High-accuracy Fractured Object Reassembly under Arbitrary Poses

Qun-Ce Xu
Tsinghua University
BNRist, Tsinghua University, Beijing 100084, China
quncexu@tsinghua.edu.cn

Yan-Pei Cao
VAST
U-Center Tower C, 28, Beijing, 100084,China
caoyanpei@gmail.com

Weihao Cheng
ARC Lab
Tencent PCG, 100084, China
airrafer@hotmail.com

Tai-Jiang Mu
Tsinghua University
BNRist, Tsinghua University, Beijing 100084, China
taijiang@tsinghua.edu.cn

Ying Shan
ARC Lab
Tencent PCG, 100084, China
yingsshan@tencent.com

Yong-Liang Yang
University of Bath
Department of Computer Science, University of Bath, UK
y.yang@cs.bath.ac.uk

Shi-Min Hu
Tsinghua University
BNRist, Tsinghua University, Beijing 100084, China
shimin@tsinghua.edu.cn

Abstract

Fractured object reassembly is a challenging problem in computer vision with broad applications in industrial manufacturing, archaeology, etc. Traditional procedural methods rely on local shape descriptors or geometric registration, which are not always robust given the small fraction of fracture faces among fragments. While recent deep learning based methods have shown promising results by incorporating semantic information, they often assume that input fragments are aligned in a canonical pose. In this paper, we propose an approach that eliminates this implicit assumption by predicting shape reassembly results under arbitrary poses. Instead of directly regressing the canonical fragment poses, our neural network predicts the complementary shape of one input fragment given the other fragment to expand potential overlapping areas for later registration.

Keywords: Shape reassembly, Geometry Processing, Deep learning.

1. Introduction

Reassembling fractured objects from ancient fragments or broken pieces is a crucial task to restore their original shapes and functionalities. Conventional assembly relies on extensive labor work with specialized tools, which can be time-consuming, expensive, and error-prone. With the development of 3D digitization techniques, there has been a growing interest in developing automated assembly methods that can improve efficiency and accuracy while reducing costs. The core problem of automated fractured shape reassembly is how to effectively assemble two fragments through pairwise shape matching and alignment, as multiple fragments can be treated as concatenating a series of pairwise fragment assemblies.

However, it is still highly challenging even for assembling two fragments. First, unlike aligning two temporally-coherent scans (e.g., captured from a hand-held depth camera), the two fragments are usually captured and reconstructed separately and thus are in irrelevant poses. Hence local registration methods based on ICP [3] can easily fall

into local minimums. Second, there is often limited overlap between two fragments, thus the redundant and exclusive information can largely affect the performance of global registration methods such as RANSAC [10]. To address the above challenges, several methods were proposed to specifically handle fractured object reassembly [33, 32, 17, 6]. The key is how to identify and align the overlapping area of the two fragments, which heavily depends on the hand-crafted features used for shape segmentation and matching, and thus may not generalize well to unseen objects or fragments. Recent deep learning-based registration methods can perform local [44] and global [5] registration with learned features that are more generalizable. But they still suffer from the same problem due to the above fragment characteristics. A very recent work [9] proposed to directly regress the poses of fragments to assemble two fragments. However, it assumed the canonical pose of the entire object, thus cannot handle arbitrary relative poses between two fragments. Also, the accuracy of the assembly can easily be affected as it fully depends on the regressed poses while the overlap between two fragments is not taken into account.

In this paper, we propose a novel method that takes advantage of both learning-based and geometry-based approaches, aiming at robust and accurate assembly results. Given a pair of arbitrarily posed fragments to be assembled, in the first stage, our method utilizes an effective neural network module, which employs a carefully designed attention mechanism for feature correlation between two fragments to predict their complements and the fracture face points under their respective poses. In this way, we can 'directly' correlate the two fragments with arbitrary poses, which is much more flexible than predicting the canonical poses of the two fragments as in [9]. Note that predicting the complementary shapes can expand potential overlapping areas, which are beneficial for the global alignment of the two fragments if the fracture faces in between are few or even incompatible due to unexpected erosion or damage. In the second stage, benefiting from large potential overlapping provided by the predicted complements and fracture surface points, we leverage a geometric alignment module to precisely assemble the two fragments in an effective coarse-to-fine manner. We first apply RANSAC to coarsely align the two fragments (along with their complements), followed by locally aligning them based on ICP acting on points sampled from their fracture faces to refine the pose.

We have extensively evaluated the proposed method on the public Breaking-Bad dataset [39]. The experimental results and ablation studies demonstrate the effectiveness of our method for accurately assembling fractured objects and its superiority over competing methods.

The major contributions of our work can be summarized as the following:

- a hybrid method for high-precision fragment assembly which can handle arbitrary relative poses between two fragments instead of predicting their canonical poses, yielding better generalization across different shapes.

- a shape prediction network that simultaneously predicts the complement and fractured points of a fragment, providing more potential overlapping regions for better fragment registration.

- a geometric alignment module that precisely aligns fragments in a coarse-to-fine manner, achieving state-of-the-art performances.

