

Completing 3D Partial Assemblies with View-Consistent 2D-3D Correspondence

Weihaio Wang¹ Yu Lan¹ Mingyu You^{1,2*} Bin He^{1,2}

¹College of Electronic and Information Engineering, Tongji University, Shanghai, China

²State Key Laboratory of Autonomous Intelligent Unmanned Systems, Shanghai, China

myyou@tongji.edu.cn

Abstract

3D assembly completion represents a fundamental task in 3D computer vision and robotics. This task aims to retrieve the missing parts from a set of candidates and predict their 6-DoF poses to make the partial assembly complete. However, due to the inherent uncertainty in completion and the similarity among candidates, even humans struggle to achieve precise completion without external guidance. To address this challenge, we introduce an auxiliary image depicting the complete assembly from a specific view. The primary challenge lies in the lack of correspondence or grounding between the partial assembly and the image, leading to ambiguities in identifying missing parts and ineffective guidance for completion. Moreover, this correspondence heavily depends on the view of image, which, unfortunately, is often unknown in real-world scenarios. To this end, we propose a novel cross-modal 3D assembly completion framework. At its core is missing-oriented feature fusion augmented by self-supervised view alignment to establish view-consistent 2D-3D correspondence between the image and the partial assembly, which effectively captures clues of missing parts from the image and provides targeted guidance for completion. Extensive experiments demonstrate our state-of-the-art performance on the PartNet dataset and show its generalization capabilities in two downstream applications: component suggestion and furniture restoration.

1. Introduction

3D assembly completion is a highly sought-after functionality in various downstream tasks, such as suggesting suitable components for half-designed furniture in computer-aided assembly design, and making virtual repair plans for broken furniture that is missing several parts. Given the point cloud of a partial assembly, the goal of this task is to identify and retrieve its missing parts from a set of candidates, then predict their 6-DoF poses to make the partial assembly complete. Distinct from the final steps in general assembly, where the

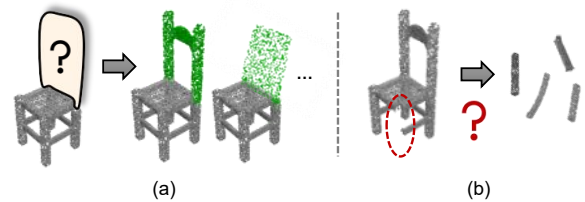


Figure 1. Challenges of 3D assembly completion. (a) Uncertainty of completion. (b) Resemblance of candidates.

remaining candidate parts are exactly the missing ones, 3D assembly completion requires more elaborate understanding of the structural completeness of partial assembly to infer the true missing parts. Even for humans, to achieve precise completion without any external guidance is no easy task. The challenges are summarized as follows.

First, the uncertainty of completion. The completion result of a partial assembly is not unique and usually encompasses a large solution space. Consider, for example, the chair with an absent back depicted in Fig. 1(a). There can be multiple completion plans with different combinations of parts to construct the back. Existing methods [19, 24, 25] employ sophisticated designs to memorize the fixed mappings between partial assemblies and their missing parts, and lack the ability to infer the desirable completion plan from external guidance, e.g., target images. Such rigid memorization significantly hampers the ability to generalize to novel assemblies. When confronted with an unseen partial assembly, these methods struggle to accurately predict the optimal completion, hindered by their reliance on constrained experiences derived from memorizing a limited number of training cases.

Second, the resemblance of candidates. Some candidate parts, such as the legs illustrated in Fig. 1(b), demonstrate a high degree of geometric similarity. Previous works in assembly alleviate this issue by assuming the availability of part-level instance segmentation for partial assemblies, which greatly assists in discriminating similar parts. With part instance segmentation, it becomes straightforward to identify the missing leg in Fig. 1(b) by comparing the candidate parts with the other three legs of the partial assembly, which share the same geometry. Unfortunately, partial as-

*Corresponding author.

semblies in real-world applications are often scanned point clouds without part-level segmentation. Even with the state-of-the-art method of 3D part instance segmentation [18], the average precision is only 56% for chairs and 43% for tables. Such levels of precision are inadequate for reliably guiding the accurate retrieval of missing parts.

In this paper, we seek a solution by introducing a single-view image, termed image-guided assembly completion. This is motivated by the fact that humans often refer to an auxiliary image for completion guidance. In practice, the image can be either a product rendering or a realistic photo of the assembly. As a completion target, this image depicts the expected part composition and structure of the assembly from a specific view, mitigating the uncertainty of completion. Moreover, it provides potential clues for identifying the missing parts from a visual perspective, alleviating the confusion of parts caused by lack of part segmentation.

Despite its advantages, guiding 3D assembly completion using a single-view image is a non-trivial task. The primary challenge lies in establishing the correct correspondence between the partial assembly and the image. In this context, correspondence can be seen as a process of visual grounding, where the goal is to identify which areas in the image correspond to the partial assembly and which represent the missing parts. Moreover, this correspondence largely depends on the view of image. Even the same part can present distinct features when observed from different views. Existing works in related fields, such as view-guided shape completion [1, 36], address this challenge by providing the image view and pre-aligning the partial point cloud accordingly. However, this approach imposes significant restrictions on the dataset, requiring additional annotations of image views and increasing the cost of data collection. Meanwhile, its applicability is limited in real-world scenarios, where the image view is often unknown. One may raise a potential solution by image-to-point cloud (I2P) registration. However, even with I2P registration [21], the average correct correspondence proportion among all I2P pairs is only 47%, and this rate tends to be even lower for partial assemblies.

