Wasserstein Hypergraph Learning for General 3D Point Cloud Alignment

UC San Diego
HALICIOĞLU DATA SCIENCE INSTITUTE

Chanyoung Park (chp026@ucsd.edu), Minghan Wu (miw039@ucsd.edu), Gal Mishne (gmishne@ucsd.edu)(Mentor), Yusu Wang (yuw122@ucsd.edu)(Mentor)
University of California, San Diego

Objectives

- Develop robust 3D point cloud alignment[1] using hypergraph learning[2]
- Evaluate performance across rigid, affine, and piecewise-rigid settings
- Improve correspondence reliability under noise, partial overlap, and sampling variation

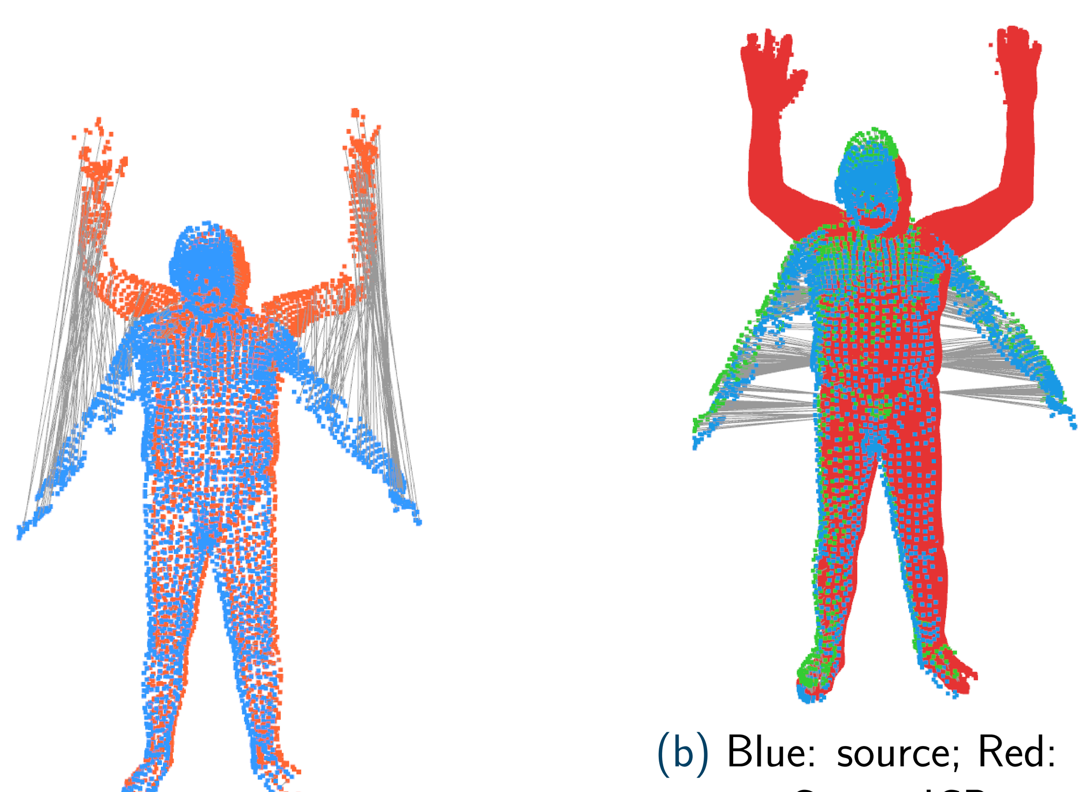
Introduction

Related Topics: 3D point cloud alignment, Hypergraph learning, Wasserstein-based aggregation

Significance

- Human body pose tracking (piecewise motion of limbs)
- Multi-view 3D reconstruction under deformation
- Medical shape alignment (anatomical structures)

