Supplementary Document for
Unpaired Translation of 3D Point Clouds with Multi-part Shape Representation

Chih-Chia Li and I-Chen Lin
National Yang Ming Chiao Tung University, Taiwan

1 Additional Module Details

The overview of the proposed framework and architecture of several important modules are described in the main manuscript. Here, we introduce the architecture of the remaining modules in our framework.

1.1 Feature mapping module

The architecture of our feature mapping module is the multilayer perceptron (MLP) as shown in Figure 1a. We used multiple feature mapping modules to separately extract the part feature from the whole feature.

1.2 Point generator

Compared to other parts-to-whole generators, attention-based methods generate fewer outliers and network parameters and converges faster. Therefore, our point generator takes an attention-based architecture described in Figure 1b. We employed multiple part-specific generators to convert different part features into point clusters. These point clusters are then aggregated by our part aggregation module to generate the target point cloud.

1.3 Discriminator

As shown in Figure 2, our point discriminator takes point clouds as input and employs PointNet to extract a feature vector. Our discriminator has two branch outputs. They are used to estimate the likelihood of real point clouds $\mathbb{P}(S|P)$ and the likelihood of the domain class $\mathbb{P}(C|P)$, respectively.

2 Additional Ablation Study

2.1 Effectiveness of the part loss

We considered both global and local features in the cycle-consistency loss and feature preservation loss. To verify the effectiveness of the loss regarding parts, we show qualitative comparisons in Figure 3. Comparing the first column in Figure 3b and Figure 3c, the horizontal bars between the legs of the input chair can be preserved on the result with part loss. Besides, the part loss makes the thickness of the chair legs generated by the model more similar to the input (e.g. second and third columns in Figure 3c).

2.2 Effectiveness of the part aggregation module

Our proposed point aggregation module deforms each part according to the part and global information. The final target point cloud is the ensemble of refined parts. We demonstrate the qualitative comparisons in Figure 4. The results of the first and last rows in Figure 4b have discontinuities in shape, which can be mitigated with the PA module. Besides, comparing the last two rows of Figure 4e-f in the right half, it is evident that the PA module can locally adjust each part to make the distinctive structure of the transferred shape be more similar to that of the input (such as the legs of chairs and counterpart tables).

2.3 Different architecture settings

In our experiments of the main manuscript, we tried another architectural design as shown in Figure 5. In this network, we used a single feature mapping module and a generator to generate a point cloud of the entire object. To further refine the transferred result, we utilized multiple part aggregation modules to displace different point groups. To compare the performance between this alternative architecture and our