Considering these limitations, we aim to solve a more practical and challenging setting of 3D assembly completion, where the partial assembly is unsegmented and the view of image is undetermined. To this end, we introduce a novel image-guided 3D assembly completion framework with view-consistent 2D-3D correspondence. As the core of the framework, we design missing-oriented feature fusion to establish the correspondence between the 2D image and the 3D partial assembly, which precisely captures the regions of missing parts from the image and efficiently utilizes these clues to determine the missing parts. An auxiliary task of view alignment is proposed to constrain the view consistency of this correspondence and ensure efficient cross-

modal guidance. Leveraging these designs, our method not only achieves state-of-the-art completion results on the PartNet dataset, but also demonstrates strong generalization to real-world applications, including component suggestion for assembly design and restoration of real-world IKEA furniture with realistic photos.

In summary, our main contributions are as following.

- Present an image-guided solution towards a more practical assembly completion scenario, eliminating the necessity for unrealistic assumptions such as pre-segmented partial assembly or pre-aligned image view.
- Design a novel view-consistent cross-modal learning framework for 3D assembly completion, incorporating self-supervised view alignment and missing-oriented feature fusion, both optimized jointly in an end-to-end manner.
- Achieve state-of-the-art in 3D assembly completion, and demonstrate flexible applicability across two real-world applications.

2. Related Work

2.1. Assembly-based Shape Modeling

Automatic assembly is a desirable ability of the intelligent robot to assist with various assembly works. Using PartNet [14], a large-scale dataset of 3D shapes with part-level annotation, recent works have explored a series of assembly tasks. Zhan et al. [34] first define the task of 3D part assembly that aims to predict 6-DoF poses for a given set of semantic parts and assemble them into a whole. After that, different attempts are explored with more efficient modeling, such as sequential models [7, 26], attention mechanisms [5, 29, 35], diffusion models [3]. Similarly, Chen et al. [2] propose shape neural matting that focuses on pairwise assembly of parts without semantic information. Li et al. [11] leverage a single image to guide assembly with a two-stage design, which requires extra 2D part segmentation for intermediate supervision. MEPNet [22] aims to assemble LEGO bricks according to step-by-step manual images rather than a single image. There are also emerging works that focus on physical and structural constraints of assembly, *e.g.*, connectivity [12].

Completing partial assemblies denotes another essential assembly task. Known as 3D assembly completion, this task is not about the final steps of a complete assembly process. Instead, it requires retrieving suitable parts and predicting their poses to complete the partial assembly. ComplementMe [19] proposes a component suggestion method, which consists of a retrieval network and a placement network. FiT [24] utilizes an encoder-decoder architecture for more elaborate modeling of the relationship between the partial assembly and the candidate parts. PhysFiT [25] further considers

several essential structural constraints in the design, *i.e.*, connectivity, stability and symmetry. Nevertheless, these works are built on the impractical assumption that the partial assembly is annotated with part segmentation. In this work, we explore a more realistic setting of completing partial assemblies without part segmentation and address the challenge by introducing image-guided assembly completion.

2.2. View-guided Point Cloud Completion

Point cloud completion is an essential task in 3D computer vision that aims to recover the complete point cloud from partial ones. The incompleteness can be caused by several reasons, *e.g.*, occlusions and low resolution of the sensor. The missing region is usually irregular without semantic meaning, which entails a different type of incompleteness compared to that in 3D assembly completion. Existing works have extensively explored this issue by learning 3D shape priors, including PCN [33], ShapeFormer [31], AutoSDF [13], SnowflakeNet [28], Seedformer [37], AdaPoinTr [32] and DiffComplete [4], among others.

Recently, Zhang et al. [36] propose to complete partial point clouds with an extra image, namely view-guided point cloud completion (ViPC). ViPC leverages a pre-trained single-view reconstruction network to reconstruct a coarse point cloud from the image and refine it jointly with the partial point cloud. Feature fusion is conducted in 3D space. Instead, XMFnet [1] extracts features for each modality and merges them in the latent space. Both methods rely on the predefined view of image. That is, the perspective of the partial assembly is pre-aligned with that of the completed assembly in the image, ensuring efficient cross-modal fusion. However, the view of image cannot always be assumed to be known in advance in real applications. EGInet [30] and CDPNet [6] alleviate this issue by more efficient multi-modal fusion designs. In this work, we address this issue by designing a novel view-consistent cross-modal learning framework.

3. Method

3.1. Overview

Initially, we provide a formal definition of the image-guided assembly completion task. Given a partial assembly A and a reference image I of the target assembly, this task aims to retrieve the missing parts $M = \{p_1, \dots, p_k\}$ from a set of candidate parts $C = \{p_1, \dots, p_N\}$ and predict their 6-DoF poses $\{(r_1, t_1), \dots, (r_k, t_k)\}$ to make the assembly complete. Here, r denotes the quaternion-based rotation and t denotes the translation. Assemblies and parts are both represented by point clouds. Importantly, our method requires neither part segmentation on the partial assembly nor its relative view from the image, presenting a more realistic yet challenging scenario.

The pipeline of our framework is illustrated in Fig. 2(a), which consists of two core modules. (1) View alignment determines the relative view between the image and partial assembly through a view predictor, which is optimized by a self-supervised projection loss between the projection of the completed assembly and the silhouette of the image. (2) The missing-oriented feature fusion first aligns (or rotates) the partial assembly according to the relative view, and then extracts a view-consistent feature for the missing parts through differential attention. This feature is further fused with that of candidate parts for the final prediction of the missing parts. The partial assembly is completed in an autoregressive manner, with view alignment jointly optimized along this process. Details are explained as follows.

3.2. Self-supervised View Alignment

When leveraging a single-view image to complete a partial assembly, we must first ground the partial assembly in the image from its specific view, in order to identify which parts already exist and which are missing. To achieve this, we employ a ResNet-based [8] view predictor to estimate the view of the partial assembly v in relation to the image, which consists of a quaternion-based rotation $r \in \mathbb{R}^4$ and a scaling c . This view is used to rotate the partial point cloud and align it with the image view, which facilitates the discovery of missing parts in the image and ensures the view consistency in 2D-3D feature fusion. In the following sections, we denote this view transformation as T_v .