2. Related Work

Procedure-based fractured shape reassembly. How to automatically reassemble a fractured object from its fragments has been actively studied in the geometry processing field. The overall procedure typically involves fragment segmentation, feature selection, and surface matching, in order to identify and align compatible fragments. Early works such as [32, 33] tackled the problem of solid 3D shape assembly. Later, as the assembly tasks became more complex, several methods [22, 17, 6] were presented to handle the assembly of more intricate shapes. Recently, despite the development of deep learning techniques in 3D shape processing, [34, 27] continued to explore geometric approaches for fragment reassembly. Although demonstrating success in different scenarios, procedural methods are usually time-consuming given the complexity of the procedure. Moreover, they rely on hand-crafted features which are hard to define and cannot easily generalize to unseen object categories with different shape characteristics.

Learning-based shape assembly. In recent years, the increasing availability of 3D shapes has enabled their usage for machine learning. Various representations such as point clouds, voxels, meshes, and implicit functions have been explored in the field of geometric learning [49]. Existing works such as [51, 20, 31] have utilized deep learning on various 3D shape datasets [7, 48, 30] to solve shape assembly problems. Moreover, specific methods targeting CAD models such as [46, 14] have also been proposed. Implicit function-based shape completion methods such as [23, 25, 24] emerged lately for shape restoration. Unlike previous works focusing more on assembling semantic shapes with complete geometry, [9] demonstrated the potential of using a learning-based approach for fractured shape assembly. Recently, several more advanced learning-based methods have been proposed for the assembly of fractured objects. One such method is Jigsaw [28], which introduces a transformer-based approach utilizing a novel geometry descriptor. Additionally, the work [47] incorporates the concept of SE(3) equivariance into the network architecture, thereby enhancing the quality of the assembled results. Furthermore, a unified diffusion-

based model has been proposed in the paper [38], capable of reassembling both 2D and 3D fractured objects.

We are also interested in assembling a fractured shape, while we do not have the canonical pose assumption of the shape and thus support arbitrary poses. Meanwhile, our combination of learning-based and geometry-processing-based approaches results in highly precise alignments.

Point cloud registration. Point cloud registration is widely used for aligning two or more point clouds [41]. Registration methods can be roughly categorized into (i) global registration where point clouds can have arbitrary poses, and (ii) local registration whose performance depends on initial poses. The former aims to provide robust initial poses while the latter targets achieving optimal poses with good initialization (e.g., from the former). The Random Sample Consensus (RANSAC) framework [10] and the derivatives therein [1, 29, 35] play an important role in global registration, especially when the input point clouds are contaminated by noises and outliers. For relatively smooth point clouds, feature-based methods are also commonly used to identify matched points with consistent hand-crafted features [19, 12, 36, 42]. Regarding local registration, the Iterative Closest Point (ICP) method [3, 8] and its variants such as [26, 11, 4] are well adopted. With the rise of deep learning, differentiable versions of RANSAC and ICP have been developed with learned features, such as those proposed in DSAC [5] and DCP [44], which are compatible with learning-based tasks. However, the small overlap between pairs of point clouds poses new challenges, which were recently addressed by incorporating ambient guidance fields [16], a learning-based approach attentive to overlap regions [18], or combining the registration and completion tasks [50]. Thanks to the shape prediction and interface labeling ability of our learning-based module, simple RANSAC and ICP can precisely align two input fragments in our geometry-processing module.

3. Methodology

Problem definition. Before elaborating on the details of our method, we first describe the problem settings. Considering a shape represented by a point cloud $P = \{p_i \in \mathbb{R}^3 | i = 1, ..., N\}$ sampled from the underlying mesh surface, it has been broken into two fragments P_a and P_b with fractured regions in arbitrary poses. Our goal is to find a seamless reassembly P_{ab} composed of P_a and P_b as accurately as possible to restore P.

Overview. As shown in Figure 1, our hybrid method consists of two major components: a shape prediction module for complementary and fracture surface points prediction, and a geometric registration module for shape alignment. In the shape prediction module,

Based on a specifically designed attention mechanism that effectively correlates P_a and P_b, the network learns to predict the fracture face points as well as the two complementary shapes P_{b^*} and P_{a^*} (for P_a and P_b, respectively), such that the combined shape P_{ab^*} composed of P_a and P_{b^*} well approximates the full shape P_{ab}, so as P_{ba^*} composed of P_b and P_{a^*}. More specifically, we have:

$$\begin{aligned} P_{ab^*} &= P_a \oplus P_{b^*} \\ P_{ba^*} &= P_b \oplus P_{a^*}, \end{aligned} \tag{1}$$

where $\oplus$ is the point cloud combination operator.

Next, we feed P_{ab^*} and P_{ba^*} into the geometric registration module to calculate the accurate transformation for relative pose estimation between the input fragments. The initial alignment between P_a and P_b is estimated by RANSAC using the predicted full shape P_{ab^*} and P_{ba^*}. The final alignment is optimized using ICP based on the fracture face points of P_a and P_b predicted by the previous the neural network module.

