

General Part Assembly Planning

Anonymous Author(s)

Affiliation

Address

email

1 A Additional Results

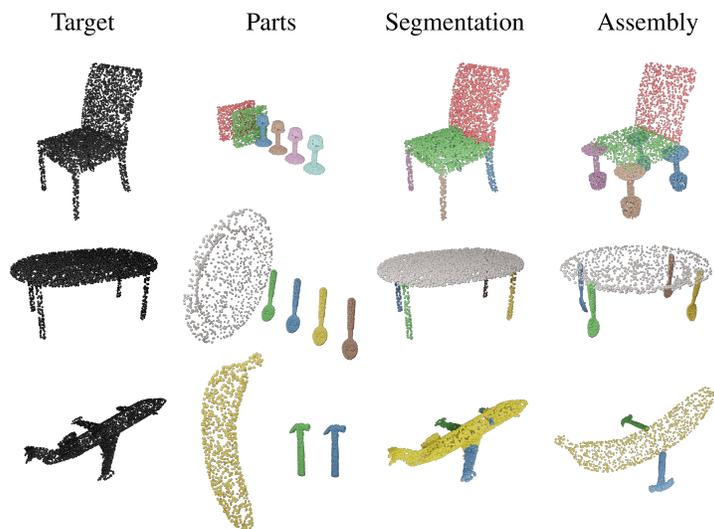
2 **Additional Visualization.** Fig. 5 and Fig. 6 show additional results on simulated and real-world
3 data, respectively.

4 **Quantitative Results for Categories.** Table 1 shows detailed quantitative evaluation for unseen
5 instances for seen categories (Chair, Lamp, Faucet) and unseen categories (Table, Display).

Table 1: Quantitative results of our algorithm on different categories.

	Canonical Pose						Random Pose					
	Precise Part			Imprecise Part			Precise Part			Imprecise Part		
	CD	PA	SR	CD	PA	SR	CD	PA	SR	CD	PA	SR
Chair	7.7	57.7	19.3	7.3	64.4	25.1	7.6	58.4	19.3	8.3	63.0	24.0
Lamp	7.6	66.9	29.1	5.9	72.1	40.9	8.1	64.2	26.2	6.0	74.4	45.5
Faucet	7.3	65.6	24.4	6.5	63.1	20.7	7.4	63.6	20.6	6.5	62.3	22.4
Table	7.5	52.0	20.6	7.0	55.1	21.5	8.1	50.8	17.8	6.9	55.7	22.2
Display	4.2	59.2	23.2	4.7	59.3	20.2	4.4	60.8	23.8	4.9	58.5	16.9

6 **GPAT builds creative assemblies.** To fully test the generalization abilities of GPAT, we provide
7 unseen part shapes like a banana, hammers, and forks as parts to create novel targets such as a plane.
8 Our model predicts creative assemblies given target shapes from unseen categories and non-exact
9 parts, as seen in Fig. 1. The shapes are taken from PartNet [1], ModelNet40 [2], and YCB dataset [3].



10 **Figure 1: Creative Assemblies.** Our model predicts creative assemblies given target shapes from unseen categories and non-exact parts. Top row: a chair assembled with lamps as chair legs. Middle row: a table assembled with a plate and spoons. Bottom row: a plane assembled with a banana and hammers.

11 **GPAT is applicable to part discovery.** GPAT is directly applicable to the task of part discovery,
 12 i.e., predict a part segmentation given a target [4], if we do not provide input parts. We show some
 13 qualitative results in Fig 2 to test GPAT’s part discovery abilities. Given non-exact parts, PAT predicts
 14 accurate segmentations as usual. If we input identical blocks, which specifies the number of parts
 15 but provides little information about the part shapes, then GPAT predicts reasonable segmentations
 16 with the specified number of segments. Finally, we omit the input parts, and GPAT successfully
 17 discovers parts in the target shape.

