Manual-PA: Learning 3D Part Assembly from Instruction Diagrams Supplementary Material

Jiahao Zhang[1] Anoop Cherian[2] Cristian Rodriguez[3] Yizhak Ben-Shabat[4] Stephen Gould[1]
[1]The Australian National University, [2]Mitsubishi Electric Research Labs
[3]The Australian Institute for Machine Learning
[4] [1]{first.last}@anu.edu.au [2]cherian@merl.com [3]crodriguezop@gmail.com [4]sitzikbs@gmail.com

Contents

A. Implementation Details

We train our model using AdamW [5] with a learning rate of 10^{-5} and a weight decay of 10^{-4}. For permutation learning, the learning rate decays by a factor of 0.9 every 5 epochs across a total of 50 epochs, whereas for pose estimation, it decays by the same factor every 50 epochs over 1000 epochs. All experiments are conducted on a single A100 GPU with 80GB of memory, requiring approximately 70 hours for the PartNet chair category, 40 hours for the table category and 16 hours for the storage category. Following [3], we set the hyperparameters as $\lambda_T = 1$, $\lambda_E = 1$, $\lambda_C = 20$, and $\lambda_S = 20$. The code and dataset will be made publicly available upon acceptance.

B. Metric Details

Shape Chamfer Distance (SCD) [3] provides a direct measure of the overall chamfer distance between predicted and ground truth shapes. Using notations introduced in Secs. 3.4 and 3.5, SCD is defined as:

$$\text{SCD} = \text{CD}\left(\bigcup_{i=1}^{N}(\hat{R}_{[i]}\mathcal{P}_i + \hat{t}_{[i]}), \bigcup_{i=1}^{N}(R_i\mathcal{P}_i + t_i)\right), \quad (11)$$

where $\cup_{i=1}^{N}$ indicates the union of N parts to form the assembled shape and $[i]$ denotes the index of the matched part corresponding to the i-th part under optimal matching $\mathcal{M}$. SCD is scaled by a factor of 10^3 for better practical interpretation.

Part Accuracy (PA) [3] assesses the correctness for individual part poses. A part pose is considered as accurate if its chamfer distance is below a specific threshold $\epsilon = 0.01$:

$$\text{PA} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{1}\left(\text{CD}(\hat{R}_{[i]}\mathcal{P}_i + \hat{t}_{[i]}, R_i\mathcal{P}_i + t_i) < \epsilon\right), \quad (12)$$

where $\mathbb{1}$ is an indicator function, which returns 1 if the condition is met and 0 otherwise.

Success Rate (SR) [4] is 1 if all parts in a shape are considered as accurate, and 0 otherwise:

$$\text{SR} = \mathbb{1}(\text{PA} = 1). \quad (13)$$

This metric measures whether an entire assembled shape is correctly predicted, making it a stringent criterion for evaluating complete assembly success.

Kendall-Tau (KT) [7] measures the ordinal correlation between the ground truth permutation $\sigma \in \mathbb{N}^N$ and the predicted permutation $\hat{\sigma} \in \mathbb{N}^N$. Formally, KT is defined as:

$$\text{KT} = \frac{c^+(\hat{\sigma}, \sigma) - c^-(\hat{\sigma}, \sigma)}{N(N-1)/2}, \quad (14)$$

where $c^+(\hat{\sigma}, \sigma)$ and $c^-(\hat{\sigma}, \sigma)$ denote the number of correctly (concordant) and incorrectly (discordant) ordered pairs in the sequence, respectively. The KT metric ranges from -1 to 1, where 1 indicates a perfect match, -1 indicates a completely reversed sequence, and 0 represents a random ordering.

