PHFormer: Multi-Fragment Assembly Using Proxy-Level Hybrid Transformer

Wenting Cui, Runzhao Yao, Shaoyi Du*

National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University
wentingcui15222206449@gmail.com, rzy3320@163.com, dushaoyi@gmail.com

Abstract

Fragment assembly involves restoring broken objects to their original geometries, and has many applications, such as archaeological restoration. Existing learning based frameworks have shown potential for solving part assembly problems with semantic decomposition, but cannot handle such geometrical decomposition problems. In this work, we propose a novel assembly framework, proxy level hybrid Transformer, with the core idea of using a hybrid graph to model and reason complex structural relationships between patches of fragments, dubbed as proxies. To this end, we propose a hybrid attention module, composed of intra and inter attention layers, enabling capturing of crucial contextual information within fragments and relative structural knowledge across fragments. Furthermore, we propose an adjacency aware hierarchical pose estimator, exploiting a decompose and integrate strategy. It progressively predicts adjacent probability and relative poses between fragments, and then implicitly infers their absolute poses by dynamic information integration. Extensive experimental results demonstrate that our method effectively reduces assembly errors while maintaining fast inference speed. The code is available at https://github.com/521piglet/PHFormer.

Introduction

Fragment assembly restores a set of fractured fragments into their original shape by predicting a 6-DoF pose for each fragment. It is a challenging problem because the solution space becomes extremely large as the number of fragments increases, and has many applications such as archaeological restoration (Toler-Franklin et al. 2010), forensic evidence analysis (Yin et al. 2011), and reduction of comminuted bone fractures (Idram et al. 2019).

Some traditional fragment assembly frameworks (Papaioannou and Karabassi 2003; Willis and Cooper 2004; Wei et al. 2011; Zhang et al. 2015; Huang et al. 2006) use pair-wise and multi-piece alignment approaches to sequentially solve poses, where hand-crafted local features (Rusu, Blodow, and Beetz 2009; Zhong 2009) are used to assign fracture-to-fracture correspondences or fracture-to-template correspondences. However, these approaches are limited by specific shape assumptions (Papaioannou and Karabassi 2003; Willis and Cooper 2004; Wei et al. 2011), complete model requirements (Yin et al. 2011; Zhang et al. 2015), and the poor discrimination of hand-crafted features (Toler-Franklin et al. 2010; Huang et al. 2006). Recently, deep learning-based methods have made significant progress in 3D vision tasks (Guo et al. 2020) due to their powerful feature extraction and logical reasoning capabilities. However, using deep networks for fragment assembly has not received much attention. Therefore, in this paper, we propose a learning-based framework for fragment assembly to fill the research gap in this field.

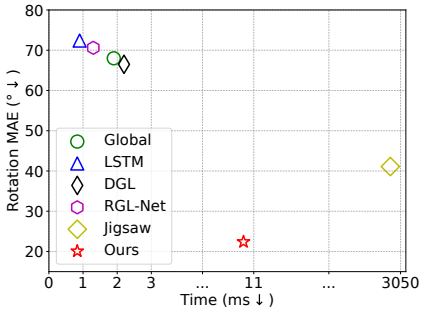

Figure 1: Experimental results on Breaking Bad (Sellán et al. 2022). Our approach significantly reduces the rotation error and maintains fast inference speed.

The most relevant research to our task is the part assembly, and many learning-based methods (Zhan et al. 2020; Li et al. 2020; Zhang et al. 2022; Narayan, Nagar, and Raman 2022; Li et al. 2023; Narayan, Nagar, and Raman 2022) have been proposed. The core of these methods is how to effectively reason about relationships between parts and infer their poses to form a complete physical object through organic combinations. However, these methods are not effective in addressing the problem of fragment assembly, as shown in Fig. 1. This is because parts have complete shapes and semantic information, while fragments have various irregular shapes without semantic consistency (Sellán et al. 2022). The semantic and structural integrity of parts facilitates network learning for understanding the role of different parts and the relationships between them. Additionally, part

*Corresponding author.

poses often conform to their functional constraints, such as chair legs being vertical to support the weight. When these favorable conditions do not exist for fragment assembly, understanding the relationships between fragments and inferring their poses becomes challenging, especially when there are a large number of fragments with thickness fracture surfaces, such as fractured thin-shell objects.

Our core idea is that when semantic integrity no longer exists, local geometric features of fragments become the most important cues for reasoning about relative position relationships between fragments. That is to say, learning global features of fragments like in part assembly (Zhan et al. 2020; Li et al. 2020; Zhang et al. 2022; Narayan, Nagar, and Raman 2022; Li et al. 2023) no longer meets the requirements of the current task, and it is necessary to extract fine-grained local features. One severe problem that arises from this line is that extracting features for every point of each fragment and then matching them would result in a massive computational burden. To address this issue, we propose a novel **P**roxy-Level **H**ybrid Trans**former** (PHFormer) that partitions the point clouds of fragments into local patches, known as proxies, and uses a proposed hybrid attention mechanism that includes intra-fragment attention and inter-fragment attention (Vaswani et al. 2017; Sarlin et al. 2020) to learn global contextual information of proxies within the fragment and the relational connections between proxies of different fragments, respectively. This hybrid attention module enables our model to better learn the knowledge of object structures for fragment assembly. Furthermore, considering that directly regressing absolute poses for multiple fragments is very difficult, we propose an adjacency-aware hierarchical pose estimator. It adopts a decompose-and-integrate strategy, which first predicts the adjacency probabilities and relative poses between fragments, and then dynamically integrates this information for absolute pose estimation. We conduct extensive experiments on a large-scale fractured objects dataset, Breaking Bad (Sellán et al. 2022), and the experimental results demonstrate the excellent performance of our method. As shown in Fig. 1, it dramatically reduces the mean absolute error of rotation while keeping fast inference speed. Overall, our contributions can be summarized:

- We propose a novel proxy-level hybrid Transformer that uses a hybrid attention mechanism to reason about the fine-grained contexts and structural relationships between the proxies.
- We propose an adjacency-aware hierarchical pose estimator using a decompose-and-integrate strategy to progressively predict accurate poses of fragments.
- Extensive experimental studies demonstrate significant improvement of our method in fragment assembly.