Datasets



(a) Correspondence

(b) Blue: source; Red: target; Green: ICP alignment

Figure: FAUST Example

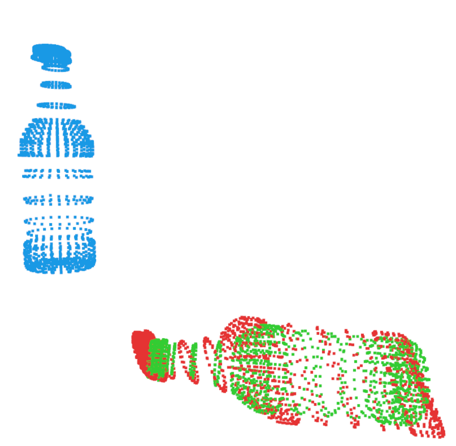
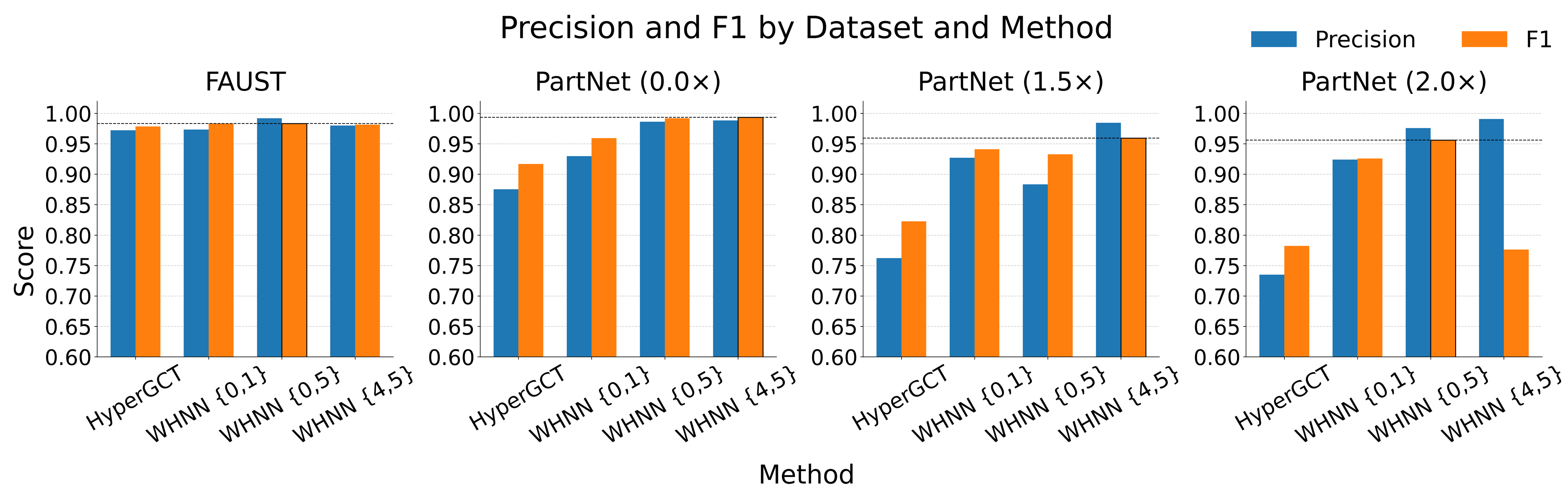


Figure: Partnet affine example. Blue: source; Red: target; Green: ICP alignment

Main Result

Wasserstein pooling better preserves local geometric structure under deformation and noise, leading to more reliable correspondences in affine / piecewise transformations.



Methods

- **Nodes:** A candidate source-target correspondence between two point clouds.
- **Hyperedges:** Groups of geometrically consistent correspondences

Wasserstein Slice

Wasserstein-2 distance:

$$W_2(P,Q) = \left(\inf_{\gamma \in \Gamma(P,Q)} \int_{\mathbb{R}^n \times \mathbb{R}^n} d(x,y)^2 \, d\gamma(x,y) \right)^{1/2},$$

Sliced Distances:

$$SW_2(P,Q) \approx \left(\frac{1}{L} \sum_{l=1}^{L} W_2^2(\mathcal{P}_{\theta_l} P, \mathcal{P}_{\theta_l} Q) \right)^{1/2},$$

OT Aggregation: Replace mean pooling with *Wasserstein pooling* in selected layers

- Pick a reference distribution and a set of 1D projection directions; compute projected transport features between each hyperedge and the reference distribution
- Use these projected transport features to embed hyperedges (nodes → hyperedges); the reverse update (hyperedges → nodes) is analogous

Alignment:

- **Hypothesis generation:** Build globally consistent correspondences using greedy expansion or two-stage spectral matching.
- **Output:** A one-to-one, geometrically consistent correspondence set for final alignment.

