DispVoxNets: Non-Rigid Point Set Alignment with Supervised Learning Proxies

Soshi Shimada¹², Vladislav Golyanik³, Edgar Tretschk³, Didier Stricker¹², Christian Theobalt³
¹University of Kaiserslautern ²DFKI ³Max Planck Institute for Informatics, SIC

Non-Rigid Point Set Registration

- Recovery of a displacement field aligning template (Y) and reference (X) point sets as well as correspondences between them.

DispVoxNets Pipeline

- Our pipeline is composed of two stages: Displacement Estimation (DE) and Refinement stage.
- DE stage regresses global displacements between Y and X.
- Refinement stage improves the initial displacements.

DispVoxNets Structure

- Y: template  X: reference
- Q: size of the voxel grid
- M: number of points in Y
- N: number of points in X
- \( \ell_{\text{P2V}} \): point-to-voxel conversion
- \( \ell_{\text{V2P}} \): voxel-to-point conversion

DispVoxNets Loss Functions

- **Displacement loss** penalises the discrepancy between the output displacements and ground truth displacements.

\[ \ell_{\text{Disp}}(Z, V_Y, V_X) = \frac{1}{M} \sum_{y \in Y} \| Z - D_{\text{nn}}(V_Y, V_X) \|_2^2 \]

- **Point projection loss** penalises the Euclidean distances between a point \( y' \) in \( Y' + (Y, X) \) and its closest point \( x_y \) in \( X \).

\[ \ell_{\text{PP}}(Y' + (Y, X), X) = \frac{1}{M} \sum_{y \in Y} \| y' - x_y \|_2 \]

Datasets

- Real Scan Data
- Deteriorated Data
- Clean Data
- Uniform Noise
- Outlier
- Missing Data

Trilinear Weighing

I. Compute trilinear weights to estimate sub-voxel displacements.
II. Record the weights and indices of the 8 nearest displacements in the affinity table.
III. Compute the point projection loss.
IV. Distribute gradients following the IDs and weights information recorded in the affinity table in II.