Related Work

Fractured fragment assembly. The demand for archaeological artifact restoration and forensic investigation has driven the growth of research in this field. Previous works can be generally categorized into fracture-region matching based assembly approaches and template-guided assembly approaches. The former line of approaches focuses on detecting fractured surfaces and extracting robust hand-crafted descriptors, *e.g.* Curvatures (Ruiz-Correa, Shapiro, and Melia 2001), Integral Invariants (Gelfand et al. 2005), SHOT (Salti, Tombari, and Di Stefano 2014), and FPFH (Rusu, Blodow, and Beetz 2009), to establish correspondences between points located on fractured surfaces. Huang *et. al.* (Huang et al. 2006) presented an integral invariant for surface segmentation and feature selection, and subsequently used pair-wise and global matching methods to achieve object reassembly. The latter line of works restores objects by matching the fragmented pieces to the complete models. Yin *et. al.* (Yin et al. 2011) introduced a two-step assembly pipeline for skulls, involving rough assembly by a template matching algorithm and refinement by globally optimizing errors of break-curves. Zhang *et. al.* (Zhang et al. 2015) integrated these two routines by matching both fragments to fragments and fragments to template. Although achieving good performance on specific datasets with limited shapes, these approaches suffer from shape assumptions (Willis and Cooper 2004; Wei et al. 2011), complete model requirements (Yin et al. 2011; Zhang et al. 2015), and the poor discrimination of hand-crafted features (Toler-Franklin et al. 2010; Huang et al. 2006).

Recently, some learning-based fragment assembly methods have been presented. NSM (Chen et al. 2022) leveraged an encoder to extract dense point features and used a Transformer network to reason correlation. However, it can only restore two fragments and cannot address the multi-fragment assembly task. Wu *et. al.* (Wu et al. 2023) exploited a SE(3) equivariant network to disentangle shapes and poses of fractured parts for geometric shape assembly. The feature embeddings of fragments were used to reason correlations by matrix multiplication, which cannot effectively learn the complex relationship between fragments. Jigsaw (Lu, Sun, and Huang 2023) employed a similar pipeline as Huang *et. al.* (Huang et al. 2006), but leveraged networks to extract dense point features and detect fractured surfaces. However, it is extremely time-consuming as shown in Fig. 1.

Semantic part assembly. 3D part assembly is a widely studied task in the robotic and computer vision fields. Previous works (Wu et al. 2020; Funkhouser et al. 2004; Chaudhuri and Koltun 2010; Jones et al. 2020) aim to synthesize complete 3D shapes, based on probabilistic models (Chaudhuri et al. 2011; Kalogerakis et al. 2012; Jaiswal, Huang, and Rai 2016; Wu et al. 2016) or generative networks (Li et al. 2017; Mo et al. 2020a,b; Li, Liu, and Walder 2022). Recently, some attention has shifted to predicting precise poses of given semantic parts to recover the complete shapes. Li *et. al.* (Li et al. 2020) presented a part assembly framework guided by a RGB image. It first extracted both geometric and textural features for each part and then used a graph network to reason the relationship of parts for pose prediction. Zhan *et. al.* (Zhan et al. 2020) proposed an assembly-oriented dynamic graph learning framework in a coarse-to-fine manner that iteratively passed messages between part nodes for pose refinement. Zhang *et. al.* (Zhang et al. 2022) leveraged a transformer-based framework to learn the relationship of parts, and particularly designed an instance encoding to al-

leviate the ambiguity of geometrically similar parts. Harish *et. al.* (Narayan, Nagar, and Raman 2022) used a recurrent graph learning framework that allows the adoption of knowledge from pre-assembled components. Li *et. al.* (Li et al. 2023) paid more attention to the joint alignment and presented a dual-level graph learning framework to reason joint connections and geometric structures.

However, they either use additional guided information (Li et al. 2020), part ordering information (Narayan, Nagar, and Raman 2022), or joint annotation (Li et al. 2023), which limits their practical application. Moreover, they require a semantic decomposition of the components of the shape and show a dramatic degradation in performance when dealing with fragments without semantics (Sellán et al. 2022). In this work, we break this limitation by proposing an automatic assembly framework for pose prediction of fragments based solely on point clouds of fragments, without other decomposition assumptions or additional knowledge.

Method

Given a set of point clouds $\mathcal{P} = \{\mathcal{P}_i | \mathcal{P}_i \in \mathbb{R}^{N_p \times 3}\}_{i=1}^{N}$, representing geometric contours of fragments, where N is the number of fragments and N_p is the number of points of each fragment, our goal is to predict 6-DoF poses $T = \{T_i | T_i \in SE(3)\}$ that can be used to transform fragments for restoration of the complete shape. To this end, we propose a proxy-level hybrid Transformer, whose framework is shown in Fig. 2. It first extracts proxy-level features using a hierarchical encoder. Then, a hybrid attention module is used to iteratively reason the global contexts of proxies within fragments and the relative correlation between proxies across fragments. Finally, an adjacency-aware hierarchical pose estimator is leveraged to progressively predict relative transformations and global poses of fragments.

Encoder

Existing part assembly methods (Zhan et al. 2020; Zhang et al. 2022; Narayan, Nagar, and Raman 2022; Li et al. 2023) utilize PointNet (Qi et al. 2017a) to extract global features of parts, which is feasible when the parts are semantically-consistent decomposed. However, for fragment assembly, fine-grained geometric features are crucial for inferring their relationships and relative poses. Extracting point-level features like NSM (Chen et al. 2022) and Jigsaw (Lu, Sun, and Huang 2023) instead introduces a heavy computational burden for subsequent correlation inference, especially when dealing with a large number of fragments. To overcome this contradiction, we adopt a balanced approach by dividing each fragment into patches, dubbed proxies, and then employ a hierarchical network to extract patch-level features.

Specifically, we leverage the farthest point sampling (Qi et al. 2017b) to sample K keypoints within each fragment $\mathcal{P}_i$, and then utilize ball query to group neighboring points, thus generating proxies. Finally, we employ EdgeConv (Wang et al. 2019) to compute the proxy features. After several encode layers, we obtain the final coordinates $\mathcal{X}_i = \{\mathcal{X}_{i,k}\}_{k=1}^{K}$ and local features $\mathcal{F}_i = \{\mathcal{F}_{i,k}\}_{k=1}^{K}$ of the proxies, where $\mathcal{F}_{i,k} \in \mathbb{R}^d$ and d is the dimension of the feature ($d = 128$).