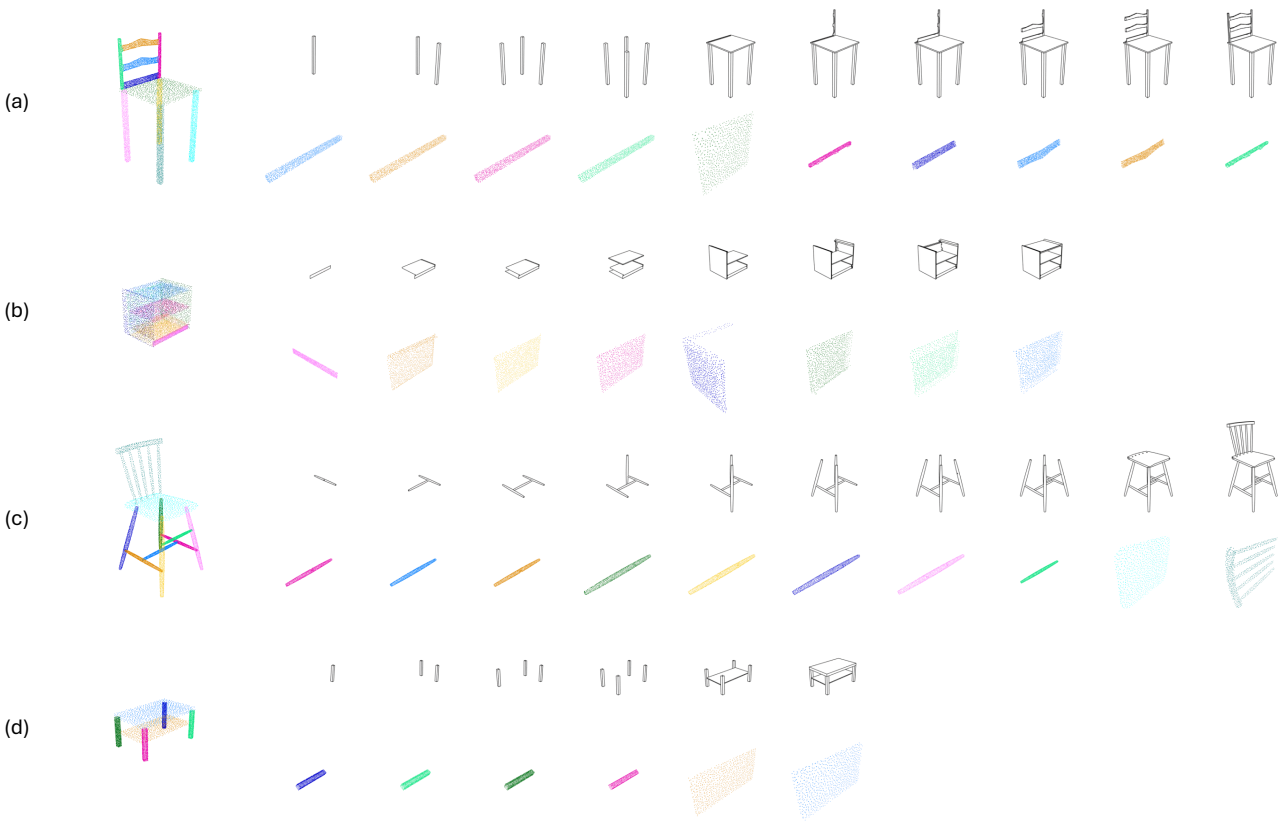

Figure 6. Illustration of several examples from the datasets. (a) and (b) are from the PartNet dataset, while (c) and (d) are from the IKEA-Manual dataset. (a) and (c) are examples of chairs, and (b) and (d) are examples of tables. For each example, the leftmost column shows the fully assembled shape in their point cloud format, the top row on the right presents the step-by-step assembly instruction manual we generated, and the bottom row displays the point cloud of the newly added parts for each step.

