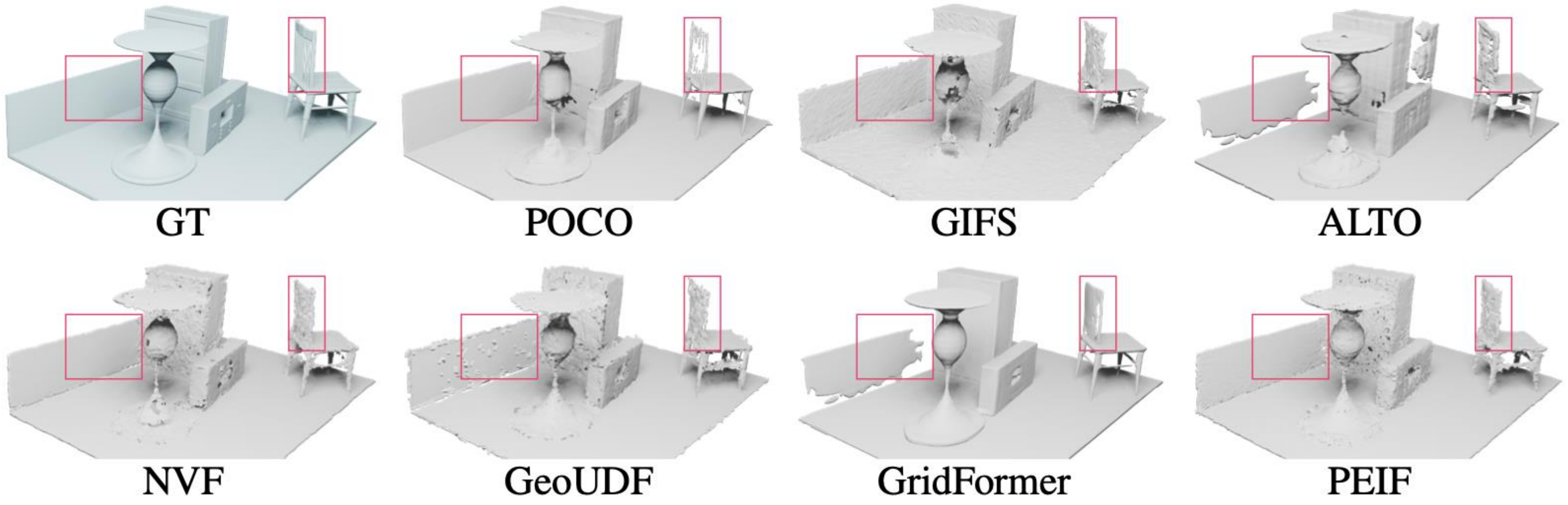
The final *SE(3)*-equivariant displacement is obtained by transforming back $\Delta\bar{q}$ with pose denormalization

$$\Delta q = \tau^{-1}(\Delta\bar{q}).$$

Experiments

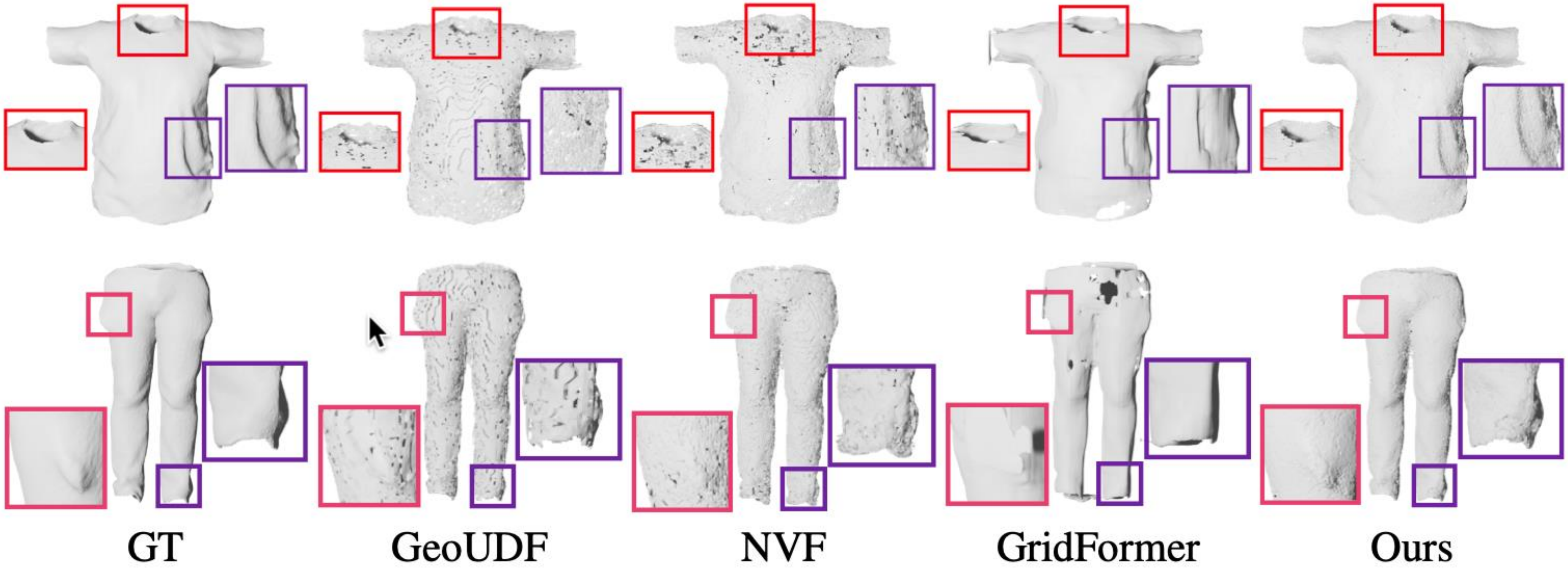


Experiments



NEURAL INFORMATION PROCESSING SYSTEMS

Experiments



NEURAL INFORMATION
PROCESSING SYSTEMS

Tank You！