Hybrid Attention Module

To predict the pose of each fragment, it is crucial to perceive the shape of the object and reason about the position of the fragments within the object. We use the proxy-level features to infer more fine-grained relationships between fragments. However, the relationships between proxies are complex and can be classified into three categories: 1) belonging to the same fragment; 2) belonging to different fragments but located on the same fractured surface; and 3) belonging to different fragments and not adjacent. To infer the complex relationships between the proxies for better pose estimation, we propose a hybrid attention module, consisting of intra-fragment attention layers and inter-fragment attention layers. Intra-fragment attention layer aims to capture the contextual information of proxies within fragments, while the inter-fragment attention layer is designed for inferring the relationships between proxies belonging to different fragments.

Intra-fragment attention layer. This layer operates on an intra graph $\mathcal{G}_{intra} = (\mathcal{V}, \mathcal{E}_{intra})$, where $\mathcal{V}_{K\times i+k} = \mathcal{X}_{i,k}$ and $\mathcal{E}_{intra} = \{(\mathcal{V}_m, \mathcal{V}_n) | \lfloor \frac{m}{K} \rfloor = \lfloor \frac{n}{K} \rfloor\}$, *i.e.* proxies belonging to the same fragment are connected. For the initial representation of each node, we combine the geometric features and the location information by encoding the coordinates $\mathcal{X}$ into vectors and adding them to the local embeddings $\mathcal{F}$, *i.e.* $\bar{\mathcal{F}}^0 = \mathcal{F} + \mathrm{PE}(\mathcal{X})$. Here, we use the sinusoidal positional encodings (Vaswani et al. 2017; Yew and Lee 2022), and refer to Appendix for more details. Let $\bar{\mathcal{F}}_m^l$ be the intermediate representation for the node $\mathcal{V}_m$ and the feature embedding is updated by the attention operation (Vaswani et al. 2017):

$$\bar{\mathcal{F}}_m^{l+1} = \bar{\mathcal{F}}_m^l + \phi(\mathrm{cat}[\bar{\mathcal{F}}_m, \sum_n \alpha_{m,n} \bar{\mathcal{F}}_n^l \mathbf{W}_\mathrm{v}]), \tag{1}$$

where $\alpha_{m,n} = \underset{n}{\mathrm{softmax}}[(\bar{\mathcal{F}}_m^l \mathbf{W}_\mathrm{q})(\bar{\mathcal{F}}_n^l \mathbf{W}_\mathrm{k})^T / \sqrt{d}]$ is the attention weight, $\mathrm{cat}[\cdot, \cdot]$ is the concatenation operation, $\mathbf{W}_\mathrm{q}, \mathbf{W}_\mathrm{k}, \mathbf{W}_\mathrm{v} \in \mathbb{R}^{d\times d}$ are learnable projection matrices, and ϕ is a multi-layer perceptron (MLP) block.

Inter-fragment attention layer. The formation of this layer is similar to the intra-fragment attention layer, except that it operates on the inter graph $\mathcal{G}_{inter} = (\mathcal{V}, \mathcal{E}_{inter})$ to aggregate messages from the proxies that belonging to different fragments, where $\mathcal{E}_{inter} = \{(\mathcal{V}_m, \mathcal{V}_n) | \lfloor \frac{m}{K} \rfloor \neq \lfloor \frac{n}{K} \rfloor\}$.

In this hybrid attention module, we iteratively perform intra- and inter-fragment attention layers for L times ($L = 2$). Thus, the output feature $\bar{\mathcal{F}}_m^L$ of each proxy encodes both the contextual information within fragments and correlation information across different fragments. Finally, these proxy-wise features are aggregated into fragment-wise features $\{\hat{\mathcal{F}}_i\}$ for the subsequent pose estimation, which is achieved by the permutation-invariant max-pooling operation within each fragment:

$$\hat{\mathcal{F}}_i = \mathrm{MAX}\{\bar{\mathcal{F}}_{i,k}^L\}_{k=1,2,\cdots,K}, i = 1, 2, \cdots, N. \tag{2}$$

Adjacency-Aware Hierarchical Pose Estimator

With the feature embedding of each fragment, we predict a 6 DoF pose T_i for each fragment:

$$\{T_1, T_2, \cdots, T_N\} = \rho(\{\hat{\mathcal{F}}_1, \hat{\mathcal{F}}_2, \cdots, \hat{\mathcal{F}}_N\}), \tag{3}$$

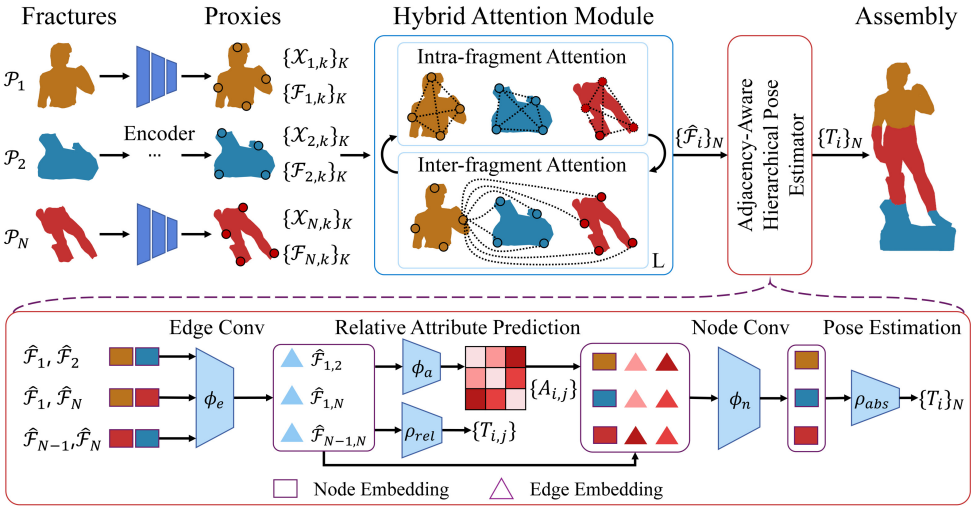

Figure 2: Framework of the proposed method. With point clouds $\{\mathcal{P}_i\}_N$ of fragments, we first use a hierarchical encoder to extract K proxies $\{\mathcal{X}_{i,k}\}_K$ and their features $\{\mathcal{F}_{i,k}\}_K$ for each fragment. Then, we design a hybrid attention module to iteratively reason the global contextual information of proxies within fragments and the relative correlation between proxies across fragments, outputting the fragment-level features $\{\hat{\mathcal{F}}_i\}$. Finally, we introduce an adjacency-aware hierarchical pose estimator to progressively predict adjacent weights $\{A_{i,j}\}$, relative poses $\{T_{i,j}\}$, and global poses $\{T_i\}$ of fragments.