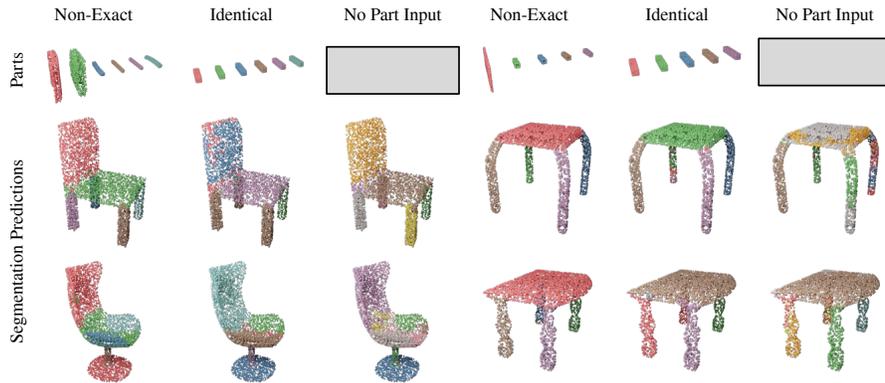


Figure 2: **Application to Part Discovery.** Given non-exact matching parts (Non-Exact), identical blocks (Identical), and no part point clouds input, GPAT predicts reasonable part segmentations of the target.

18 **GPAT is aware of part scales.** Part assembly often involves parts that have the same geometry
 19 but different scales, so it is necessary for a model to discriminate parts of different scales to create
 20 correct assemblies. As an qualitative illustration in Fig. 3, we adjust the scale of the parts that have
 21 same geometry (the legs of chair/table), and the model correctly associates parts of different scales
 22 to the target to build the desired assemblies.

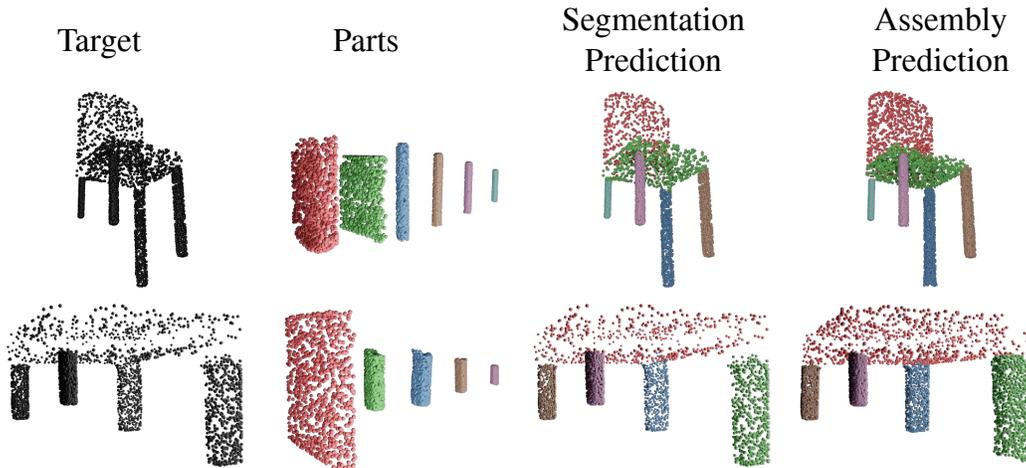


Figure 3: **Sensitivity to Scale.** The legs of the chair/table are manually scaled, and the model correctly associate parts of the same shape but different sizes.

23 **Optimization is prone to local minima.** The optimization baseline (Opt) achieves the lowest
 24 chamfer distance (CD) in some scenarios, but its part accuracy (PA) and success rate (SR) are
 25 significantly lower. As seen in Fig. 4, directly optimizing the part poses to match the target often
 26 result in predictions at local minima where the predicted assembly matches the contour of the target,
 27 but the assembly makes no semantic sense.