C. Dataset Creation Details

We reuse the dataset provided by Li et al. [3] for PartNet and apply the same preprocessing pipeline for IKEA-Manual. Specifically, for each part, we first randomly select 10,000 vertices from its mesh model, followed by sampling 1,000 points using Farthest Point Sampling (FPS). All point clouds are normalized to be centered at the world origin, adopting a canonical coordinate system derived via Principal Component Analysis (PCA) [6]. The longest diagonal of their Axis-Aligned Bounding Box (AABB) is scaled to unit length, eliminating the effects of scale variations across furniture. Additionally, to group geometrically similar parts, we adopt a robust heuristic based on AABB diagonal lengths. Parts with similar diagonal lengths are classified into the same group, regardless of their original pose or orientation. This grouping accounts for symmetries and ensures that similar components, such as table legs or chair arms, are treated uniformly during the assembly process.

As shown in Fig. 6, for both the PartNet and IKEA-Manual datasets, we generate step-by-step furniture assembly manuals to simulate real-world instructional guides. Using Blender's [1, 2] Freestyle functionality, we render 2D line drawing diagrams. First, the fully assembled furniture is placed at the world origin, with the camera positioned to provide a clear frontal view. For each subsequent assembly step, one part is removed, and the scene is re-rendered. Freestyle's edge-enhancement capabilities ensure that the diagrams highlight the part edges effectively, resembling traditional technical manuals. The assembly order is determined using two complementary criteria: (1) parts grouped by AABB diagonal lengths are ordered from bottom to top along the z-axis, reflecting a natural bottom-up assembly process, and (2) within each group, parts are ordered by their distance from the camera, starting with the farthest. This ensures visibility and interpretability of the remaining parts in the diagram.

D. More Experiment Results

D.1. Details of Multi-Step and Multi-View.

To extend our framework to more realistic assembly scenarios, we implement two modifications: Multi-Part, which allows multiple parts to be added in a single step, and Multi-View, which introduces viewpoint variations across steps. Both modifications are performed by fine-tuning our previously trained model, as it already possesses the foundational ability for manual-guided 3D part assembly.

Multi-Step. Motivated by the observation that real-world instruction manuals frequently introduce multiple parts simultaneously, we extend our framework to handle Multi-Part scenarios. This allows us to assess our model's capability in more complex and realistic assembly tasks. Specifically, given an initial set of N parts, we group these parts into M groups (where $M \leq$) based on geometric similarity computed using Chamfer Distance. Each group of similar parts is then associated with a single step diagram. Consequently, we adjust the positional encodings according to these M groups rather than individual parts. Importantly, this grouping simplifies positional encoding assignments, and the model's input is updated accordingly to reflect these M groups. The similarity matrix used within our framework thus becomes an $M \times M$ square matrix

Multi-View Real-world manuals frequently present steps from varying viewpoints to avoid occlusions and clearly illustrate part connections. To reflect this practical scenario, we define a set of eight predefined viewpoints, corresponding to the vertices of a 3D cube surrounding the object. During dataset preparation, each step diagram's viewpoint is randomly selected from these eight viewpoints, introducing realistic visual variation across assembly steps. To facilitate viewpoint alignment, we initialize each instruction manual with an additional blank diagram as the first step, ensuring consistent reference for subsequent viewpoint encoding. Moreover, to maintain canonical space across training, we enforce that the final step diagram always uses a fixed viewpoint (denoted as View 0). In our model, consecutive step diagrams, including the blank initial step, are individually processed through the vision encoder, yielding feature representations f_t and f_{t+1}. These features are then concatenated into $f' = [f_t; f_{t+1}]$, allowing the model to effectively capture the relationships and viewpoint changes between adjacent steps, thereby enhancing robustness to varying visual perspectives.

D.2. Number of Parts.

We analyze the assembly quality across varying numbers of parts on PartNet chair test split, as shown in Fig. 7. It is generally accepted that, assembling objects with more parts is more challenging due to the combinatorial explosive issue. However, our method consistently achieves a higher success rate across all part counts, demonstrating robust adaptability to different levels of assembly complexity. Notably, for shapes with more than 10 parts, 3DHPA and Image-PA exhibit near-zero success rates, whereas our method, Manual-PA, continues to produce competitive assembly results.