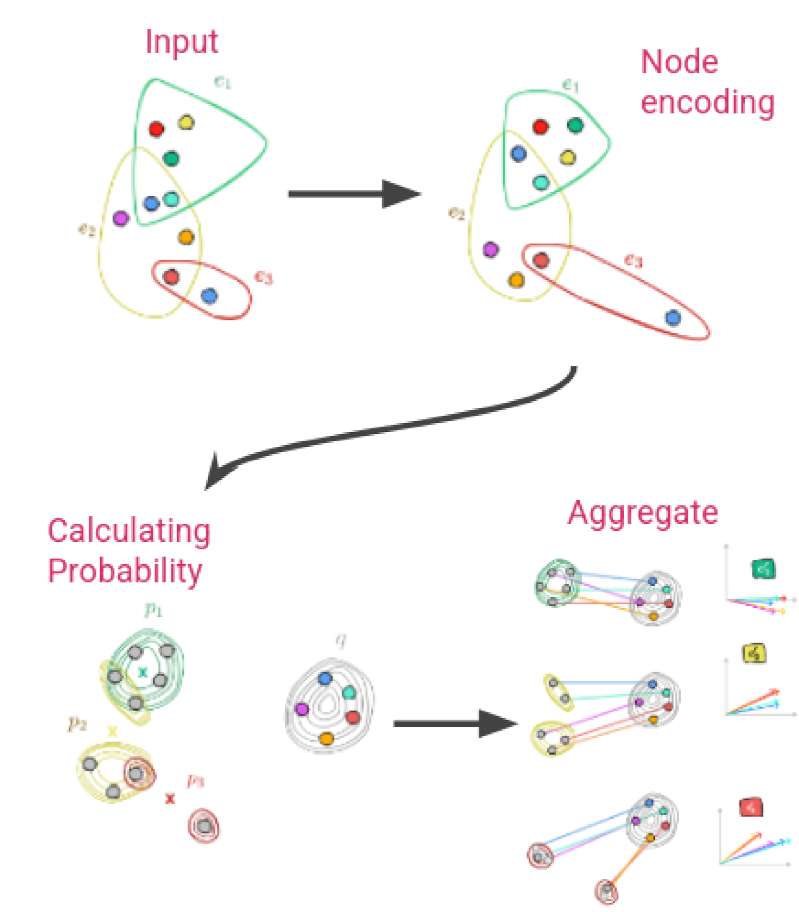


Figure: WHNN[4]