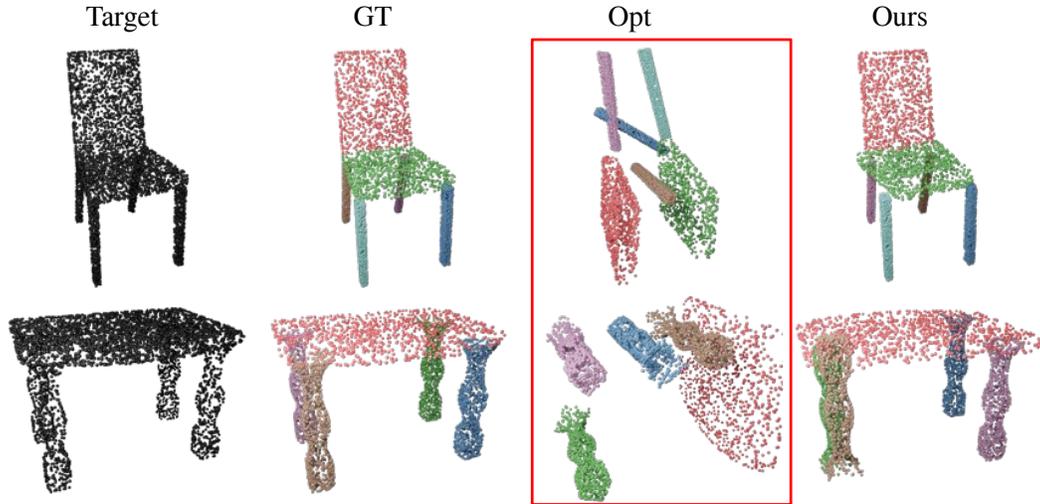


Figure 4: **Optimization is prone to local minima.**

28 B Data and Training Details

29 We use Furthest Point Sampling (FPS) to sample 1,000 points for each part point cloud and 5,000
 30 points for each target point cloud. Followed the previous work [6], we also zero-center all the point
 31 clouds, and align the principle axes of the part point clouds with the world axes using Principle
 32 Component Analysis (PCA).

33 In our training, we downsample the target point features by a factor of 10, so for each sample, we
 34 obtain 500 target point features. We use a feature dimension of 256 and we use 8 GPAT layers, with
 35 k values of 16, 16, 32, 32, 64, 64, 500, 500. We use Adam [7] with a learning rate of 0.00004, a batch
 36 size of 36. We train for 2000 epochs in total.