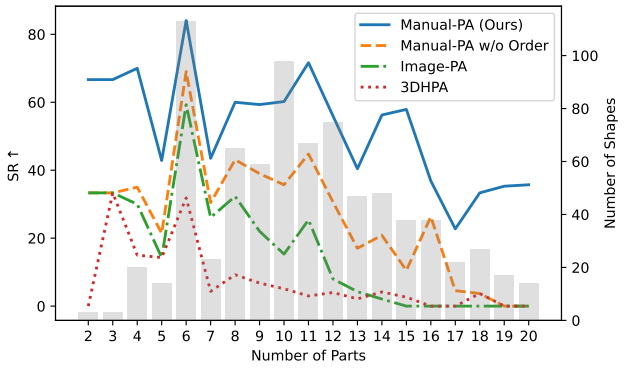

Figure 7. Comparison of average success rates (SR) across varying numbers of parts for different methods on PartNet chair test split. Number of chairs tested shown in background bar chart.

Table 4. 3D part assembly results on the PartNet chair test split. We train and test the model with ground truth order.

Exp.	SCD↓	PA↑	SR↑
Manual-PA (Ours)	**1.7**	**95.38**	**73.07**
w/o RoPE	1.8	94.80	69.53
w/ 3 Decoder Layers	2.8	86.63	42.73

D.3. More Ablation Studies

As shown in Tab. 4, we conduct two additional ablation studies. In the first study, we incorporate RoPE (Rotary Position Embedding) [8] into the attention mechanism. We observe that RoPE does not conflict with the pre-existing positional encoding (PE) in the features and that its inclusion further improves the model's performance. In the second study, we examine the impact of the number of transformer decoder layers. In our default model, the decoder consists of six layers. When we reduce the number of layers to three, the performance drops significantly, particularly in the SR metric, which decreases by 27 percentage points. These results highlight the importance of using more decoder layers for the 3D part assembly task.

D.4. Kendall Tau vs. Performance

As shown in Fig. 8, we conduct experiments to investigate the impact of part order on the performance of the 3D part assembly model. Starting with a model trained using the ground truth order, we introduce varying levels of Gumbel noise to the permutation matrix to randomly perturb the order and then perform inference for 3D part

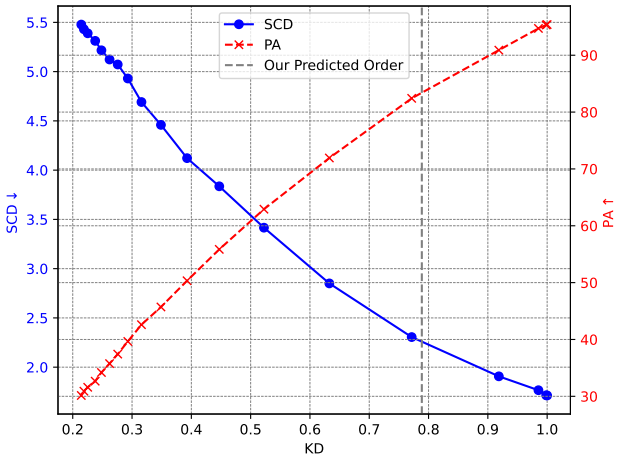

Figure 8. The impact of different part orders on the performance of the 3D part assembly model evaluated on PartNet chair test split. As the Kendall Tau (KD) approaches 1, the order of the parts becomes increasingly correlated with the order specified in the manual, whereas lower KD values indicate less correlation.

assembly. The results reveal that assembly performance improves as Kendall Tau (KD) approaches 1, indicating a stronger correlation between the perturbed and ground truth orders. Conversely, lower KD values lead to poorer performance, which aligns with intuition: incorrect correspondences make it difficult for the model to identify the correct step diagram for each part, thereby hindering pose prediction. The results also highlight the performance upper bound of our method when KD $= 1$. Interestingly, the order learned through permutation learning achieves a KD of approximately 0.79, outperforming randomly perturbed orders with similar KD values. This advantage stems from the fact that our method's randomness primarily arises from the indeterminacy of part order within geometrically equivalent groups, while maintaining relatively accurate alignment across groups.