During optimization, we apply T_v to the final completed assembly \hat{A} and use a differential renderer R [9] to compute its differential projection $R(T_v(\hat{A}))$. This projection is supervised by the silhouette of the image S_I through the calculation of a self-supervised projection loss:

$$\mathcal{L}_{proj} = \lambda_{proj} \|S_I - R(T_v(\hat{A}))\|_1. \quad (1)$$

3.3. Missing-oriented Feature Fusion

In our setting, the image depicts the anticipated part composition and structure of the target assembly from a specific view. In view alignment, we have grounded the partial assembly with the image, *i.e.*, aligned it with the image view. Next, we design a missing-oriented feature fusion with an encoder-decoder architecture to establish the correspondence between the image and partial assembly. The cross-modal encoder efficiently extracts potential key information of the missing parts. The part decoder fuses this information with the candidate parts for final prediction of the missing parts.

Cross-modal encoder. Given a partial point clouds and an image, we initially utilize distinct backbones to encode each modality into a latent feature space. Specifically, the image is divided into p_1 patches and embedded into $F_I \in \mathbb{R}^{p_1 \times 256}$ using a multilayer perceptron (MLP). Similarly, we employ

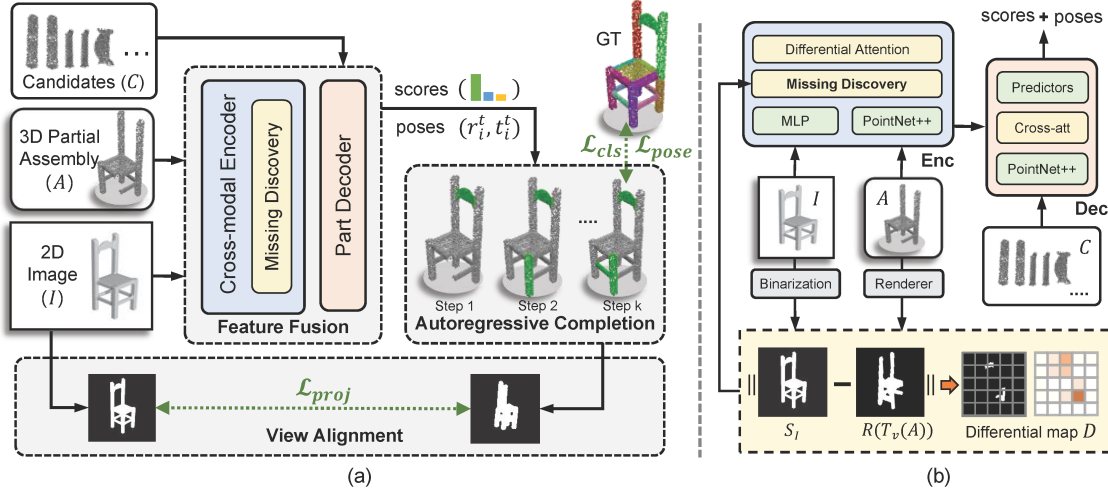


Figure 2. (a) Illustration of our framework. Missing-oriented feature fusion learns the correspondence between the 3D partial assembly A and the 2D target image I . View alignment ensures the view consistency of this correspondence for efficient fusion. The assembly is completed autoregressively by part retrieval and pose prediction from a set of candidates C . (b) The encoder-decoder architecture in feature fusion. We visualize an example to illustrate the process of missing discovery with differential attention.

PointNet++ [16] to derive patch features from the partial point clouds. This involves selecting p_2 points as centers via farthest point sampling, followed by the allocation of neighboring points to each center via ball query. These point patches are hierarchically embedded into patch features $F_A \in \mathbb{R}^{p_2 \times 256}$ via a two-layer set abstraction.

Then we compute the attention mechanism between F_I and F_A to establish their correspondence. We first employ self-attention [20] to capture the relationships among patch features within the partial assembly. The updated feature F_A interacts with F_I via the differential attention, which is formally defined in Eq. (2):

$$\text{diff-att} = \text{Softmax} \left(\frac{\mathbf{Q}\mathbf{K}^T \oplus \mathbf{D}}{\sqrt{d_k}} \right) \mathbf{V}, \quad (2)$$

$$\mathbf{Q} = [F_A, \text{pos}_A]W_Q, \mathbf{K}/\mathbf{V} = [F_I, \text{pos}_I]W_{K/V}.$$

\mathbf{Q} is derived from F_A and \mathbf{K}/\mathbf{V} from F_I using different weight matrices, concatenated with position encoding pos_A and pos_I , respectively. Specifically, we employ an absolute position encoding for image features F_I and the center of the point patch for the features of partial assembly F_A .

When establishing their correspondence, the regions of missing parts in the image offer crucial guidance for the completion. Therefore, we design a missing discovery module that focuses on identifying and highlighting these missing regions. This is achieved by computing a differential attention map D as defined in Eq. (3):

$$D = \|S_I - R(T_v(A))\|. \quad (3)$$

Here, S_I denotes the silhouette of the image, and $R(T_v(A))$ denotes the rendering of partial assembly. T_v denotes the view transformation from view alignment. This differential attention map is utilized to modulate the attention weights through a counterpoint addition operation denoted as \oplus in Eq. (2), which directs the cross-modal fusion module to concentrate on the precise regions of missing parts, rather than the entire image. An illustration of this differential attention map is provided in Fig. 2(b). As depicted, the regions corresponding to missing parts exhibit larger weights in the map (highlighted in orange). Its effectiveness is demonstrated in the ablation study.