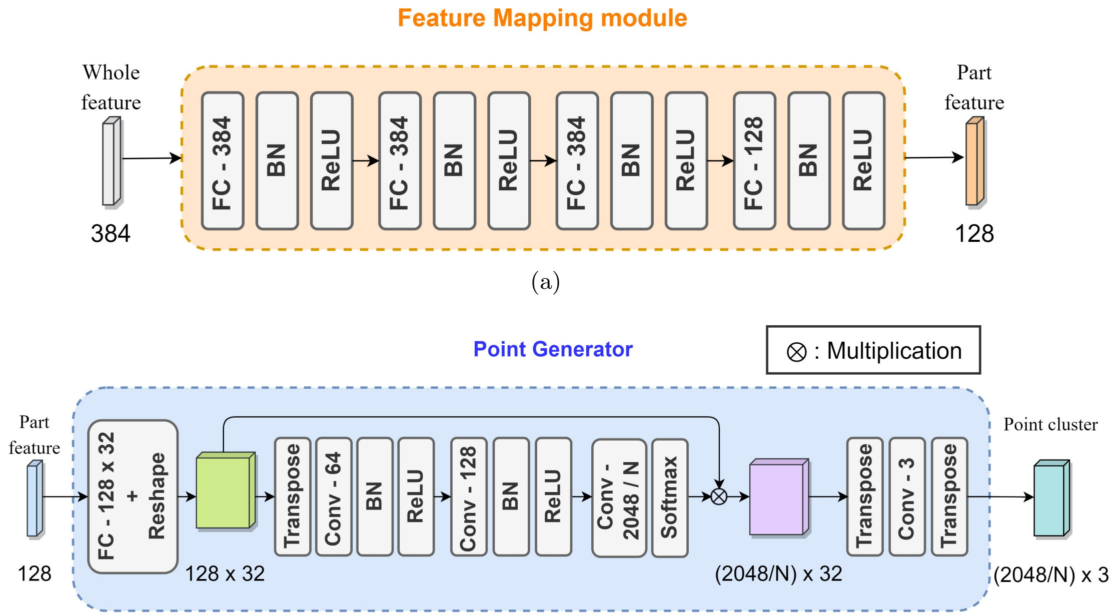

Figure 1: (a) The architecture of our feature mapping module. The feature mapping module is an MLP structure that can map the whole feature to different local features. (b) The architecture of the point generator, which is an attention-based network that transforms latent codes to point clouds. The variable N represents how many parts the point cloud is divided into.

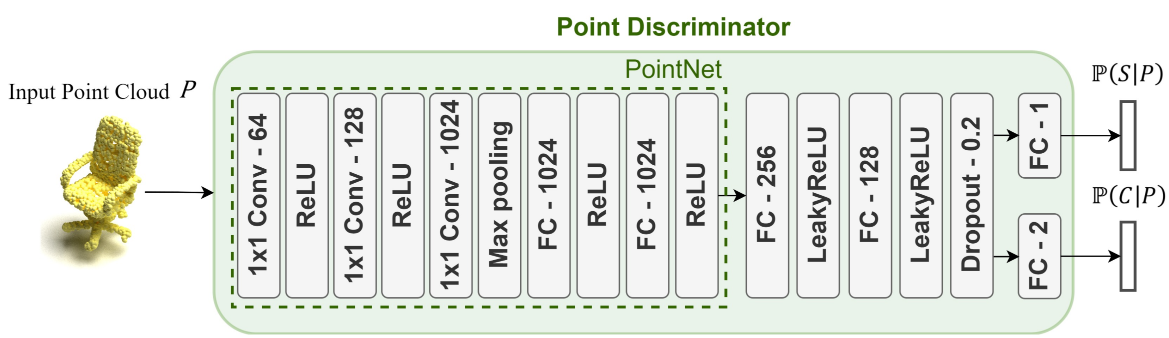

Figure 2: The architecture of our discriminator. The point discriminator takes a point cloud as input, and predicts the likelihood to be a real input $\mathbb{P}(S|P)$ and the likelihood of the domain class $\mathbb{P}(C|P)$.

proposed architecture, we visualize the translation results on the chair and table dataset in Figure 6.

Although using the methods in Table 6(e) can generate smoother point clouds, our proposed model better preserves the local details from the input shape. This may be because our method divides a point cloud into multiple parts first for later part-specific processing, besides the loss functions $\mathcal{L}_{cycle}$ and $\mathcal{L}_{fp}$ are also applied on each part. Our approach enables the model to learn local features and better know how to deform each part to form the target shape. We think the awareness of local structures of the proposed frame work copes with the core problem of unpaired shape translation, and a few rugged surface can be processed by post-process filtering or including additional loss functions for smoothness.