D.5. Cross Attention Map on Step Diagrams.

As shown in Fig. 9, we visualize the cross-attention maps between each part and the step diagrams. High attention values correspond to regions in the step diagram where the model pays more attention, which aligns with the spatial placement of parts during assembly. For instance, when the first blue board focuses on the back of the chair, it is positioned on the chair back, and when it attends to the area under the seat, it acts as a support between the legs. Comparing the two methods, we observe that without order as a soft guidance (represented by "Manual-PA w/o Order"), the attention regions are more dispersed, resulting in less accurate part placement. For example, the third cyan board does not focus entirely on the left chair leg without soft guidance,

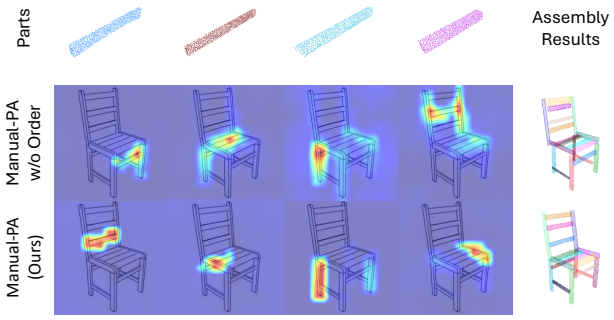

Figure 9. Visualization of cross-attention maps on step diagrams. The cross-attention represents the mean aggregation for each part across all step diagrams. Color red indicates a higher attention.

leading to a misaligned leg position. The absence of order as explicit guidance also results in incorrect part placement. For instance, the fourth purple board, which should be part of the seat, is instead assembled onto the chair back.