Part decoder. The merged feature $F_{fuse} \in \mathbb{R}^{p_2 \times 256}$ obtained from the cross-modal encoder encodes potential information of the missing parts. The part decoder (Dec) conditions on this merged feature to predict the missing parts, as illustrated in Fig. 2(b). We utilize PointNet++ to extract patch features $F_C \in \mathbb{R}^{N \times p_3 \times 256}$ for each candidate part. Here, N denotes the number of candidates and p_3 denotes the number of patches for each candidate. These features, contained in F_C , are employed as queries to compute cross-attention with F_{fuse} . Subsequently, the patch features for each candidate part are concatenated, resulting in per-part features denoted as $F_{agg} \in \mathbb{R}^{N \times (p_3 \times 256)}$. Finally, we use two predictors to predict binary classification scores and 6-DoF poses for the candidate parts. The model autoregressively retrieves and assembles the missing parts based on the scores and poses, which is further elucidated in the subsequent section.

3.4. Autoregressive Learning

The framework is designed in an autoregressive completion manner, as outlined in Algorithm 1 (See supplementary ma-

terial). To initiate the process, we predict the view of the image using the view predictor and extract features for both the partial assembly and the image. In each step, the framework retrieves a candidate part p_i with the highest score, designating it as the missing part. Subsequently, it assembles the selected part $T_i(p_i)$ with the partial assembly, where T_i represents the joint transformation of pose (r_i, t_i) . Then we add $T_i(p_i)$ to the partial assembly and remove this part from the list of candidate parts for the next step. The algorithm runs for k iterations where k denotes the number of missing parts.

Learning of the framework encompasses multiple tasks, including candidate classification, pose prediction, and view alignment. These tasks are supervised by a combination of loss functions:

$$\begin{aligned} \mathcal{L} &= \lambda_{cls} \mathcal{L}_{cls} + \mathcal{L}_{pose} + \lambda_{proj} \mathcal{L}_{proj}, \\ \mathcal{L}_{pose} &= \lambda_t \mathcal{L}_t + \lambda_r \mathcal{L}_r + \lambda_{scd} \mathcal{L}_{scd} + \lambda_{sym} \mathcal{L}_{sym}. \end{aligned} \quad (4)$$

We supervise the predicted classification scores of candidate parts y_j^i with the Cross-Entropy (CE) loss:

$$\mathcal{L}_{cls} = \frac{1}{kN} \sum_{i=1}^k \sum_{j=1}^N CE(y_j^{i*}, y_j^i). \quad (5)$$

y_j^{i*} denotes the ground-truth label of part j in the i -th iteration, where the true missing parts are assigned with 1s and otherwise 0s. The predicted poses (r_i, t_i) are supervised with the translation loss and rotation loss:

$$\begin{aligned} \mathcal{L}_t &= \frac{1}{k} \sum_{i=1}^k \|t_i - t_i^*\|_2^2, \\ \mathcal{L}_r &= \frac{1}{k} \sum_{i=1}^k d_c(r_i(p_i), r_i^*(p_i)). \end{aligned} \quad (6)$$

To ensure the overall geometric consistency and structural symmetry of completed assembly, we employ the shape Chamfer distance loss and symmetric loss [24] between the completed assembly \mathcal{S} and the ground-truth assembly \mathcal{S}^* :

$$\mathcal{L}_{scd} = d_c(\mathcal{S}, \mathcal{S}^*), \mathcal{L}_{sym} = d_c(\Phi(\mathcal{S}), \mathcal{S}^*), \quad (7)$$

where Φ denotes the mirror transformation.

4. Experiments

In this section, we present experimental settings and provide a comprehensive evaluation of the proposed method through comparisons with state-of-the-art methods and an ablation study of our core designs.

4.1. Experimental Settings

Dataset. We conduct experiments on PartNet [14], which is a large-scale dataset of 3D shapes annotated with part-level segmentation. We choose chair, table, lamp and storage furniture, the major categories of furniture. Since part of

PartNet assemblies lack textures, we use PyTorch3D [17] to render 8 texture-less images by rotating the assembly at 45-degree intervals in the yaw angle while keeping the pitch and roll angles fixed, covering most common views. The images have a resolution of 128×128 . During training, we randomly select an image from the 8 renderings as input.

Implementation details. The framework is implemented with Pytorch [15] and optimized with the AdamW optimizer for 500 epochs on 4 Nvidia V100 GPUs with a learning rate of 1.5×10^{-4} . The batch size is set to 64. We randomly sample k parts as the missing parts and the remaining as the partial assembly. The candidate parts are set with $N = 20$, containing k ground-truth missing parts and $N - k$ disruptive parts sampled from other assemblies in the same batch. We defined three task modes: easy, medium, and hard, corresponding to different values of k ($k = 1$ for easy, $k = 3$ for medium, and $k = 5$ for hard). To reduce computational complexity, we resample the partial assembly with 2048 points for differential rendering. The projection of assembly has the same resolution as the image. The layers of attention module are set to 3 as default. The patch size of image/partial assembly/candidate part is set to 64/64/8.

4.2. Metrics

We adopt three metrics from previous assembly works [24, 34] to evaluate the quality of part retrieval and pose prediction. Match Accuracy (MA) measures the proportion of missing parts that are correctly retrieved from the candidates. Completion Chamfer Distance (CCD) is defined as the overall Chamfer distance between the predicted parts and the ground-truth missing parts. A lower distance indicates a higher geometrical consistency. Part Accuracy (PA) denotes the proportion of retrieved parts within a Chamfer distance of 0.01 compared with the ground truths. Definition can be found in supplementary material.

4.3. Main Results

We compare our method with several relevant works that focus on completing partial assemblies, including assembly-based methods (3DPA [11], FiT [24]) and generative methods (AdaPoinTr [32], XMFnet [1], EGIInet [30], TRELIS [27]). The implementation details of these methods and results of TRELIS are reported in supplementary material.