where ρ is a pose estimator. It can be observed that the solution space is particularly large due to combinatorial effects. Some part assembly methods (Zhan et al. 2020; Li et al. 2020; Zhang et al. 2022) use a MLP block to directly regress poses. However, this approach is impracticable for fragment assembly. This is because in part assembly there is a more distinct regularity in the pose of the parts, *e.g.* the backrest and legs of a chair are usually perpendicular to the floor, while the chair seat is parallel to the floor. However, the lack of semantics of the fragments leads to greater variability in their poses. Therefore, it is more complicated to estimate the poses of fragments.

To address this problem, we propose an adjacency-aware hierarchical pose estimator, exploiting a decompose-and-integrate strategy. The core idea is to decompose the problem defined in Eq. (3) into three sub-problems: pair-wise adjacency prediction, pair-wise transformation prediction, and absolute pose regression, and leverage graph learning technologies (Battaglia et al. 2018) to solve them. To this end, the fragments are formulated as a graph, where the pair-wise adjacency and transformation matrices are predicted using edge features aggregated from nodes. Subsequently, the messages from edges are selectively integrated into nodes for absolute pose estimation.

Relative attribute prediction. With the attribute $\hat{\mathcal{F}}_i$ of each node, we first perform edge convolution to aggregate pair-wise information,

$$\hat{\mathcal{F}}_{i,j} = \phi_e(\text{cat}[\hat{\mathcal{F}}_i, \hat{\mathcal{F}}_j]), \tag{4}$$

where $\phi_e : \mathbb{R}^{2d} \to \mathbb{R}^d$ is a MLP block with linear projection, BatchNorm (Ioffe and Szegedy 2015), and ReLU (Agarap 2018) layers. Based on the edge feature, we predict the adjacency and relative transformation between two fragments:

$$T_{i,j} = \rho_{rel}(\hat{\mathcal{F}}_{i,j}), A_{i,j} = \text{sigmoid}(\phi_a(\hat{\mathcal{F}}_{i,j})), \tag{5}$$

where $\phi_a : \mathbb{R}^d \to \mathbb{R}^1$ is a MLP block to predict an adjacent weight $A_{i,j}$, $\rho_{rel} : \mathbb{R}^d \to \mathbb{R}^7$ is a pose regressor to predict relative pose $T_{i,j}$, consisting a MLP block, two linear projection layers for translation and rotation regression. Similar to existing works (Li et al. 2020, 2023), we use Quaternion to represent rotations, and the output rotation vectors are normalized. These two predicted attributes are supervised by two losses, which we will describe in detail later.

Adjacency-aware absolute pose prediction. After predicting the relative attributes $(T_{i,j}, A_{i,j})$, we aggregate this information to predict the absolution pose T_i of each fragment. Instead of explicitly using the relative transformation $T_{i,j}$, we implicitly aggregate relative information from features $\hat{\mathcal{F}}_{i,j}$. Considering that the relative information between non-adjacent fragment pairs may be unreliable, we pass the messages of edges to nodes weighted by the adjacent probability $A_{i,j}$:

$$T_i = \rho_{abs}(\phi_n(\text{cat}[\hat{\mathcal{F}}_i, \frac{1}{\sigma_i}\sum_{j=1}^{N} A_{i,j}\hat{\mathcal{F}}_{i,j}])), i = 1, 2, \cdots, N, \tag{6}$$

where $\sigma_i = \sum_{j=1}^{N} A_{i,j}$ is a normalized factor, $\phi_n : \mathbb{R}^{2d} \to \mathbb{R}^d$ is a MLP block for node feature update, and $\rho_{abs} : \mathbb{R}^d \to \mathbb{R}^7$ is the absolute pose regressor, whose components are same as ρ_{rel}.

Losses

We use multiple losses to supervise the assembly network, including adjacent loss, relative and absolute pose losses, Chamfer distance loss, and L2 distance loss.

Adjacent loss. we measure the mean square error (MSE) between ground-truth adjacent labels $A^*_{i,j}$ and the predicted

adjacent weights $A_{i,j}$,

$$\mathcal{L}_{\text{adj}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}(A_{i,j} - A^*_{i,j})^2. \tag{7}$$

To obtain the adjacent labels, we first assemble fragments using ground-truth poses and then count the number of adjacent points between two fragments. If the number is greater than a certain threshold (*e.g.* 10), they are considered to be adjacent fragments.

Pose loss. The pose loss is used to supervise the predictions of rotation and translation. Given the ground-truth pose as $T^*_i = (R^*_i, t^*_i)$, the absolute pose loss is formulated as:

$$\mathcal{L}^{\text{abs}}_{\text{pose}} = \frac{1}{N}\sum_{i=1}^{N}\lambda_{\text{rot}}\left\|R_i^T R^*_i - I\right\|_2^2 + \left\|t_i - t^*_i\right\|_2^2. \tag{8}$$

To supervise the relative transformations of fragment pairs, we derive the ground-truth transformations by $T^*_{i,j} = (T^*_j)^{-1}T^*_i$, and compute the relative pose loss $\mathcal{L}^{\text{rel}}_{\text{pose}}$ by the Eq. (8). Note that only the losses of adjacent fragment pairs are calculated for the relative pose loss.

Shape loss. Because many objects have symmetric shapes (*e.g.* bottles and bowls), leading to multiple pose solutions. We use Chamfer distance loss $\mathcal{L}_{\text{cham}}$ and L2 distance loss $\mathcal{L}_{L_2}$ to alleviate such ambiguity as prior works (Zhan et al. 2020; Li et al. 2020). Please refer to the Appendix for more details about these losses.