E. More Qualitative Results

E.1. More Comparisons

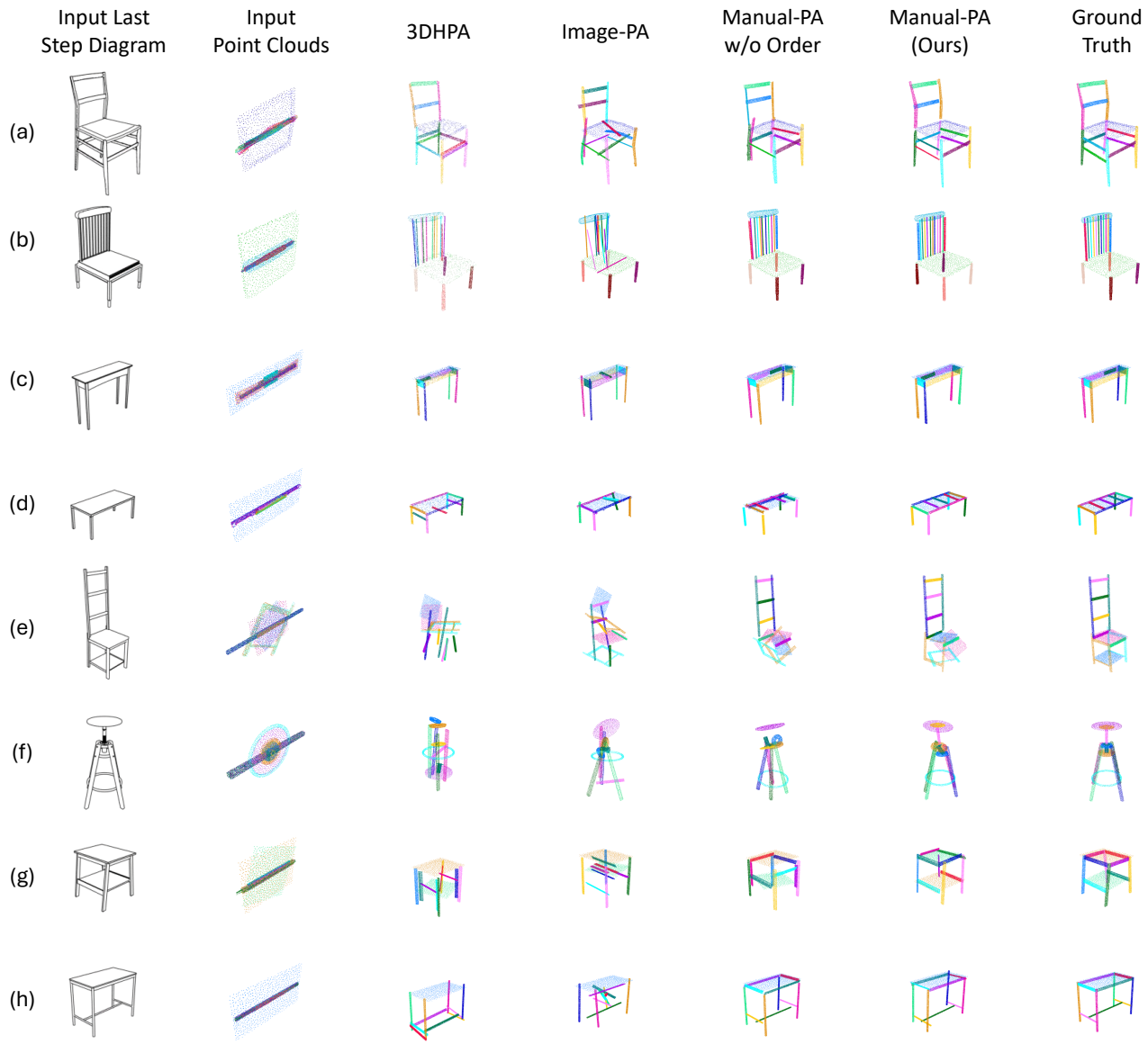

Figure 10. Qualitative comparison of various 3D part assembly methods. Eight examples are shown: chair (a), (b) and table (c), (d) from the PartNet dataset, and chair (e), (f) and table (g), (h) from the IKEA-Manual dataset.