Comparison results. The quantitative results, summarized in Tab. 1 and Tab. 2, demonstrate that our method outperforms all competitors across most settings. The advantages of our method become increasingly evident as the difficulty level rises from easy to hard. 3DPA is unable to capture the accurate correspondence between parts of 3D partial assembly and the image, and exhibits poor performance consequently, especially in hard mode. FiT heavily relies on accurate part segmentation to generate instance encoding,

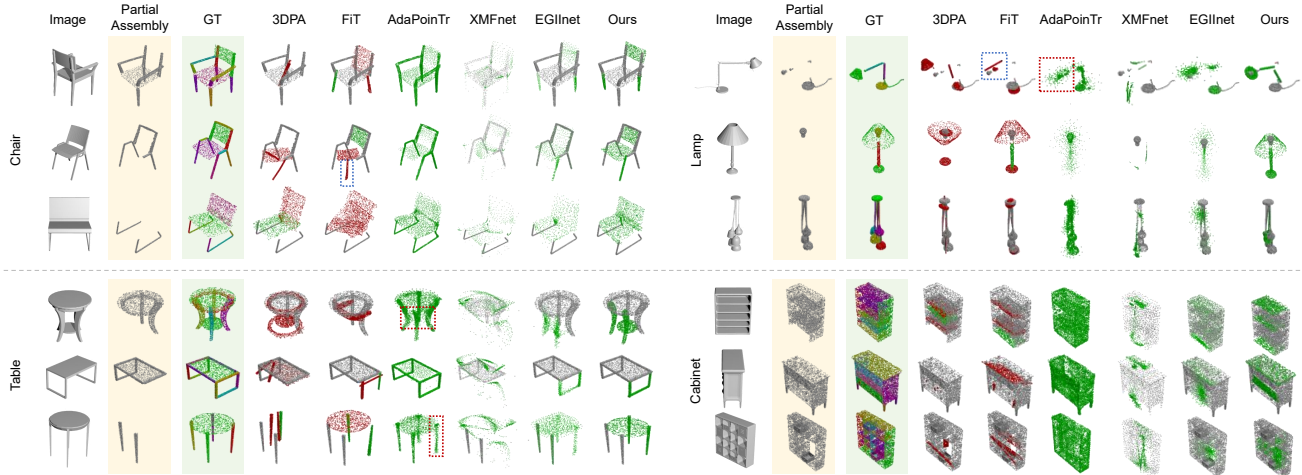


Figure 3. Visualization of completion results with a medium difficulty ($k = 3$). *GT* denotes the ground-truth assembly annotated with part segmentation. In completion results, color gray/green/red denote the partial assembly/correct predictions/wrong predictions. The generated points of AdaPoinTr, XMFnet and EGIInet are visualized in green. Zoom in for better visualization.

Mode	Method	PA(%) \uparrow				MA(%) \uparrow				CCD(10^{-2}) \downarrow			
		Chair	Table	Lamp	Cabinet	Chair	Table	Lamp	Cabinet	Chair	Table	Lamp	Cabinet
Easy	3DPA	19.89	28.21	20.56	14.49	51.68	59.49	64.49	35.67	2.61	1.32	3.67	0.87
	FiT	28.43	36.92	22.89	11.89	49.81	53.85	60.75	26.34	1.56	8.72	2.81	0.91
	Ours	55.27	64.63	28.19	39.61	58.70	69.37	63.05	43.88	0.86	0.55	3.13	0.55
Medium	3DPA	13.81	23.24	9.89	13.75	38.70	45.45	39.06	28.67	3.42	1.80	4.10	0.92
	FiT	20.69	24.18	14.32	11.19	37.66	48.17	40.36	25.87	2.33	1.49	3.89	0.97
	Ours	39.84	37.16	14.52	28.24	49.43	57.69	41.76	39.47	0.99	0.69	2.94	0.59
Hard	3DPA	1.30	2.71	3.64	3.77	20.08	27.02	23.27	19.16	5.54	2.86	6.13	1.45
	FiT	1.34	3.05	6.18	3.22	22.26	24.34	24.36	16.92	5.84	2.89	5.57	1.43
	Ours	23.14	26.56	10.81	14.93	35.64	44.68	36.19	31.01	0.73	0.41	1.93	0.60

Table 1. Quantitative results compared with the assembly-based methods.

which provides strong evidence for missing part retrieval. Without accurate segmentation, FiT presents significant declines in PA and MA. AdaPoinTr yields subpar results in CCD, attributed to its design on modeling the entire assembly without efficient designs for handling missing parts. XMFnet assumes the view of the image to be known and loses efficacy in our setting, where the partial assembly is not aligned with the image view. It fails to capture the precise missing parts of the partial assembly, resulting in worse performance. Compared to different categories of furniture, lamp and cabinet are more difficult. Lamps usually contain irregular parts, such as lampshade and bulb. Cabinets have a more complex structure and contain more geometrically similar parts.

We provide more qualitative results with a medium difficulty in Fig. 3. FiT suffers from inaccurate part retrieval, especially in distinguishing geometrically similar parts, as indicated by the blue dotted boxes. While AdaPoinTr achieves better visual results in completion, it does not effectively resolve the uncertainty of completion, resulting in inconsistent geometries, blurry boundaries and connections as indicated

by the red dotted boxes. XMFnet inaccurately models the distribution of missing parts due to the lack of view alignment, leading to poor generalization across novel partial assemblies. In comparison, our method precisely identifies the missing parts and reduces completion ambiguities with the guidance from image, alleviating the issues of uncertainty and similarity.