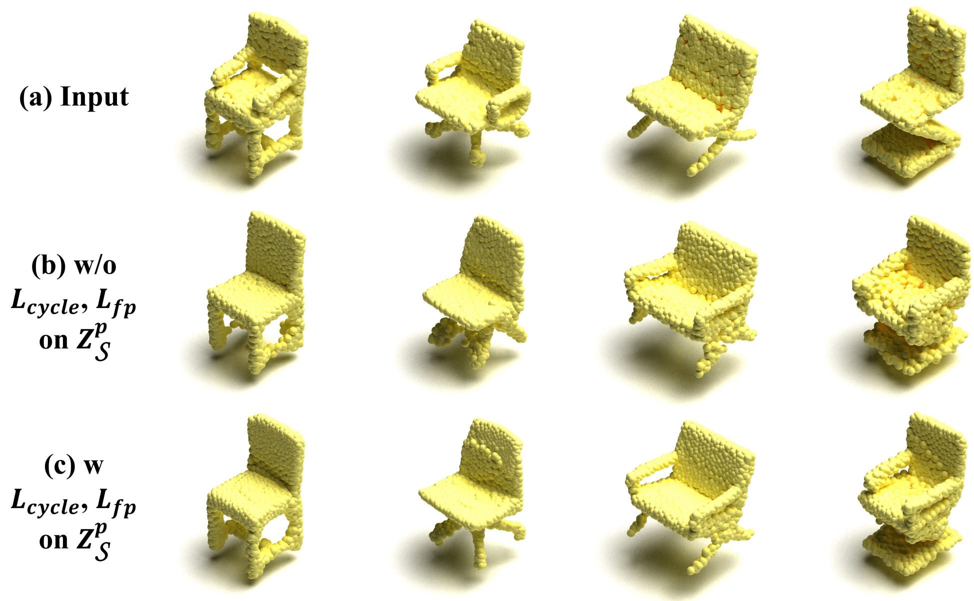

Figure 3: Qualitative comparison of results with and without part feature Z_S^p terms in the cycle-consistency loss and feature preservation loss. The first two columns are the results of *armchair* $\rightarrow$ *armless chair* transfer. The last two columns are the results of *armless chair* $\rightarrow$ *armchair* transfer.

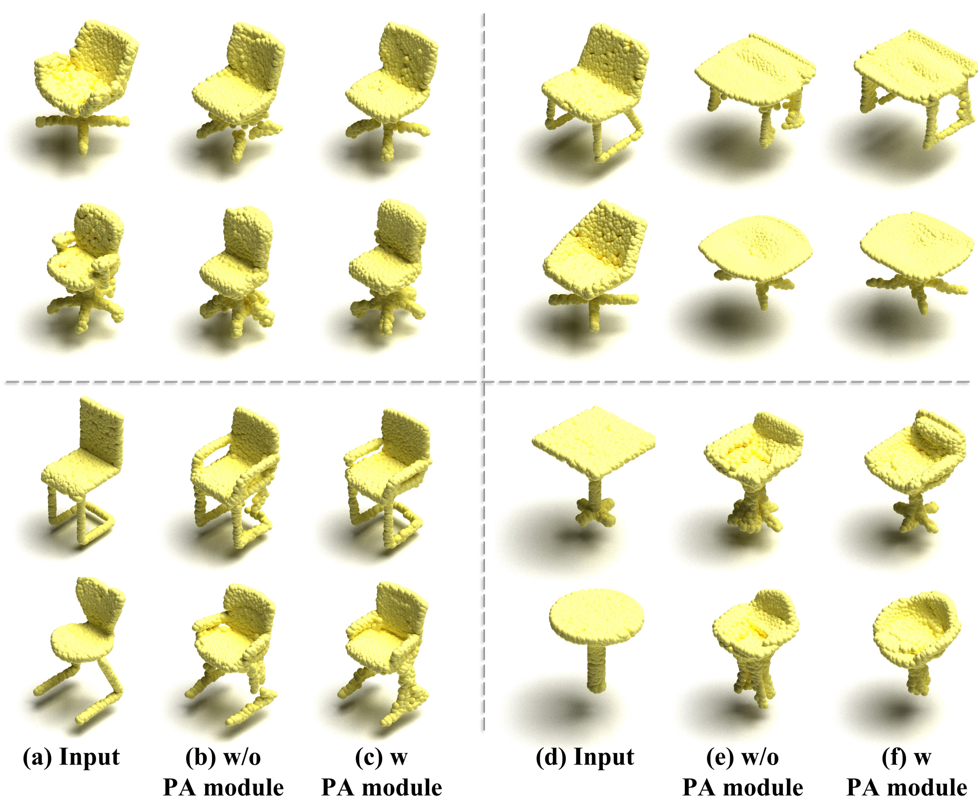

Figure 4: Qualitative comparison of results with and without part aggregation (PA) module. The left half is the translation between armchairs and armless chairs, and the right half is the translation between chairs and tables.