3.1. Shape Prediction for Fragments

Considering that the fraction of fracture faces among fragments is usually small, a sufficient amount of overlapping points should be provided before we can perform the registration for fragments. To achieve this, we propose to learn the complementary of a fragment, given the real one as a reference, and simultaneously predict the fracture surface points on the input fragments. We employ an encoder-decoder architecture to fulfill this goal, as depicted in Figure 1 (top row).

Point Cloud Feature Encoder. We first feed the point clouds P_a and P_b of the two input fragments separately into a shared point cloud feature backbone to obtain their initial features f_a and f_b. To balance performance and efficiency, we choose Dynamic Graph CNN (DGCNN) [45] as the point cloud feature backbone. f_a and f_b each have the dimension of 1024×128, where 1024 is the number of input points and 128 is the feature dimension per point.

The two input point clouds are highly correlated since they can be assembled into a full shape. However, the above initial point features have not been related to each other. Inspired by recent work [37] of feature matching, we employ a graph neural network (GNN) with carefully-designed attention mechanisms to enhance the initial features based on their correlations for better complementary shape and fracture surface points prediction.

Similar to [37], we adopt self-attention [43] based on self-edges to build relationships within each point cloud. Self-attention computes a weight for each feature in one input based on how well it matches with other features in the same input, allowing the network to capture relationships between different features within a point cloud. To handle the relationship between two different point clouds, we use cross-attention that builds edges between points in P_a and P_b. The message passing formulation can be referred

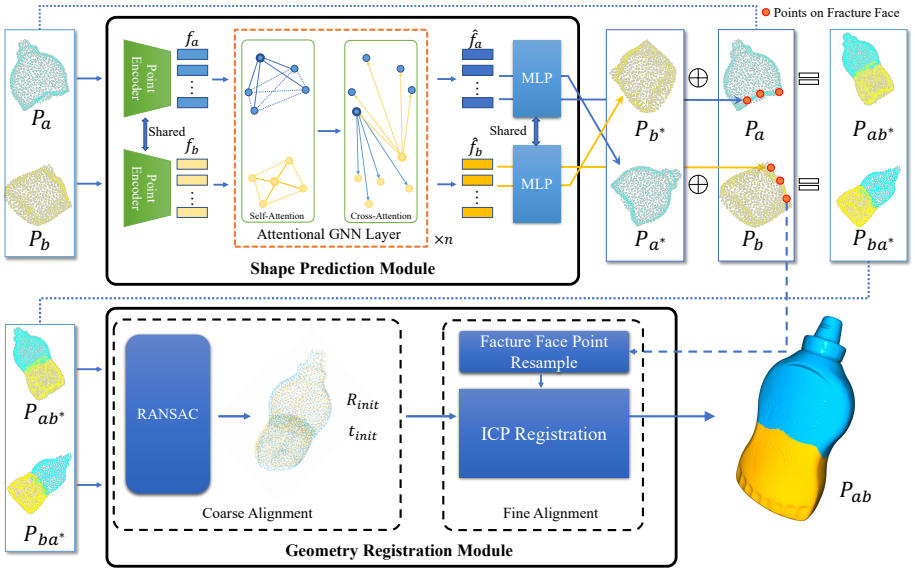

Figure 1. **Pipeline.** Our method mainly contains two components: a shape prediction module and a geometric registration module. The former consists of a point cloud encoder based on DGCNN, a graph network with a specifically designed attention mechanism, and an output layer based on MLP. Given two input fragments P_a and P_b, it learns to predict both the fracture face points and their complementary shapes P_{b^*} and P_{a^*}, resulting in combined shapes P_{ab^*} and P_{ba^*}, respectively. The geometric registration module takes P_{ab^*} and P_{ba^*} as input and employs RANSAC to achieve an coarse alignment $\{R_{init}, t_{init}\}$ between P_a and P_b, followed by ICP acting on the predicted fracture face points to compute a refined alignment for the final reassembly P_{ab}.

to [13, 2]. The multi-head attention [43] is also used to improve expressive ability. The multi-head attention mechanism allows the network to attend to multiple parts of the input simultaneously, which is useful for capturing complex relationships between features of the two point clouds. The alternating usage of self- and cross-attention can make the network sufficiently learn the geometric features and the transformations among each other, resulting in correlated features $\hat{f}_a$ and $\hat{f}_b$ for P_a and P_b, respectively. Different from previous work [37] applying on images, we use the features $\hat{f}_a$ and $\hat{f}_b$ from GNN layers to predict shapes rather than matching descriptors. In our experiments, using $n = 6$ GNN layers suits both memory and performance requirements.

3.1.1 Network Module Details

Point Cloud Encoder. Previous paragraph describes the usage of Dynamic Graph CNN (DGCNN) as the point cloud encoder. The encoder is designed with 5 EdgeConv (as denoted in [45]) convolution layers followed by an Instance Norm layer and a LeakyReLU layer with a negative slope of 0.2 as the activation function. The number of filters in each layer are [64, 64, 128, 256, 128]. The feature dimension is set to 128, and the k-NN parameter in graph feature integration is set to 16.