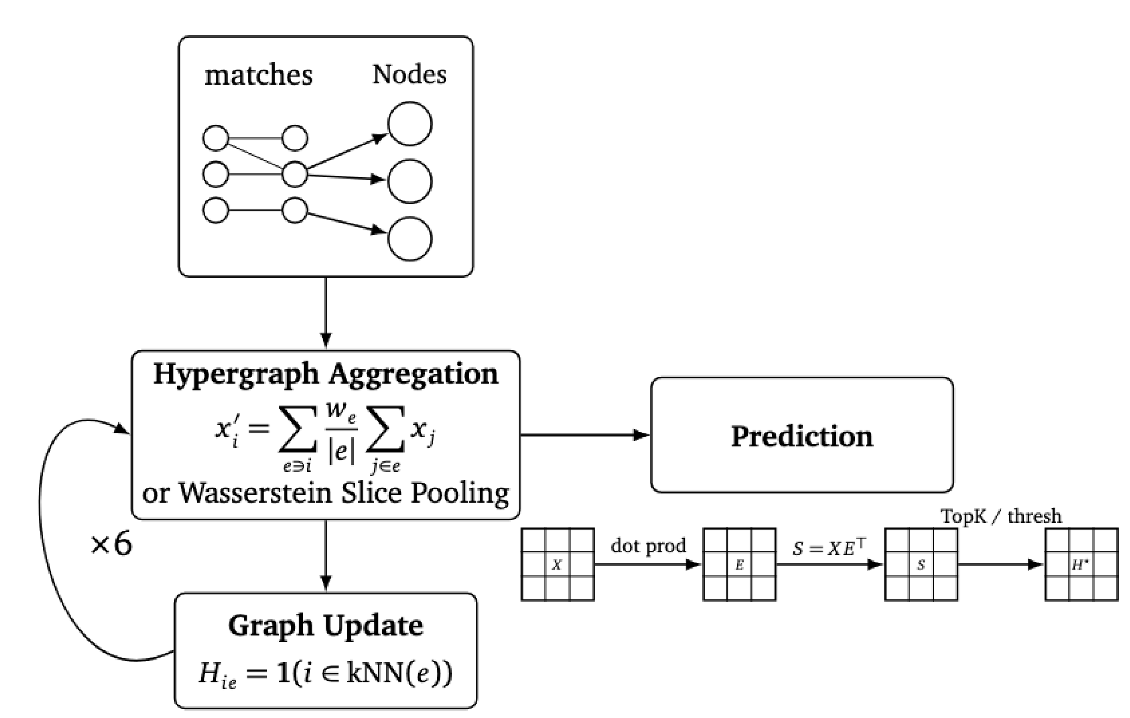
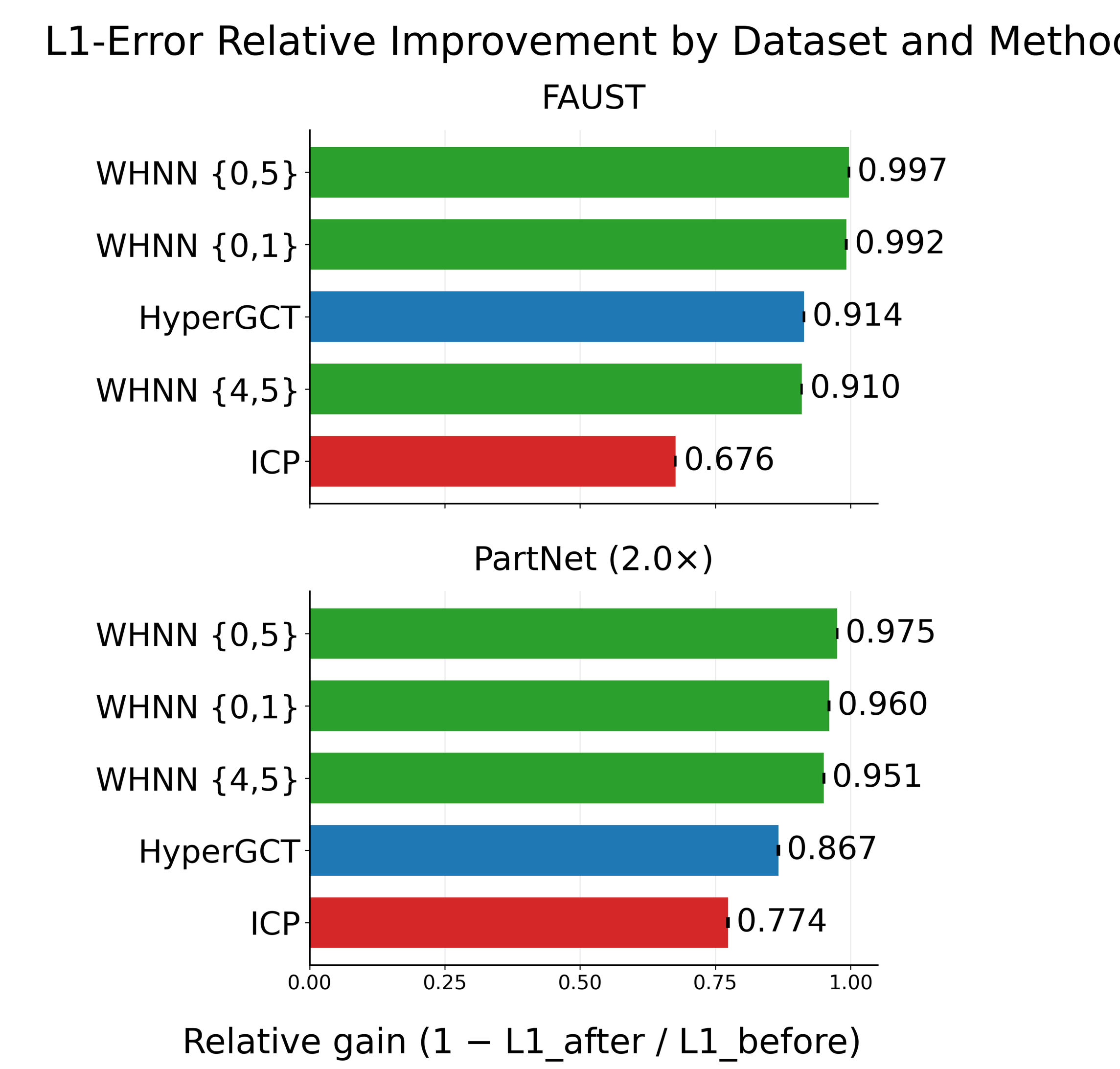


Figure: HyperGCT

Results



Dataset	Best Model	F1	Prec.	ΔF1
FAUST (pw-rigid)	WHNN all	0.991	0.991	0.013
PartNet (rigid)	WHNN {4,5}	0.993	0.988	0.076
PartNet (1.5× aff.)	WHNN {4,5}	0.959	0.984	0.136
PartNet (2.0× aff.)	WHNN {0,5}	0.956	0.976	0.174

Conclusion

- At least one Wasserstein pooling configuration outperformed the mean-pooling HyperGCT baseline in every evaluated regime.
- The best layer placement depended on the transformation setting, with different optima across piecewise-rigid, rigid, and affine cases.

References

[1] Besl, P. J., and McKay, N. D. *A Method for Registration of 3-D Shapes.* IEEE TPAMI, 1992.

[2] Feng, Y. et al. *Hypergraph Neural Networks.* AAAI, 2019.

[3] Zhang, X. et al. *HyperGCT: A Dynamic Hyper-GNN-Learned Geometric Constraint for 3D Registration.* ICCV, 2025.

[4] Duta, I., and Liò, P. *Wasserstein Hypergraph Neural Network.* arXiv preprint, 2025.