Analysis of view. Images under different views present distinct information for assembly completion. If the missing parts are obscured by other parts in a certain view, extracting effective guidance for assembly completion becomes challenging. This results in the differential attention mechanism failing to yield informative cues about the absent parts. We provide additional qualitative results to evaluate the effect of view in Fig. 4(a). The overall performance is robust. Side views are more beneficial for completing partial assemblies, which convey more useful information for completion. We also visualize an example of chair that misses a back, a seat and a leg in Fig. 5. Occlusion (the first view) and overlap (the third view) hinder the identification of the missing leg

Mode	Method	Chair	Table	Lamp	Cabinet
Easy	AdaPoinTr	2.89	2.11	19.04	12.42
	XMFnet	20.99	15.69	19.57	11.99
	EGInet	1.31	8.29	15.51	8.15
	Ours	0.86	0.55	3.13	0.55
Medium	AdaPoinTr	8.85	3.97	8.01	3.28
	XMFnet	8.57	5.48	19.05	2.88
	EGInet	4.87	2.42	6.35	1.98
	Ours	0.99	0.69	2.94	0.59
Hard	AdaPoinTr	4.83	1.92	5.01	1.67
	XMFnet	5.25	3.96	24.43	1.38
	EGInet	3.02	1.37	4.37	1.10
	Ours	0.73	0.41	1.93	0.60

Table 2. CCD (10^{-2}) compared with generative methods.

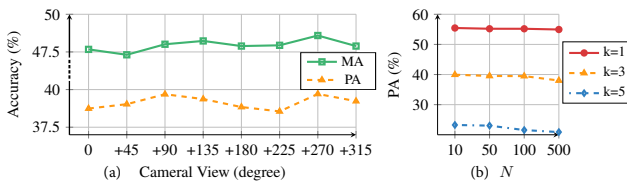


Figure 4. (a) Effect of images from different views. 0 represents the front view and +180 the back. (b) Evaluation of N .

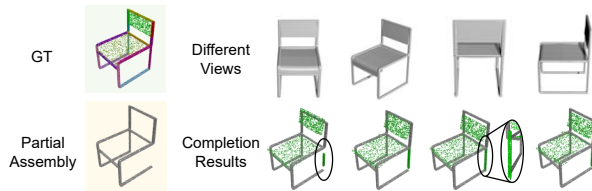


Figure 5. Visualization of an example, completed with images of the same chair from different views.

through differential attention. Consequently, this leads to the retrieval of a leg part that is either disproportionately short or excessively long.

Analysis of N . The number of candidate parts N can be flexibly configured. As N increases, our method exhibits robust performance, shown in Fig. 4(b).

4.4. Applications

Component suggestion. The ability to complete partial assemblies by reusing parts from existing 3D models is a highly sought-after functionality in assembly design, referred to as component suggestion [19]. Our proposed method facilitates image-conditioned component suggestion, enabling the completion of partially designed assemblies into diverse, plausible full designs based on novel images. In this context, we configure the toolkit with $N = 100$, where candidate parts are randomly sampled from a batch of 64 assemblies. Fig. 6 illustrates the assembly completion results for a chair,

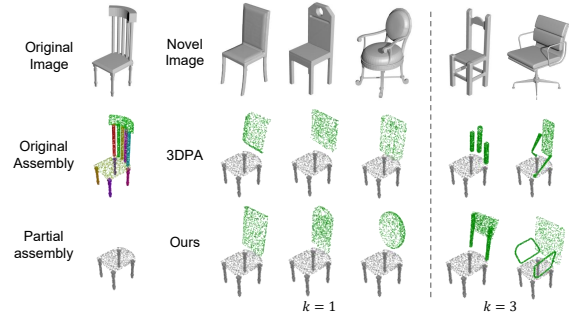


Figure 6. Results of image-guided component suggestion.

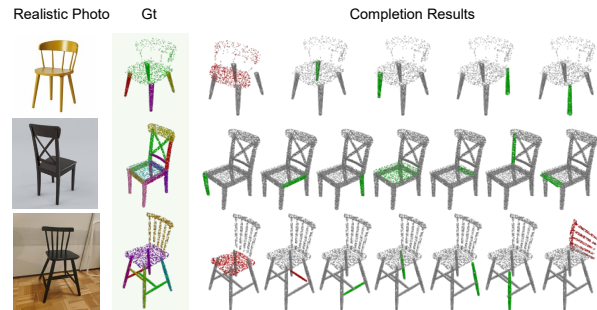


Figure 7. Completion results of chairs from IKEA-Manual. The correctly completed parts are visualized in green, otherwise in red.

where the backrest is in progress, using one part and three parts, respectively. As demonstrated, our method is more adept at understanding style variations in chair backs from novel images, thereby constructing structures that are more consistent with the visual cues in the image.

Furniture restoration. Our method can also be extended to generate feasible repair plans for broken furniture in a virtual environment, using a realistic photo as input. We test our model on the chairs from IKEA-Manual [23], which is collected from real-world IKEA furniture, with dense annotations of 3D part decomposition and assembly plan, aligned with visual manuals. We present three case examples in Fig. 7 (additional cases are provided in supplementary material). In this context, we sample candidate parts from the assembly itself and set the maximum number of candidates to $N = 20$. The reference images are realistic photos collected from the web. We adopt SAM [10] and grayscale processing to obtain the texture-less foreground object as the input of our model. As shown, our method is capable of accurately identifying the missing parts in most cases, without view constraints on the image. Errors mainly arise in cases involving integrated parts, such as the backrest of the third chair, which differs from the fine-grained parts learned by the model.

Ablation	PA(%) \uparrow	MA(%) \uparrow	CCD(10^{-2}) \downarrow
w/o \mathcal{L}_t	26.77	38.05	1.04
w/o \mathcal{L}_r	36.02	46.43	1.08
w/o \mathcal{L}_{scd}	25.02	42.84	1.67
w/o \mathcal{L}_{sym}	29.79	44.47	1.41
w/o image	25.25	34.68	0.69
w/o alignment	28.56	39.04	1.47
w/o discovery	28.61	38.53	1.46
w/o autoregression	37.61	46.12	1.17
Ours	39.84	49.43	0.99

Table 3. Ablation of the loss components and designed modules.