Attentional Graph Neural Network. The transformer consists of 6 attention-based GNN layers with self- and cross-attention mechanisms. Each GNN layer includes an attentional propagation module containing 4 multi-head attention layers and an MLP with three hidden layers of dimensions 128, 512, and 128. An Instance Norm is applied following the propagation module. During forward propagation, self- and cross-attention are alternately executed in this part. The attention layers are based on the approach described in [37], and the detailed settings of these layers can be found therein.

Multi-Layer Perceptron. In the last part of the network, an MLP is used to predict the offset of points instead of regressing the pose, which is different from previous works such as [9, 51]. The MLP has 3 hidden layers of dimensions 256, 256, and 128. It maps the features from the GNN layers to 3D displacement vectors for the input point clouds P_a and P_b. The MLP also predicts the probability indicating whether a point lies on the fracture faces.

Complementary Shape and Fracture Points Prediction.

Instead of directly regressing the poses of shape components [9, 51], we learn the complementary shape P_{b^*} (or P_{a^*}) of the given shape P_a (or P_b) by predicting the displacement of each point in P_b (or P_a), and the fracture surface in-between by also predicting the key points therein. We use a Multi-Layer Perceptron (MLP) with 3 hidden layers (256, 256 and 128 neurons for each) to map $\hat{f}_a$ and $\hat{f}_b$ to the final predictions. Specifically, the MLP outputs a three-dimensional vector as the displacement for each input point of P_b and P_a to abtain P_{b^*} and P_{a^*}. The MLP also out-

puts a probability that indicates how possible a point lies in the fracture surface connecting the two input fragments. After combining P_{b^*} and P_{a^*} with the input fragments P_a and P_b, respectively, the generated point cloud (P_{ab^*} and P_{ba^*}) forms a shape under the given poses of P_a and P_b, not limiting to any canonical pose.

Loss functions. During training, we use the Mean Squared Error (MSE) loss $\mathcal{L}_{mse}$ and the geometry reconstruction loss $\mathcal{L}_{recon}$ for complementary shape prediction. More specifically, the MSE loss controls the shape of the predicted complementary point cloud according to the ground truth shape:

$$\mathcal{L}_{mse} = \mathcal{L}^a_{mse} + \mathcal{L}^b_{mse}, \tag{2}$$

where

$$\mathcal{L}^x_{mse} = \frac{1}{N_x}\sum_{i=1}^{N_x}(p^i_{x^*} - p^i_x)^2. \tag{3}$$

Here $x \in \{a, b\}$ and N_x is the number of points of P_a or P_b. $p^i_{x^*}$ refers to the i-th point in P_{x^*}, while p^i_x denotes the corresponding point in the ground truth point cloud P_x.

The reconstruction loss L_{recon} regularizes the distances D between every pair of points in a single point cloud. For a point cloud with N points, D can be represented as a distance matrix:

$$D = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1N} \\ d_{21} & d_{22} & \cdots & d_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ d_{N1} & d_{N2} & \cdots & d_{NN} \end{bmatrix}, \tag{4}$$

where d_{ij} is the Euclidean distance between two points p_i and p_j. L_{recon} is defined as the sum of two losses L^a_{recon} and L^b_{recon}:

$$\mathcal{L}_{recon} = \mathcal{L}^a_{recon} + \mathcal{L}^b_{recon}, \tag{5}$$

where $L^x_{recon}, x \in \{a, b\}$ is defined as the $L1$ norm of the difference between two distance matrices:

$$\mathcal{L}^x_{recon} = ||D_{P_{x^*}} - D_{P_x}||_1. \tag{6}$$

Here D_{P_x} represents the distance matrix for the ground truth point cloud P_x, and $D_{P_{x^*}}$ represents the distance matrix for the reconstructed point cloud P_{x^*}.

Additionally, the Binary Cross-Entropy (BCE) loss L_{label} is used to compute the error of fracture surface points prediction:

$$\mathcal{L}_{label} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(y^*_i) + (1 - y_i)\log(1 - y^*_i)] \tag{7}$$

Here, N is the total number of points in the input fragments, $y_i \in \{0, 1\}$ is the ground truth label (1 for points on the fracture surface otherwise 0) for the i-th point, and y^*_i is the predicted probability of the i-th point.

Finally, the total loss function used to train our neural network is the sum of the above losses by treating them with equal contribution:

$$\mathcal{L} = \mathcal{L}_{mse} + \mathcal{L}_{recon} + \mathcal{L}_{label}. \tag{8}$$

3.2. Geometric Registration Module

Given P_{ab^*} and P_{ba^*} with labeled points on the fracture faces connecting P_a and P_b, we explicitly process their geometries to accurately align P_a and P_b in a coarse-to-fine manner to obtain a seamless assembly, as shown in Figure 1 (bottom row).

3.2.1 Coarse Alignment using Predicted Shapes

Given P_a and P_b (so as P_{ab^*} and P_{ba^*}) are in arbitrary poses, we first employ the classic Random Sample Consensus (RANSAC) algorithm to obtain a coarse alignment between P_{ab^*} and P_{ba^*}, each of which comprises 2048 points.