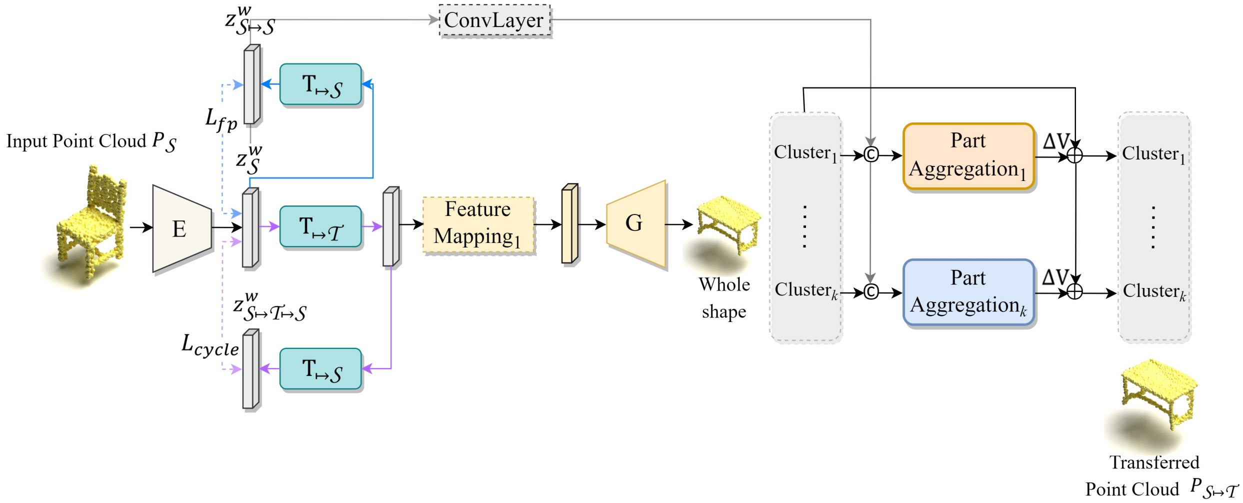

Figure 5: The architecture of the model described in Table 6(e) of the main manuscript.

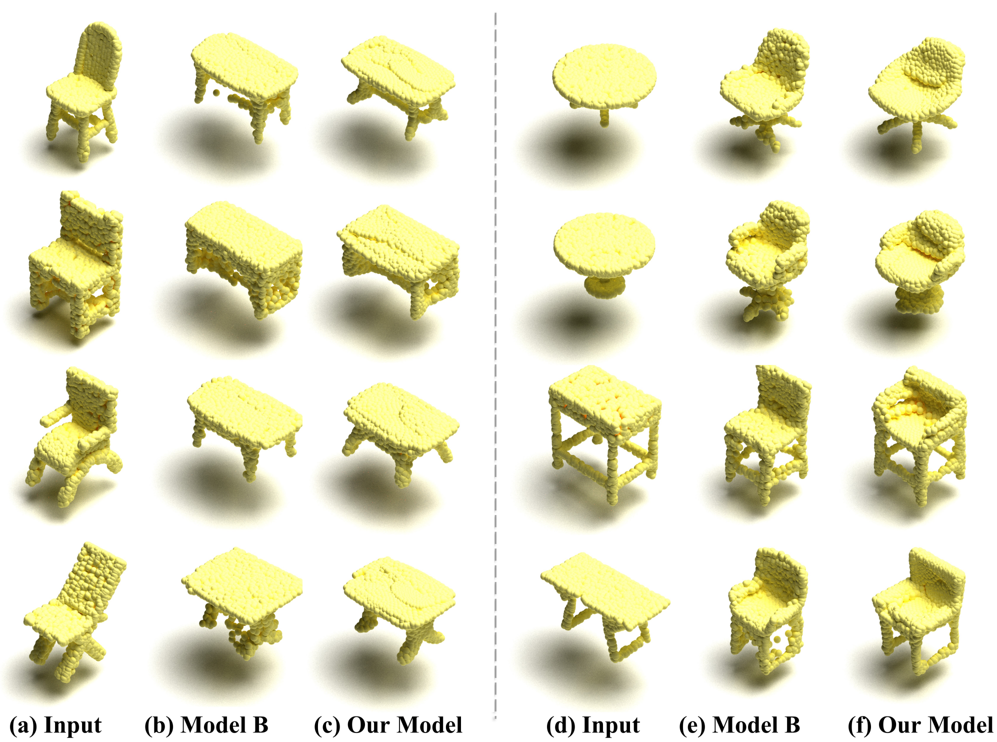

Figure 6: Qualitative comparison of results by an alternative architecture and ours. Model B is the method described in Table 6(e) of the main manuscript. The left half is the translation from chairs to tables, while the right half is the translation from tables to chairs.