4.5. Ablation Study

Effect of the additional image. To explore the function of image, we utilize the features of partial assembly F_A to compute cross-attention with those of the candidate parts, disabling the cross-modal encoder. As shown in Tab. 3, this leads to an improvement in CCD while a decline in the other metrics (refer to *w/o image*). This phenomenon can be attributed to the substantial uncertainty in the completion process when image information is not available. Without visual guidance, various combinations of parts could potentially ‘fit’ the region of the missing parts, resulting in a smaller CCD despite the incorrectness of these parts. This also indirectly validates the effectiveness of our designed image-guided assembly completion framework in eliminating ambiguities of completion and guiding precise assembly, which is further corroborated by the visual examples in Fig. 9(a). In the absence of the image, it is hard to correctly locate the missing bars on the back, whereas the introduction of image information compensates for this deficiency.

Effect of view alignment. In Fig. 8, we visualize the rendering of partial assembly with view transformation and compare it with the silhouette of image. As shown, the predicted view is consistent with that of the image with little deviation. The blur on the boundary is caused by the nature of differential rendering. Without view alignment, the cross-modal fusion between partial assembly and image becomes aimless and inefficient, with a drastic decline in performance (see *w/o alignment* in Tab. 3).

Effect of missing discovery. In feature fusion, missing discovery with a differential attention captures potential information of missing parts from image modality and facilitates the establishment of view-consistent 2D-3D correspondence. As shown in Tab. 3, we observe a decline in PA and in MA without missing discovery (see *w/o discovery*), indicating its effectiveness. Additionally, a qualitative analysis is presented in Fig. 9(a). Equipped with missing discovery, our method is able to distinguish the missing bars on the back that are close together and locate their positions precisely. Note that this design relies on view alignment to determine

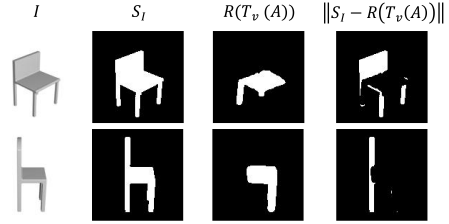


Figure 8. Qualitative evaluation of view alignment.

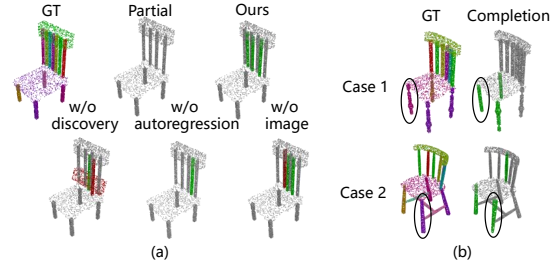


Figure 9. (a) Effect of designed components. (b) Limitations.

the correct missing regions.

Effect of loss components. We investigate the effect of loss components by removing them separately. Tab. 3 reveals that both \mathcal{L}_t and \mathcal{L}_{scd} are crucial for supervising the predicted poses. Additionally, \mathcal{L}_t significantly influences the accuracy of part retrieval. \mathcal{L}_{sym} and \mathcal{L}_{scd} contribute to the overall completion effect (CCD).

Effect of autoregressive learning. We remove autoregressive learning and predict all k missing parts once. In this case, we select top- k candidate parts ranked by their scores as missing parts. As shown in Fig. 9(a), multiple identical parts are likely to compete for the same position without autoregressive learning, which is an issue reported by PhysFiT [25].

5. Conclusion

In this paper, we present a novel image-guided 3D assembly completion framework. Our core idea lies in joint learning of missing-oriented feature fusion and self-supervised view alignment, to establish a view-consistent correspondence between 3D partial assembly and 2D image. As evaluated, our method better comprehends the visual details of missing parts from the image for more precise and generalizable assembly completion.

Limitations. We illustrate several challenging examples in Fig. 9(b). Even with image guidance, distinguishing parts that are extremely similar in geometry remains difficult (case 1). Additionally, identifying the correct orientation for elongated, stripe-like parts can be challenging (case 2). Further improvement could be achieved by incorporating connectivity constraints for more precise completion or by extending the framework towards the completion of furniture in real-world environments with real-scanned data.

Acknowledgements. This work was supported in part by the National Natural Science Foundation of China under Grant No. 62473290, 62088101, National Key Research and Development Program of China under Grant No. 2024YFB3311801, and the Shanghai Municipal Commission of Economy and Informatization under Grant No. 2024-GZL-RGZN-01008.