E.2. More Visualizations

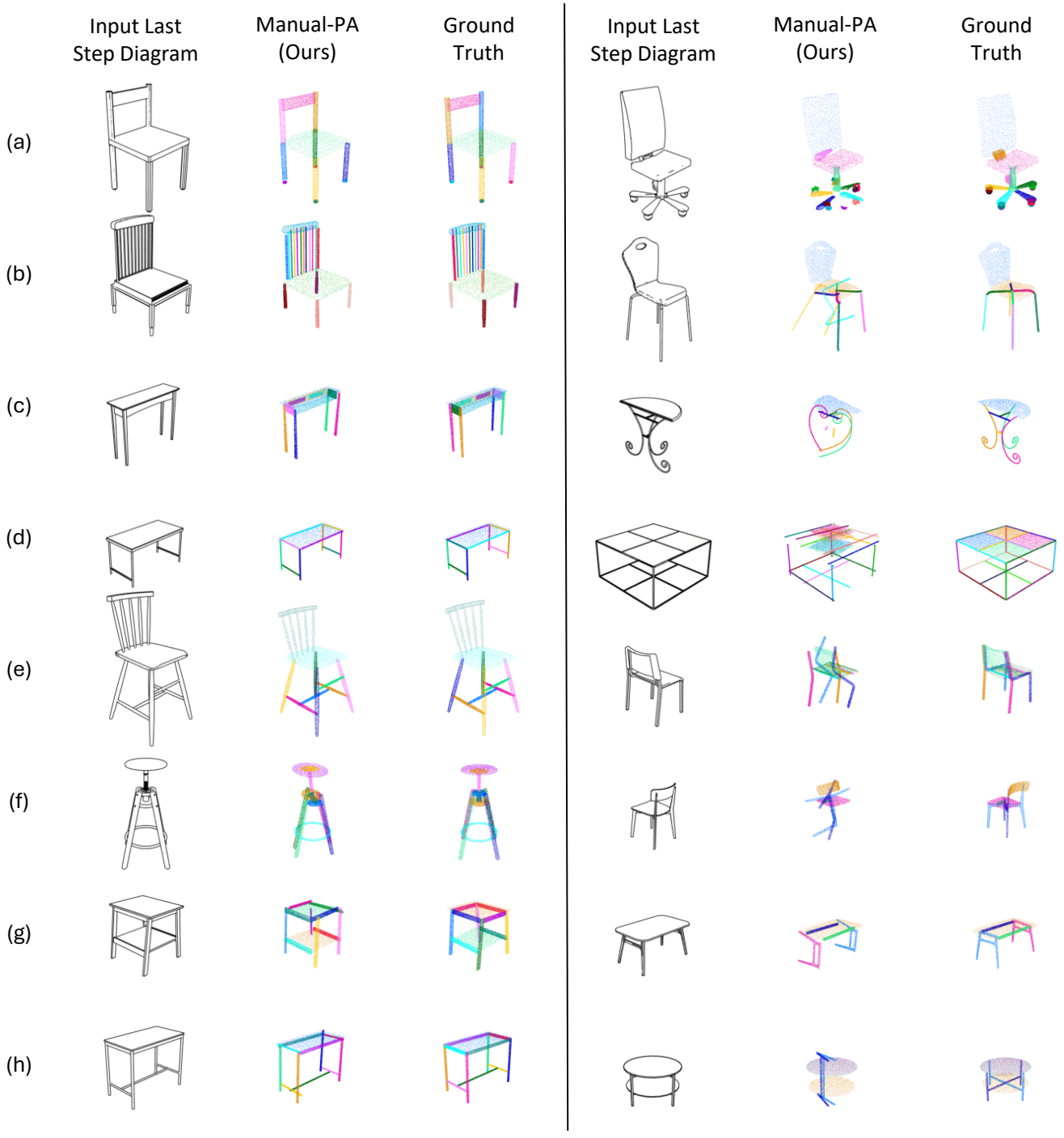

Figure 11. Results visualization of our method Manual-PA. The left column showcases examples where the method performs well, while the right column illustrates cases with less satisfactory outcomes. Eight examples are included: chairs (a), (b) and tables (c), (d) from the PartNet dataset, and chairs (e), (f) and tables (g), (h) from the IKEA-Manual dataset..