Overall loss. The overall loss $\mathcal{L}$ is used to supervise the model with hyper-parameters balancing different loss terms,

$$\mathcal{L} = \lambda_{\text{adj}}\mathcal{L}_{\text{adj}} + \lambda_{pose}(\mathcal{L}^{\text{abs}}_{\text{pose}} + \mathcal{L}^{\text{rel}}_{\text{pose}}) + \lambda_{\text{cham}}\mathcal{L}_{\text{cham}} + \lambda_{L_2}\mathcal{L}_{L_2}. \tag{9}$$

Experiments

In this section, we evaluate the performance of our method on the Breaking Bad dataset (Sellán et al. 2022). Both qualitative and quantitative results demonstrate the superiority of our method. In addition, ablation studies are performed to analyze the effectiveness of the components of our method.

Experimental Setup

Dataset. We conduct experiments on a large-scale fractured object dataset, Breaking Bad (Sellán et al. 2022), which contains 10,474 shapes and 1,047,400 breakdown patterns. It is divided into three sub-datasets, including Everyday, Artifact, and Other. Following (Sellán et al. 2022), the Everyday dataset is used to perform comparison experiments, the Artifact is used to conduct pre-training experiments, and the Other dataset is used for generalization studies. For each sub-dataset, 80% objects are leveraged for training, and the remaining 20% objects for testing.

Baseline methods. We compare our method with learning-based part assembly approaches, Global (Li, Niu, and Xu 2020; Schor et al. 2019; Zhan et al. 2020), LSTM (Wu et al. 2020; Zhan et al. 2020), DGL (Zhan et al. 2020), RGL-Net (Narayan, Nagar, and Raman 2022) and recent learning-based fragment assembly method, SGSA (Wu et al. 2023), and Jigsaw (Lu, Sun, and Huang 2023).

Evaluation metrics. We evaluate the predicted rotation and translation of fragments using root mean square error (RMSE) and mean absolute error (MAE), following (Chen et al. 2022; Sellán et al. 2022). To evaluate the restored shape, we adopt shape Chamfer distance (CD) and part accuracy (PA) metrics as the protocol in (Li et al. 2020; Sellán et al. 2022). Please refer to the Appendix for more details.

Implementation details. We implement our approach using PyTorch (Paszke et al. 2019) on two NVIDIA RTX3090 GPUs. The Adam optimizer (Kingma and Ba 2014) is adopted for training, with initial learning rate 5×10^{-4}, terminated learning rate 5×10^{-6}, and a cosine scheduler (Loshchilov and Hutter 2016). We train the model 400 epochs with 20 epochs for warm-up. The batch size is set to 32. The poses of shapes in the dataset are normalized into the canonical space using PCA. We randomly sample 1000 points for each fragment point cloud that are randomly rotated and translated to simulate the broken results. In the encoder, the fragment point clouds are down-sampled three times, from 1000 points to 256, 64, and 32, to obtain proxies. This means that 32 proxies ($K = 32$) are generated for each fragment with the feature dimension $d = 128$. In the hybrid attention module, we perform intra- and inter-attention loops 2 times. For the hyper-parameters in the loss function, λ_{adj}, λ_{rot}, λ_{pose} , λ_{cham} , λ_{L_2} are set to 1, 0.2, 1, 10, and 1, respectively, following (Sellán et al. 2022).

Experiments and Analysis

Comparison experiments. We conduct comparison experiments on the Everyday subset, which contains 20 categories of objects commonly seen in life, such as bottles, bowls, cups, etc. Existing works (Sellán et al. 2022; Zhan et al. 2020; Li et al. 2020, 2023) train a category-specific model for each category and report the averaged results overall categories. This wastes computational resources and cannot test the model's ability to handle multi-category objects. In contrast, we train a category-general model for all categories and do not receive the category labels of objects. We use the codes provided by the benchmark (Sellán et al. 2022) to train models of Gobal, LSTM, DGL, and RGL-Net. RGL-Net uses a recurrent graph network to iteratively reason the poses of parts and requires the order of semantic parts as prior information. We introduce this prior by sorting the input fragments to ensure any fragment is connected with one of the previous fragments. In addition, we bring the results of S-GSA reported in their paper and test Jigsaw using the official models and codes. Following (Sellán et al. 2022), all models are trained and tested on objects with at most 20 fragments. [1] The averaged evaluation results overall categories are reported in Table 1.

We can observe that our method dramatically outperforms these baseline methods in terms of assembly errors, while keeping fast inference speed. Compared to part assembly approaches, Jigsaw demonstrates more competitive performance. However, it is extremely time-consuming due to its point-wise matching and global alignment steps and

[1]The performance of the model to handle objects with more than 20 fragments is discussed in the Appendix.

Methods	Rotation RMSE ↓	Rotation MAE↓	Translation RMSE↓	Translation MAE↓	CD ↓	PA ↑	Time ↓
	degree	degree	$\times 10^{-2}$	$\times 10^{-2}$	$\times 10^{-3}$	%	ms
Global	80.7	68.0	15.1	12.0	14.6	24.6	1.9
LSTM	84.2	72.4	16.2	12.6	15.8	22.7	**0.9**
DGL	79.4	66.5	15.0	11.9	14.3	31.0	2.2
RGL-Net	83.3	70.6	14.9	11.9	14.8	25.3	1.3
SGSA	75.3	-	14.1	-	-	26.7	-
Jigsaw	47.9	41.1	12.4	10.1	-	**51.6**	3048.4
Ours	**26.1**	**22.4**	**9.3**	**7.5**	**9.6**	50.7	10.7

Table 1: Quantitative experimental results of different approaches when handling fractured objects with at most 20 fragments.