In our experiment, we randomly select two 4-point subsets from P_{ab^*} and P_{ba^*}. Suppose P_{ab^*} (including P_a) serves as the reference, then based on the bijective mapping between the two subsets, a candidate pose of P_{ba^*} (including P_b) can be explicitly deduced to align with P_{ab^*}. The preference of the candidate pose is measured by the number of inliers (well-matched points) after aligning P_{ab^*} and P_{ba^*} based on it. This procedure is repeated for a fixed number of iterations (100 in our setting) or until a satisfactory alignment is found. The pose (including a rotation matrix R_{init} and a translation vector t_{init}) yielding the maximal number of inliers is selected to produce the final coarse alignment between P_{ab^*} and P_{ba^*} (also applies for P_a and P_b).

Note that the coarse alignment can only roughly align P_a and P_b due to two reasons: 1) the predicted complementary parts P_{b^*} and P_{a^*} would not be perfect, and 2) the randomized nature of RANSAC would not guarantee the global optimum. Therefore, we further refine the alignment as follows.

3.2.2 Alignment Refinement using Predicted Fracture Points

Although the coarse alignment is not optimal, it already provides an initial pose that is good enough for further refinement. As our last step of shape assembly, we employ the Iterative Closest Point (ICP) method to refine the initial pose. In our implementation, we apply conventional ICP to the predicted fracture surface points of P_a and P_b by iteratively identifying the closest points as the corresponding points and minimizing their point-to-point distances.

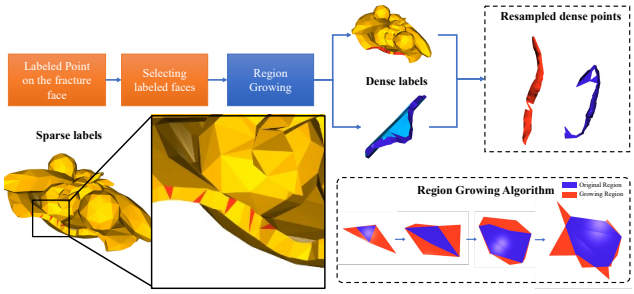

Figure 2. Dense points are sampled based on the predicted fracture surface points and their associated mesh faces. Region growing is applied to fill the gaps between mesh faces and enable dense sampling.

However, P_a and P_b each contains 1024 points thus only a limited number of points were sampled at the fracture surfaces whose areas are usually small w.r.t. the entire shape. Hence closest points are not good approximations of corresponding points. To resolve this, we up-sample fracture surface points with the help of the original mesh surfaces and apply ICP to align dense points, yielding better alignment performance. As demonstrated in Figure 2, in the up-sampling process, we first extract the underlying mesh faces on which the predicted fracture surface points lie. For sparsely sampled points, the extracted mesh faces are likely to be disconnected or have holes among them. We then apply region growing on the extracted mesh faces to form a disk-like interface patch. Finally, we up-sample points (1k in our setting) from the fracture patch and perform ICP using the densely sampled points, which allows better point correspondences thus further improving the accuracy of the alignment.

4. Experiments and Results

This section demonstrates our experimental results. We first present how we prepare data to evaluate fractured object reassembly from a pair of fragments followed by quantitative evaluation metrics. We then demonstrate the advantage of our method in terms of accuracy and robustness by comparing it with both baseline and state-of-the-art methods. Finally, we validate the effectiveness of different components of our method through an ablation study.

4.1. Data Preparation

We conduct experiments on the emerging Breaking Bad dataset [39], which consists of more than 20 object categories and features 2,547 meticulously crafted 3D models. Each object category has numerous instances that vary in scale and pose. The "Everyday" model set in the Breaking Bad dataset [39] was used in our work. It consists of 20 categories of objects, including 'BeerBottle', 'Bottle', 'Bowl', 'Cookie', 'Cup', 'DrinkBottle', 'DrinkingUtensil', 'Mirror', 'Mug', 'PillBottle', 'Plate', 'Ring', 'Spoon', 'Statue', 'Teacup', 'Teapot', 'ToyFigure', 'Vase', 'WineBottle', 'WineGlass'. The scripts provided by [39] were used to extract fractured objects as original mesh .obj files for point sampling. And we focused on fractured objects each with two pieces of fragments.

The 3D models in the Breaking Bad dataset are initially man-made and represented as surface meshes. We leverage all the pairwise fragments with different fractured types, even those with a low cut-off ratio (the ratio of the fragment over the complete object) in a total of 13,954 pairs. We use a 60/20/20 scheme for the training/validation/test split. To convert the data from surface meshes to more general point clouds, we first conducted a dense sampling to generate 10K points from each mesh. We then down-sample it to 1024 points through farthest point sampling. For each pair of fragments, we normalize them by the longer bounding box diagonal. During the training phase, we sample rotation matrices and translation vectors to generate random poses for each pair of fragments. When applying random transformation for each fragment, its complementary fragment is also transformed "implicitly" to serve as ground truth, but not as input to our method. Note that the models provided by the Breaking Bad dataset are all in canonical poses by default. We also create a "non-canonical" dataset by randomly perturbing the canonical poses to evaluate the influences of different methods.