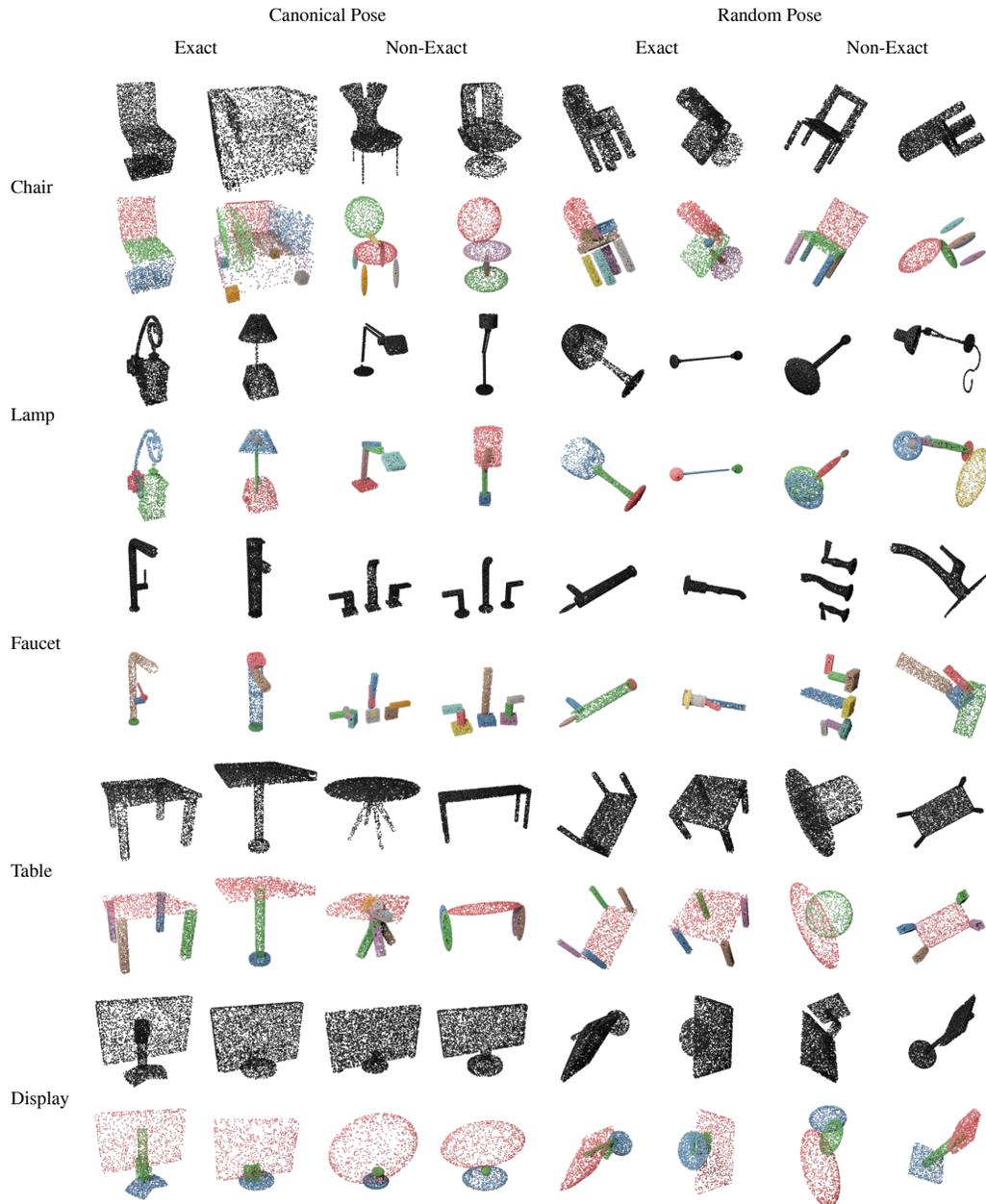


Figure 5: **Qualitative Results.** Chairs, lamps, and faucets are seen during the training. Tables and displays are unseen categories. The first row of each category displays targets in black, and the second row shows our predictions.