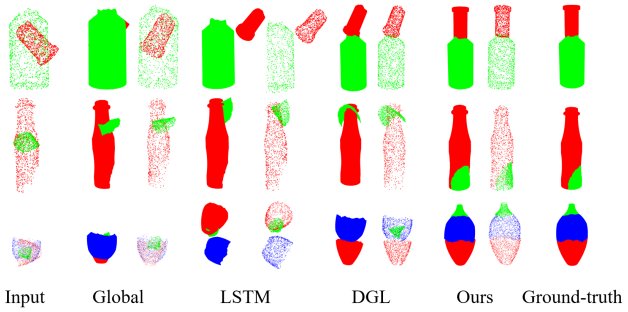

Figure 3: Visualized results of different methods. We use both meshes and point clouds for more clear visualizations. It can be observed that our approach can accurately assemble thin-shell objects.

takes about 3048.4ms to assemble a fractured object, while our method only uses 10.7ms, which is much faster. For SGSA, only slight improvement has been observed compared to semantic part assembly approaches because it overlooks the importance of learning fined-grained local features and designing an effective module to reason their relationship. Although introducing the order prior, RGL-Net still performed poorly, because it cannot effectively handle fragments without semantic meanings. For other part assembly approaches, their performance is much worse than our method. This is because the hybrid attention mechanism in our method utilizes fine-grained local features to better infer the relative relationships between proxies, while the relative relationship inference modules in these baseline methods are based on fragment-level features, which are difficult to learn local details to help infer poses. In addition, because our adjacency-aware hierarchical pose estimator adopts a divide-and-conquer strategy to break down the complex problem, it facilitates a better prediction of fragment poses. Besides, we report the results of different methods in assembling objects with varying numbers of fragments in the Appendix. Furthermore, we also show some visualized experiments in Fig. 3. Our method successfully assembles thin-shell objects, explicitly demonstrating the superiority of our method.

Analysis of model pre-training. Following Breaking Bad, we validate the performance of our method with the assistance of pre-training strategy. To do this, we trained two models on the Artifact dataset: one from scratch and another utilizing the model trained on the Everyday dataset. Similarly, both training and testing are done with fractured objects containing fewer than 20 fragments. The results of both models are shown in the Table 2. First of all, our method significantly outperforms the three baseline methods on the Artifact subset, regardless of whether the model is trained from scratch or fine-tuned, consistent with the results observed on the Everyday subset. Additionally, we can see that adopting a pre-training strategy can improve the performance of models, and the gains achieved by our method are more significant than those of the baselines. For baseline methods, pre-training results in a reduction in rotation error and an increase in PA, while our method improves all metrics.

Methods	Rotation RMSE ↓	Rotation MAE↓	Translation RMSE↓	Translation MAE↓	CD ↓	PA ↑
	degree	degree	$\times 10^{-2}$	$\times 10^{-2}$	$\times 10^{-3}$	%
Results of training the model from scratch						
Global	84.8	73.0	16.7	14.0	19.0	12.7
LSTM	85.2	73.6	17.2	14.3	23.5	6.6
DGL	85.8	74.2	16.8	13.9	19.4	12.8
Ours	**36.4**	**31.6**	**13.3**	**11.0**	**13.2**	**27.9**
Results of fine-tuning from the model trained on Everyday dataset						
Global	83.8	71.8	16.6	13.8	19.0	13.3
LSTM	84.6	73.1	16.8	14.0	21.5	11.7
DGL	81.7	69.7	16.6	13.8	17.3	19.4
Ours	**32.1**	**27.7**	**11.4**	**9.5**	**11.8**	**37.1**

Table 2: Quantitative experimental results on the Artifact dataset of different methods with the assistance of pre-training strategy.

Generalization experiments. We investigate the generalization abilities of our model to unseen objects. The experiments are conducted on the Other subset. Both the model trained on the Everyday subset and the model fine-tuned on the Artifact subset are evaluated. As shown in Table 3, all models experience performance degradation when encountering unseen data. However, our approach exhibits better generalization capabilities compared to the others. Addi-

Methods	Rotation RMSE ↓	Rotation MAE↓	Translation RMSE↓	Translation MAE↓	CD ↓	PA ↑
	degree	degree	$\times 10^{-2}$	$\times 10^{-2}$	$\times 10^{-3}$	%
Results of testing the model trained on Everyday dataset						
Global	86.4	72.9	19.4	16.3	42.2	6.0
LSTM	84.9	73.1	18.7	15.5	45.3	4.8
DGL	86.6	73.5	20.1	16.6	38.5	7.5
Ours	**44.5**	**39.1**	**16.6**	**13.9**	**33.5**	**11.1**
Results of testing the fine-tuned model trained on Artifact dataset						
Global	83.9	71.9	18.8	15.5	39.2	6.7
LSTM	82.9	70.3	17.9	14.9	40.3	5.5
DGL	81.3	69.9	17.2	14.5	36.6	8.3
Ours	**42.0**	**36.8**	**16.9**	**14.0**	**26.7**	**12.3**

Table 3: Generalization experiments on the Other subset.

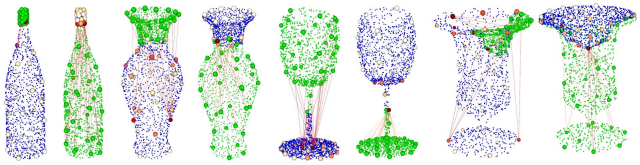

Figure 4: Visualizing inter-fragment attention. We visualize the edges with top-20 attention scores $\alpha_{m,n}$ between proxies from the blue to the green fragment. The proxies in blue fragments are colored by the summed scores $\sum_m \alpha_{m,n}$, where red indicates large scores and white indicates small scores.

tionally, we observe that the model trained on the Everyday dataset and fine-tuned on the Artifact dataset demonstrates better generalization performance. This is a reasonable phenomenon, as training on a broader range of data can facilitate model generalization.