4.2. Evaluation Metrics

To quantitatively evaluate our results, we follow [44, 9] to measure the error between the predicted rotation R^* and translation t^* and the ground truth ones. More specifically, we compute the rotation error E_r and translation error E_t as follows:

$$\begin{aligned} E_r(R^*) &= ||R_{gt}^T R^* - I||_F \\ E_t(t^*) &= ||t_{gt} - t^*||_2 \end{aligned} \tag{9}$$

Here, $\{R_{gt}, t_{gt}\}$ and $\{R^*, t^*\}$ denote the ground truth and estimated transformation, respectively. I is the identity matrix. $||\cdot||_F$ and $||\cdot||_2$ denote matrix Frobenius norm and vector $L2$ norm, respectively.

4.3. Implementation Details

Our model was implemented using Jittor [15] and trained using a small batch size of 4 on a Linux server with an Intel Xeon Sliver 4210 CPU and a single TITAN RTX GPU with 24GB of memory. We used the Adam optimizer [21] with a learning rate of 0.0001 to train the network. In terms of running time performance, the network is designed to be lightning-fast and it typically takes only seconds to infer the shape predictions and labels for each pair with 1,024 points for each fragment. The geometric registration usually takes $0.5 \sim 1.0$ minute, mainly spent on point sampling.

Methods	Canonical Dataset		Non-canonical Dataset	
	$E_r(\times 10^{-3}) \downarrow$	$E_t(\times 10^{-3}) \downarrow$	$E_r(\times 10^{-3}) \downarrow$	$E_t(\times 10^{-3}) \downarrow$
DCP [44]	1390.69	416.67	1457.24	362.45
ICP [3]	1715.98	403.01	1899.88	451.33
CTF-Net [50]	884.79	933.52	940.11	952.33
PREDATOR [18]	458.70	267.11	432.81	245.62
DGL [51]	105.22	94.78	234.31	110.13
JIGSAW [28]	18.11	23.61	60.19	64.24
SE(3)-Equiv [47]	22.98	26.34	46.72	53.52
Ours - Regressor	10.16	21.40	42.56	49.98
Ours	**8.95**	**18.04**	**9.47**	**20.42**

Table 1. Quantitative comparison of reassembly accuracy on the Breaking Bad dataset against baseline methods.

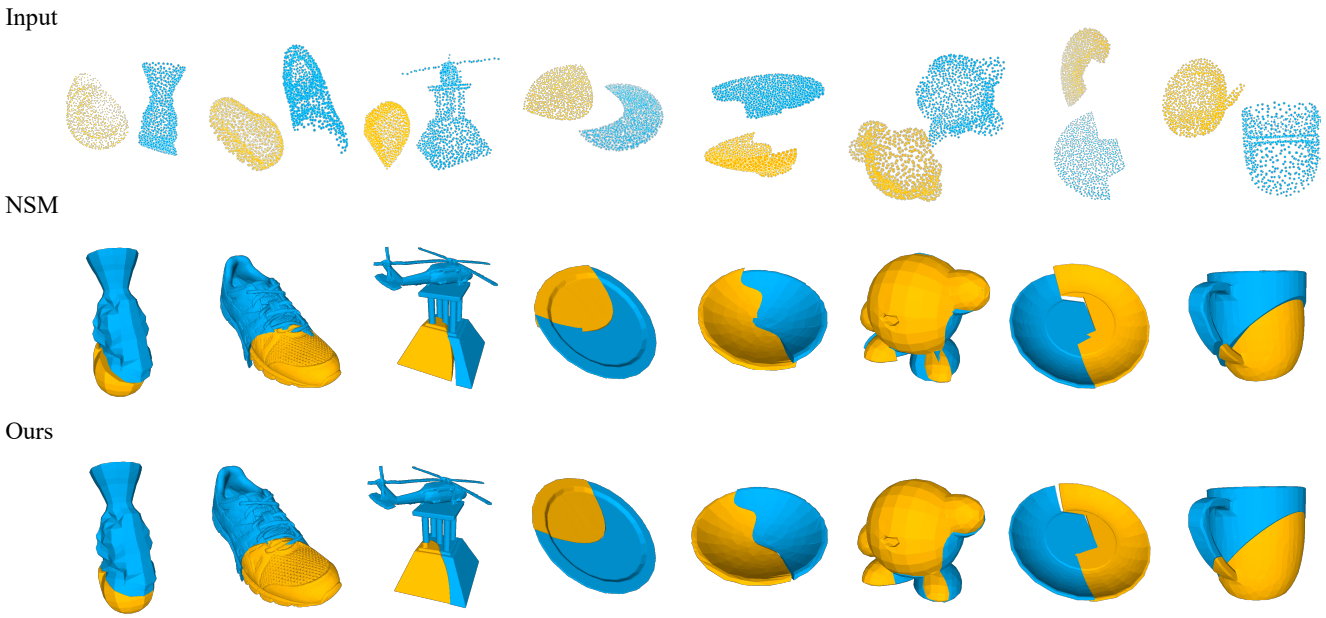

Figure 3. Visual comparison with Neural Shape Mating [9].