References

- [1] Emanuele Aiello, Diego Valsesia, and Enrico Magli. Cross-modal learning for image-guided point cloud shape completion. *Advances in Neural Information Processing Systems*, 35:37349–37362, 2022. 2, 3, 5
- [2] Yun-Chun Chen, Haoda Li, Dylan Turpin, Alec Jacobson, and Animesh Garg. Neural shape mating: Self-supervised object assembly with adversarial shape priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12724–12733, 2022. 2
- [3] Junfeng Cheng, Mingdong Wu, Ruiyuan Zhang, Guanqi Zhan, Chao Wu, and Hao Dong. Score-pa: Score-based 3d part assembly. *arXiv preprint arXiv:2309.04220*, 2023. 2
- [4] Ruihang Chu, Enze Xie, Shentong Mo, Zhenguo Li, Matthias Nießner, Chi-Wing Fu, and Jiaya Jia. Diffcomplete: Diffusion-based generative 3d shape completion. *Advances in Neural Information Processing Systems*, 2023. 3
- [5] Bi'an Du, Xiang Gao, Wei Hu, and Renjie Liao. Generative 3d part assembly via part-whole-hierarchy message passing. *arXiv preprint arXiv:2402.17464*, 2024. 2
- [6] Zhenjiang Du, Jiale Dou, Zhitao Liu, Jiwei Wei, Guan Wang, Ning Xie, and Yang Yang. Cdpnet: cross-modal dual phases network for point cloud completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1635–1643, 2024. 3
- [7] Abhinav Narayan Harish, Rajendra Nagar, and Shanmuganathan Raman. Rgl-net: A recurrent graph learning framework for progressive part assembly. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 647–656. IEEE, 2022. 2
- [8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3
- [9] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. *Advances in neural information processing systems*, 31, 2018. 3
- [10] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 7
- [11] Yichen Li, Kaichun Mo, Lin Shao, Minhyuk Sung, and Leonidas Guibas. Learning 3d part assembly from a single image. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 664–682. Springer, 2020. 2, 5
- [12] Yichen Li, Kaichun Mo, Yueqi Duan, He Wang, Jiequan Zhang, Lin Shao, Wojciech Matusik, and Leonidas Guibas. Category-level multi-part multi-joint 3d shape assembly. *arXiv preprint arXiv:2303.06163*, 2023. 2
- [13] Paritosh Mittal, Yen-Chi Cheng, Maneesh Singh, and Shubham Tulsiani. Autosdf: Shape priors for 3d completion, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 306–315, 2022. 3
- [14] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019. 2, 5
- [15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019. 5
- [16] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 4
- [17] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv preprint arXiv:2007.08501*, 2020. 5
- [18] Chun-Yu Sun, Xin Tong, and Yang Liu. Semantic segmentation-assisted instance feature fusion for multi-level 3d part instance segmentation. *Computational Visual Media*, 9(4):699–715, 2023. 2
- [19] Minhyuk Sung, Hao Su, Vladimir G Kim, Siddhartha Chaudhuri, and Leonidas Guibas. Complementme: Weakly-supervised component suggestions for 3d modeling. *ACM Transactions on Graphics (TOG)*, 36(6):1–12, 2017. 1, 2, 7
- [20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4
- [21] Haiping Wang, Yuan Liu, Bing Wang, Yujing Sun, Zhen Dong, Wenping Wang, and Bisheng Yang. Freereg: Image-to-point cloud registration leveraging pretrained diffusion models and monocular depth estimators. *arXiv preprint arXiv:2310.03420*, 2023. 2
- [22] Ruocheng Wang, Yunzhi Zhang, Jiayuan Mao, Chin-Yi Cheng, and Jiajun Wu. Translating a visual lego manual to a machine-executable plan. In *European Conference on Computer Vision*, pages 677–694. Springer, 2022. 2
- [23] Ruocheng Wang, Yunzhi Zhang, Jiayuan Mao, Ran Zhang, Chin-Yi Cheng, and Jiajun Wu. Ikea-manual: Seeing shape assembly step by step. *Advances in Neural Information Processing Systems*, 35:28428–28440, 2022. 7
- [24] Weihao Wang, Rufeng Zhang, Mingyu You, Hongjun Zhou, and Bin He. 3d assembly completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2663–2671, 2023. 1, 2, 5

- [25] Weihao Wang, Mingyu You, Hongjun Zhou, and Bin He. Physfit: Physical-aware 3d shape understanding for finishing incomplete assembly. *ACM Transactions on Graphics*, 2024. [1](#), [2](#), [8](#)
- [26] Rundi Wu, Yixin Zhuang, Kai Xu, Hao Zhang, and Baoquan Chen. Pq-net: A generative part seq2seq network for 3d shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 829–838, 2020. [2](#)
- [27] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jialong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024. [5](#)
- [28] Peng Xiang, Xin Wen, Yu-Shen Liu, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Zhizhong Han. SnowflakeNet: Point cloud completion by snowflake point deconvolution with skip-transformer. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2021. [3](#)
- [29] Boshen Xu, Sipeng Zheng, and Qin Jin. Spaformer: Sequential 3d part assembly with transformers. *arXiv preprint arXiv:2403.05874*, 2024. [2](#)
- [30] Hang Xu, Chen Long, Wenxiao Zhang, Yuan Liu, Zhen Cao, Zhen Dong, and Bisheng Yang. Explicitly guided information interaction network for cross-modal point cloud completion. In *European Conference on Computer Vision*, pages 414–432. Springer, 2024. [3](#), [5](#)
- [31] Xingguang Yan, Liqiang Lin, Niloy J Mitra, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Shapeformer: Transformer-based shape completion via sparse representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6239–6249, 2022. [3](#)
- [32] Xumin Yu, Yongming Rao, Ziyi Wang, Jiwen Lu, and Jie Zhou. Adapointr: Diverse point cloud completion with adaptive geometry-aware transformers. *arXiv preprint arXiv:2301.04545*, 2023. [3](#), [5](#)
- [33] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. Pcn: Point completion network. In *2018 international conference on 3D vision (3DV)*, pages 728–737. IEEE, 2018. [3](#)
- [34] Guanqi Zhan, Qingnan Fan, Kaichun Mo, Lin Shao, Baoquan Chen, Leonidas J Guibas, Hao Dong, et al. Generative 3d part assembly via dynamic graph learning. *Advances in Neural Information Processing Systems*, 33:6315–6326, 2020. [2](#), [5](#)
- [35] Rufeng Zhang, Tao Kong, Weihao Wang, Xuan Han, and Mingyu You. 3d part assembly generation with instance encoded transformer. *IEEE Robotics and Automation Letters*, 2022. [2](#)
- [36] Xuancheng Zhang, Yutong Feng, Siqi Li, Changqing Zou, Hai Wan, Xibin Zhao, Yandong Guo, and Yue Gao. View-guided point cloud completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15890–15899, 2021. [2](#), [3](#)
- [37] Haoran Zhou, Yun Cao, Wenqing Chu, Junwei Zhu, Tong Lu, Ying Tai, and Chengjie Wang. Seedformer: Patch seeds based point cloud completion with upsample transformer. In *European conference on computer vision*, pages 416–432. Springer, 2022. [3](#)