Ablation experiments. To better understand the roles of various components in our model, we conduct ablation experiments, and the experimental results are shown in Table 4. We select a simple model (A) as our baseline, which extracts fragment features using PointNet (Qi et al. 2017a), uses attention layers to reason about relationships between fragments, and finally uses a MLP block to directly regress the absolute poses of the fragments. Based on this model, we progressively study the effects of different components. Firstly, we explore different encoders by replacing PointNet by EdgeConv (Wang et al. 2019) (B), the improvement indicates the significance of learning fine-grained local geometric features. Secondly, we replace the absolute pose estimator with our proposed adjacency-aware hierarchical pose estimator (C) and observe improvements in all metrics. We then replace fragment-level features with proxy-level features and used a fragment-aware attention module (D) to reason about relationships between proxies. Fragment-aware implies that we use one-hot vectors to encode which proxies belong to the same fragment. The results of the experiment show a significant improvement, demonstrating the

	Components				Rotation RMSE ↓	Rotation MAE↓	Translation RMSE↓	Translation MAE↓	CD ↓	PA ↑
	Enc.	Fe.	Re.	Po.	degree	degree	$\times 10^{-2}$	$\times 10^{-2}$	$\times 10^{-3}$	%
A	PN	F	I	A	50.8	44.2	13.7	10.9	19.6	24.2
B	EC	F	I	A	42.7	36.9	12.4	10.0	16.7	34.0
C	PN	F	I	H	48.0	41.6	13.3	10.7	18.5	27.5
D	EC	P	FI	H	28.0	24.1	10.2	8.2	12.0	46.6
E	EC	P	HB	A	28.4	24.4	9.9	8.0	10.9	47.2
F	EC	P	HB	H^-	26.9	23.0	9.4	7.7	10.7	49.0
G	EC	P	HB	H	**26.1**	**22.4**	**9.3**	**7.5**	**9.6**	**50.7**

Table 4: Ablation experiment results. "Enc." refers to the encoder, including PointNet (PN) and EdgeConv (EC). "Fe." indicates the feature embeddings of fragments (F) or proxies (P). "Re." represents the relational reasoning module, including intra-attention (I), fragment-aware intra-attention (FI), and our hybrid attention (HB). "Po." indicates the used pose estimators, including the absolute (A) estimator, our hierarchical (H) estimator, and hierarchical estimator without adjacent prediction (H^-).

importance of fine-grained features. However, the result still lags behind the use of our proposed hybrid attention module (G), revealing its significance. Moreover, we visualize inter-fragment attention in Fig. 4 for better understanding. It can be observed that the proxies in a fragment close to the fracture surface tend to have stronger connection weights with points that outline the contour of another fragment. This may suggest that assembling fragments requires not only the local structural information of the fractured surfaces but also knowledge of the object's global contour information to infer the poses of the fragments in the canonical space.

Additionally, we conduct experiments about pose estimators. Building on an absolute pose predictor (E), we gradually introduce the hierarchical estimator (F) and adjacency prediction (G). With each step, we observe a progressive performance improvement, demonstrating their efficacy.

Conclusion

In this paper, we have proposed a learning-based fragment assembly framework, mainly composed of a hybrid attention module and an adjacency-aware hierarchical pose estimator. The hybrid attention module leverages intra- and inter-fragment attention layers to refer complex relationships between proxies of different fragments, based on fine-grained local features. For better pose estimation of fragments, the proposed approach employs a decompose-and-integrate strategy, which first decomposes the multi-piece pose estimation problem into relative pose and adjacent probability prediction sub-problems, and then dynamically integrates this information for implicit global pose estimation. Extensive comparison and ablation experiments demonstrate the superior performance of our approach. However, the challenge of restoring objects with a large number of fragments (*e.g.*, more than 20) is still an issue that requires further investigation. We plan to conduct further research in the future to address this challenge.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant Nos. 62327808, 61971343, 62073257, and 62088102.

References

Agarap, A. F. 2018. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

Battaglia, P. W.; Hamrick, J. B.; Bapst, V.; Sanchez-Gonzalez, A.; Zambaldi, V.; Malinowski, M.; Tacchetti, A.; Raposo, D.; Santoro, A.; Faulkner, R.; et al. 2018. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.

Chaudhuri, S.; Kalogerakis, E.; Guibas, L.; and Koltun, V. 2011. Probabilistic reasoning for assembly-based 3D modeling. In *Proceedings of the ACM Special Interest Group on Computer Graphics and Interactive Techniques*, 1–10.

Chaudhuri, S.; and Koltun, V. 2010. Data-driven suggestions for creativity support in 3D modeling. In *Proceedings of the ACM Special Interest Group on Computer Graphics and Interactive Techniques Asia*, 1–10.

Chen, Y.-C.; Li, H.; Turpin, D.; Jacobson, A.; and Garg, A. 2022. Neural shape mating: Self-supervised object assembly with adversarial shape priors. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 12724–12733.

Funkhouser, T.; Kazhdan, M.; Shilane, P.; Min, P.; Kiefer, W.; Tal, A.; Rusinkiewicz, S.; and Dobkin, D. 2004. Modeling by example. *ACM Transactions on Graphics*, 23(3): 652–663.

Gelfand, N.; Mitra, N. J.; Guibas, L. J.; and Pottmann, H. 2005. Robust global registration. In *Eurographics Symposium on Geometry Processing*, volume 2, 5–15.

Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; and Bennamoun, M. 2020. Deep learning for 3D point clouds: A survey. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 43(12): 4338–4364.

Huang, Q.-X.; Flory, S.; Gelfand, N.; Hofer, M.; and Pottmann, H. 2006. Reassembling fractured objects by geometric matching. *ACM Transactions on Graphics*, 25(3): 569–578.

Idram, I.; Bintara, R. D.; Lai, J.-Y.; Essomba, T.; and Lee, P.-Y. 2019. Development of mesh-defect removal algorithm to enhance the fitting of 3D-printed parts for comminuted bone fractures. *Journal of Medical and Biological Engineering*, 39: 855–873.

Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning*, 448–456.

Jaiswal, P.; Huang, J.; and Rai, R. 2016. Assembly-based conceptual 3D modeling with unlabeled components using probabilistic factor graph. *Computer-Aided Design*, 74: 45–54.

Jones, R. K.; Barton, T.; Xu, X.; Wang, K.; Jiang, E.; Guerrero, P.; Mitra, N. J.; and Ritchie, D. 2020. Shapeassembly: Learning to generate programs for 3D shape structure synthesis. *ACM Transactions on Graphics*, 39(6): 1–20.

Kalogerakis, E.; Chaudhuri, S.; Koller, D.; and Koltun, V. 2012. A probabilistic model for component-based shape synthesis. *ACM Transactions on Graphics*, 31(4): 1–11.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Li, J.; Niu, C.; and Xu, K. 2020. Learning part generation and assembly for structure-aware shape synthesis. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 34, 11362–11369.

Li, J.; Xu, K.; Chaudhuri, S.; Yumer, E.; Zhang, H.; and Guibas, L. 2017. Grass: Generative recursive autoencoders for shape structures. *ACM Transactions on Graphics*, 36(4): 1–14.