4.4. Comparison with other methods

We compared our method with both traditional ICP [3] and state-of-the-art learning-based methods, including DCP [44], PREDATOR [18], and DGL [51]. The quantitative comparison results are listed in Table 1, which demonstrates significant superiority of our method in both rotation and translation errors compared to other methods. Qualitative results in Figure 11 show that our method outperforms the other methods on different types of models.

Traditional ICP relies heavily on the initial pose and the availability of overlapped regions and performs the worst for fragment assembly among competing methods. DCP is a deep learning based ICP method [3]. It also heavily relies on initial poses and point correspondences, which would be challenging to obtain in our setting with arbitrary fragment poses. Similarly, PREDATOR works on low-overlap point cloud registration, whereas our task can be more challenging since the only overlapped region is on fractured surfaces. DGL utilizes a GNN to infer relationships between 3D parts and directly predicts their poses. This may not be effective for accurate pairwise fragment alignment.

We also compared with a modified version of our method, dubbed "Ours - Regressor," which replaced the shape prediction design with pose regression similar to [51, 9]. However, when objects in the dataset are not canonically posed for training, the regressor tends to perform poorly in predicting rotations. One possible reason is that pose prediction may be influenced by implicit semantic information such as canonical poses, which can affect the generalization performance under arbitrary poses. In contrast, our relative

pose learning method focuses on the information between object fragments rather than global semantics, making it more robust to challenging real-world data.

Comparison to completion-and-registration [50]. Our method shares some common designs with [50] in terms of using completion to facilitate alignment. The main difference is that we aim at a different task - fractured object assembly and our method also predicts fractured surface points to further refine the alignment, achieving accurate reassembly results. This is possible in our task as points on the fractured surface are generated based on fracture simulation, but not object modeling as those sampled from the original object. Therefore, directly applying the previous completion-and-registration method to our task would lower the performance. We compare with [50] by re-training their model using our dataset. The rotation error $E_r(\times 10^{-3})$ and translation error $E_t(\times 10^{-3})$ are (8.95, 18.04) for our method and (884.79, 933.52) for [50], showing the superiority of our work. Note that the training in [50] further requires the ground-truth missing part that is not present in either P_a or P_b, thus cannot generalize well to our task (assumes no missing part).

In addition to comparing with the method in [50], we also compared our approach with the state-of-the-art (SOTA) methods [28] and [47]. The quantitative and qualitative results demonstrate our method's superiority, which stems from the advanced geometry registration module. It is worth noting that [47] outperforms [28] on non-canonical datasets because it addresses the rotation problem through SE(3) equivalence to some extent.

We also attempted to compare with the recent NSM [9]. Unfortunately, this is not feasible without access to the pre-trained model or the datasets used for training. As an alternative, we provide a visual comparison in Figure 3 based on the example results shared by the authors. It can be seen that our method can further improve the accuracy of the alignment as witnessed by the visually seamless overlapping area in the final reassembly.

Moreover, our method demonstrates effectiveness in handling different relative poses. When various non-canonical poses are used as input, our proposed approach achieves satisfactory assembly results.(see Fig. 4)

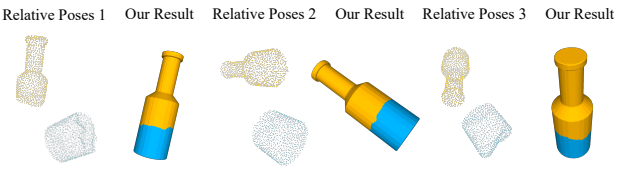

Figure 4. Qualitative results of different relative poses.

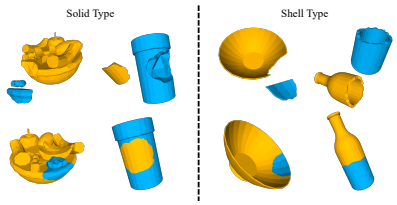

Figure 5. Qualitative examples of the solid type and shell type.

4.5. Evaluation on Solid/Shell Shape

Fractures can be categorized into two types, namely 'solid' and 'shell', based on their shapes and breaking patterns (see Fig. 5). We have specifically evaluated each fracture type, and the results are presented in Table 2 (upper part). It is worth noting that the number of solid and shell fractures in our dataset is not balanced, making it difficult to train using separate shapes for each type. To overcome this, we have trained our model on all fracture shapes and tested it on randomly selected subsets of 100 cases for each type. Due to the inherently lower contact area of fracture faces in shell fractures, fewer points are generated on these fracture faces compared to solid fractures. This can pose a challenge in achieving accurate geometry registration, thereby affecting the overall performance of the model.