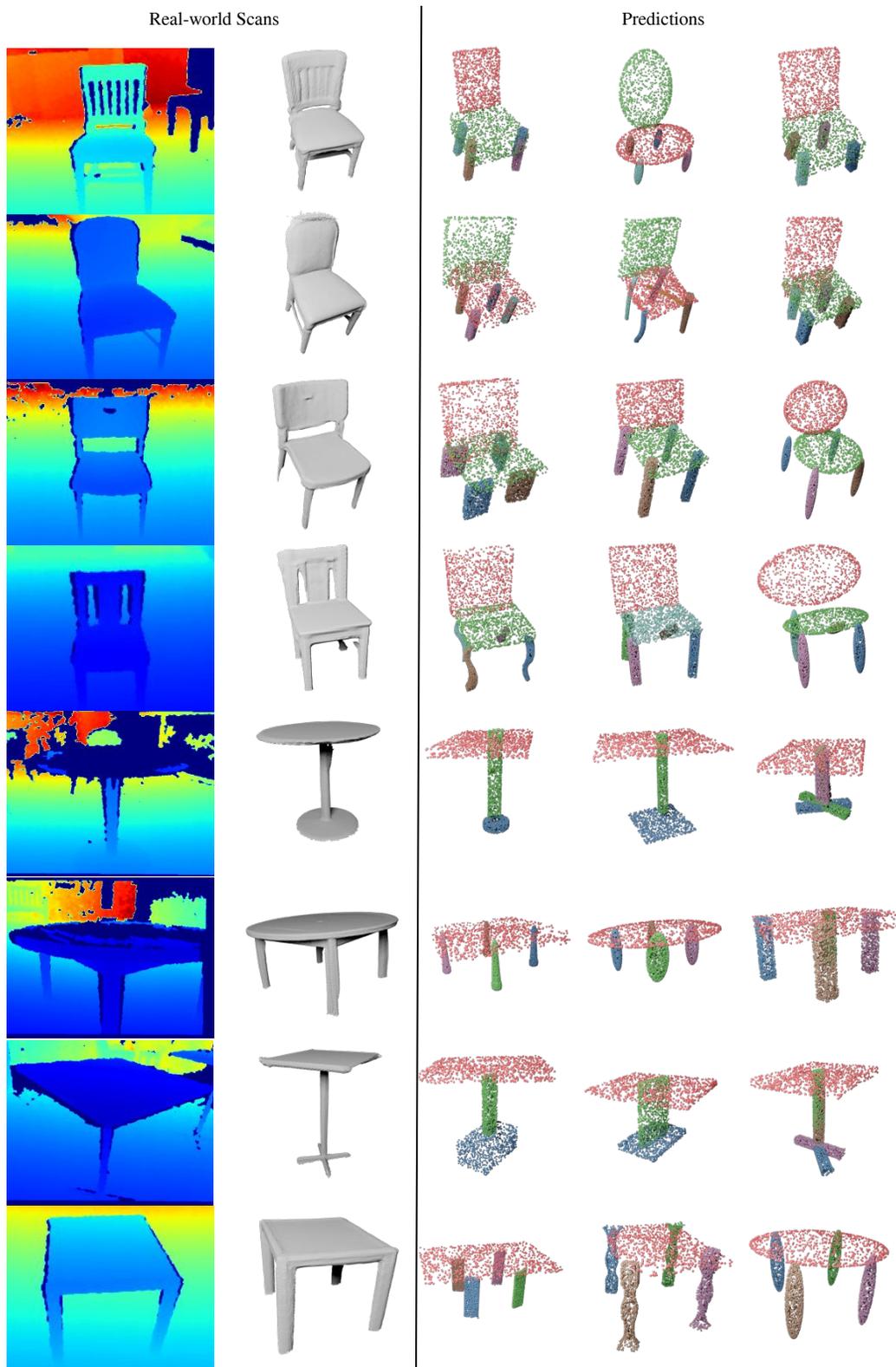


Figure 6: **Results on Real-world Data.** Non-exact parts from the same category as the target point cloud, which are teal-world scans taken from Redwood dataset [5].

37 **References**

- 38 [1] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. Su. Partnet: A large-scale
39 benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings*
40 *of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019.
41 1
- 42 [2] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep
43 representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision*
44 *and pattern recognition*, pages 1912–1920, 2015. 1
- 45 [3] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar. Benchmarking
46 in manipulation research: The ycb object and model set and benchmarking protocols. *arXiv*
47 *preprint arXiv:1502.03143*, 2015. 1
- 48 [4] T. Luo, K. Mo, Z. Huang, J. Xu, S. Hu, L. Wang, and H. Su. Learning to group: A bottom-up
49 framework for 3d part discovery in unseen categories. *arXiv preprint arXiv:2002.06478*, 2020.
50 2
- 51 [5] S. Choi, Q.-Y. Zhou, S. Miller, and V. Koltun. A large dataset of object scans. *arXiv preprint*
52 *arXiv:1602.02481*, 2016. 5
- 53 [6] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon. Dynamic graph cnn
54 for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019. 3
- 55 [7] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint*
56 *arXiv:1412.6980*, 2014. 3