Li, S.; Liu, M.; and Walder, C. 2022. EditVAE: Unsupervised Parts-Aware Controllable 3D Point Cloud Shape Generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1386–1394.

Li, Y.; Mo, K.; Duan, Y.; Wang, H.; Zhang, J.; Shao, L.; Matusik, W.; and Guibas, L. 2023. Category-Level Multi-Part Multi-Joint 3D Shape Assembly. *arXiv preprint arXiv:2303.06163*.

Li, Y.; Mo, K.; Shao, L.; Sung, M.; and Guibas, L. 2020. Learning 3D part assembly from a single image. In *Proceedings of the European Conference on Computer Vision*, 664–682.

Loshchilov, I.; and Hutter, F. 2016. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.

Lu, J.; Sun, Y.; and Huang, Q. 2023. Jigsaw: Learning to Assemble Multiple Fractured Objects. In *Proceedings of the Advances in Neural Information Processing Systems*.

Mo, K.; Guerrero, P.; Yi, L.; Su, H.; Wonka, P.; Mitra, N. J.; and Guibas, L. J. 2020a. StructEdit: Learning structural shape variations. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 8859–8868.

Mo, K.; Wang, H.; Yan, X.; and Guibas, L. 2020b. PT2PC: Learning to generate 3D point cloud shapes from part tree conditions. In *Proceedings of the European Conference on Computer Vision*, 683–701. Springer.

Narayan, A.; Nagar, R.; and Raman, S. 2022. RGL-NET: A recurrent graph learning framework for progressive part assembly. In *Proceedings of the Winter Conference on Applications of Computer Vision*, 78–87.

Papaioannou, G.; and Karabassi, E.-A. 2003. On the automatic assemblage of arbitrary broken solid artifacts. *Image and Vision Computing*, 21(5): 401–412.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Proceedings of the Advances in Neural Information Processing Systems*, 32.

Qi, C. R.; Su, H.; Mo, K.; and Guibas, L. J. 2017a. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the conference on Computer Vision and Pattern Recognition*, 652–660.

Qi, C. R.; Yi, L.; Su, H.; and Guibas, L. J. 2017b. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the Advances in Neural Information Processing Systems*.

Ruiz-Correa, S.; Shapiro, L. G.; and Melia, M. 2001. A new signature-based method for efficient 3-d object recognition. In *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, 1–8.

Rusu, R. B.; Blodow, N.; and Beetz, M. 2009. Fast point feature histograms (FPFH) for 3D registration. In *Proceedings of the InternationalConference on Robotics and Automation*, 3212–3217.

Salti, S.; Tombari, F.; and Di Stefano, L. 2014. SHOT: Unique signatures of histograms for surface and texture description. *Computer Vision and Image Understanding*, 125: 251–264.

Sarlin, P.-E.; DeTone, D.; Malisiewicz, T.; and Rabinovich, A. 2020. SuperGlue: Learning feature matching with graph neural networks. In *Proceedings of the conference on Computer Vision and Pattern Recognition*, 4938–4947.

Schor, N.; Katzir, O.; Zhang, H.; and Cohen-Or, D. 2019. Componet: Learning to generate the unseen by part synthesis and composition. In *Proceedings of the International Conference on Computer Vision*, 8759–8768.

Sellán, S.; Chen, Y.-C.; Wu, Z.; Garg, A.; and Jacobson, A. 2022. Breaking Bad: A Dataset for Geometric Fracture and Reassembly. In *Proceedings of the conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Toler-Franklin, C.; Brown, B.; Weyrich, T.; Funkhouser, T.; and Rusinkiewicz, S. 2010. Multi-feature matching of fresco fragments. *ACM Transactions on Graphics*, 29(6): 1–12.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Proceedings of the Advances in Neural Information Processing Systems*, 30.

Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S. E.; Bronstein, M. M.; and Solomon, J. M. 2019. Dynamic graph cnn for learning on point clouds. *ACM Transactions On Graphics*, 38(5): 1–12.

Wei, L.; Yu, W.; Li, M.; and Li, X. 2011. Skull assembly and completion using template-based surface matching. In *Proceedings of the Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission*, 413–420.

Willis, A. R.; and Cooper, D. B. 2004. Bayesian assembly of 3D axially symmetric shapes from fragments. In *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition*, volume 1, 82–89.

Wu, J.; Zhang, C.; Xue, T.; Freeman, B.; and Tenenbaum, J. 2016. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. *Proceedings of the Advances in Neural Information Processing Systems*, 29.

Wu, R.; Tie, C.; Du, Y.; Zhao, Y.; and Dong, H. 2023. Leveraging SE(3) Equivariance for Learning 3D Geometric Shape Assembly. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 14311–14320.

Wu, R.; Zhuang, Y.; Xu, K.; Zhang, H.; and Chen, B. 2020. PQ-NET: A generative part seq2seq network for 3D shapes. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 829–838.

Yew, Z. J.; and Lee, G. H. 2022. REGTR: End-to-end point cloud correspondences with transformers. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 6677–6686.

Yin, Z.; Wei, L.; Li, X.; and Manhein, M. 2011. An automatic assembly and completion framework for fragmented skulls. In *Proceedings of the International Conference on Computer Vision*, 2532–2539.

Zhan, G.; Fan, Q.; Mo, K.; Shao, L.; Chen, B.; Guibas, L. J.; Dong, H.; et al. 2020. Generative 3D part assembly via dynamic graph learning. *Proceedings of the Advances in Neural Information Processing Systems*, 33: 6315–6326.

Zhang, K.; Yu, W.; Manhein, M.; Waggenspack, W.; and Li, X. 2015. 3D fragment reassembly using integrated template guidance and fracture-region matching. In *Proceedings of the International Conference on Computer Vision*, 2138–2146.

Zhang, R.; Kong, T.; Wang, W.; Han, X.; and You, M. 2022. 3D Part Assembly Generation With Instance Encoded Transformer. *IEEE Robotics and Automation Letters*, 7(4): 9051–9058.

Zhong, Y. 2009. Intrinsic shape signatures: A shape descriptor for 3D object recognition. In *Proceedings of the International Conference on Computer Vision workshops*, 689–696.