E.3. Demonstration Video

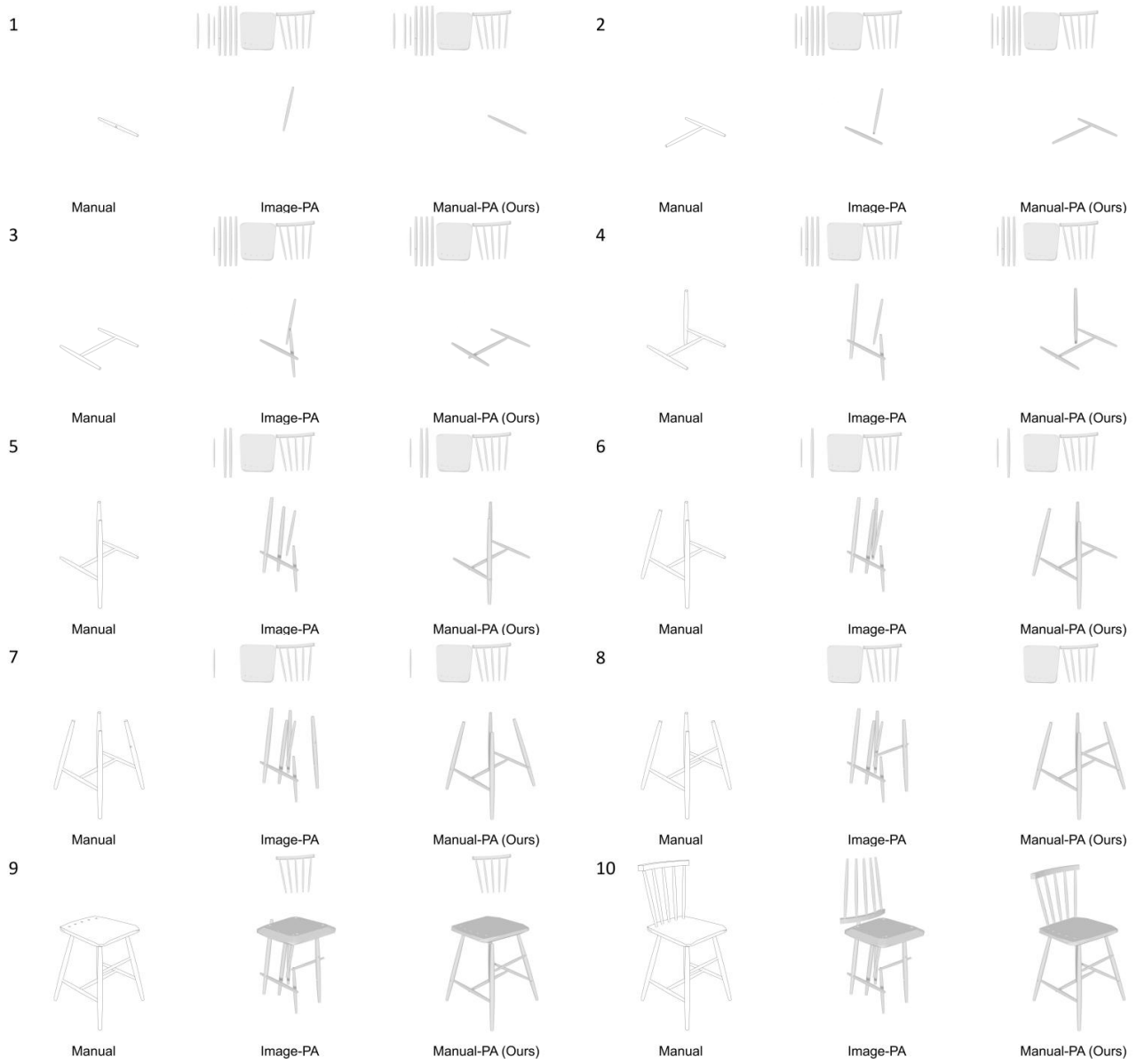

Figure 12. Snapshots from the demonstration video of the assembly process. For each frame, the leftmost column displays the step diagram from the manual, the middle column shows the assembly result of Image-PA, and the rightmost column presents the assembly result of our Manual-PA method. End frames of each step are selected for illustration. The video is provided as a part of supplementary material.

References

[1] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2024. 2

[2] Maximilian Denninger, Dominik Winkelbauer, Martin Sundermeyer, Wout Boerdijk, Markus Knauer, Klaus H. Strobl, Matthias Humt, and Rudolph Triebel. Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901, 2023. 2

[3] Yichen Li, Kaichun Mo, Lin Shao, Minhyuk Sung, and Leonidas Guibas. Learning 3d part assembly from a single image. In *ECCV*, pages 664–682. Springer, 2020. 1, 2

[4] Yulong Li, Andy Zeng, and Shuran Song. Rearrangement planning for general part assembly. In *7th Annual Conference on Robot Learning*, 2023. 1

[5] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 1

[6] Karl Pearson. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin philosophical magazine and journal of science*, 2(11):559–572, 1901. 2

[7] Rodrigo Santa Cruz, Basura Fernando, Anoop Cherian, and Stephen Gould. Deeppermnet: Visual permutation learning. In *CVPR*, pages 3949–3957, 2017. 1

[8] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. 3