4.6. Evaluation on Noisy Point Clouds

In practical applications, point cloud data acquired from depth sensors are often affected by noise. To evaluate the impact of noise on the performance of our model, we augmented our dataset by adding Gaussian noises to the clean point clouds. Specifically, we added noise with a mean of 0.0 and a standard deviation of 0.05 to each point in the dataset. Table 2 (second last row) also shows the model's performance on the augmented dataset. It can be seen that the presence of noises only slightly degrades the performance of the model, demonstrating that our method is robust to noisy data (see also qualitative examples in Fig. 6). The performance drop is mainly due to the impact on the probability predictions. Numerically, The predicted accuracy of the label from network has decreased by about 10% with noisy input data.

Experiment Settings	Canonical Dataset		Non-canonical Dataset	
	$E_r(\times 10^{-3})\downarrow$	$E_t(\times 10^{-3})\downarrow$	$E_r(\times 10^{-3})\downarrow$	$E_t(\times 10^{-3})\downarrow$
Original	8.95	18.04	9.47	20.42
Original + test on solid shapes only	8.34	17.58	9.21	18.67
Original + test on shell shapes only	16.57	18.61	18.84	21.22
Noisy Data Reassembly	10.72	20.90	12.53	22.48
Unseen Category Reassembly	45.28	29.33	53.71	35.98

Table 2. Quantitative results of additional experiments with different settings.

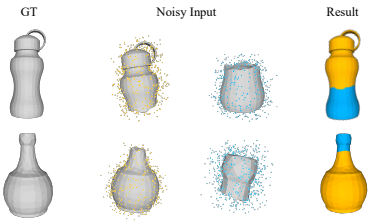

Figure 6. Qualitative examples of reassembling noisy point clouds.

4.7. Generalization for Unseen Category

We also conducted experiments to validate the generalization of our method. We tested on the 'Bottle', 'Plate', 'Bowl', and 'Cup' categories based on training using the remaining 16 categories. Experiment results can be found in Table 2 (last row). Note that our network design is based on relative poses, and focuses more on geometry-related information rather than global semantic information. Compared with the training setting with all categories, the performance degrades only moderately (comparable to NSM [9]). Our training data includes information on both the shape of fractures and the types of cuts present on their faces. While some categories in our training data may have similar shapes to the test data, our results suggest that the accuracy of point label predictions on fracture faces is influenced not only by their shape but also by the unseen types of fracture faces. The performance degradation we observed is likely due to multiple factors, and it is possible that the initial transformation from RANSAC is affected by the poor quality of the generated shapes. Moreover, incorrect label prediction may cause issues with the sampling of dense points before ICP, affecting the registration performance of ICP.

We mainly focus on the recent fractured object reassembly task [9, 39], which aims to reassemble fragmented parts with incomplete and irregular shapes, rather than composing semantic parts with complete geometry as in the previous shape assembly task [20, 31, 51]. Our evaluation is based on the latest benchmark [39]. The 20 categories therein well represent common fractured objects (e.g., glass objects, china tablewares, clay potteries, etc.), and are generated using a state-of-the-art physical simulation framework 'Breaking Good' [40]. Our method can even handle unseen, non-categorized objects in [39] (see Fig. 7). We understand that ShapeNet/PartNet contains more categories but it is more useful for the semantic shape assembly task, as the objects are not 'randomly' fractured but 'regularly' composed of semantic parts.

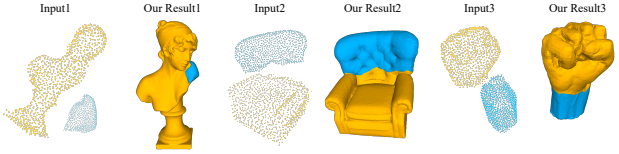

Figure 7. Additional unseen examples.

4.8. Ablation Study

We also conducted ablation experiments to validate the effectiveness of the design choices of our method. More specifically, we compared our full method with down-graded versions where we removed RANSAC, Fracture Points Resample, and ICP one at a time. First, we remove RANSAC from the geometric registration module, where P^*_{ab} and P^*_{ba} from the shape prediction module are directly used for ICP without a good initial alignment. Second, we only perform ICP on sparse key points predicted as fracture surface points without resampling dense points. Third, we remove ICP along with the dense point resampling process. The quantitative comparisons are detailed in Table 3, while the qualitative examples are shown in Figure 9.

We can see that our method achieves good performance even without the ICP module. This indicates that, benefiting from the large potential overlapping regions provided by our shape prediction network, the RANSAC module can already provide good initial pose. Reassembly with higher accuracy can be achieved with ICP on top of it. On the other hand, missing RANSAC or fracture point resample modules causes the usage of ICP in an ill-posed condition, leading to non-optimal registration results. Note that in Table 3, ICP is applied to two completed point clouds P^*_{ab} and P^*_{ba} from the shape prediction module, while the methods in Table 1 are given incomplete point clouds P_a and P_b. This also demonstrates the benefit of using reconstructed global shape information other than the limited overlapping region for reassembly.

It is indeed effective as demonstrated by the overall performance compared with previous methods, and also the ablation experiments (only applying ICP without RANSAC on the completed point clouds leads to promising results).

For more validation, we quantitatively compare the predicted point clouds using different losses with the ground truth based on Chamfer Distance: with reconstruction loss - 0.0725, without reconstruction loss (only MSE loss) - 0.0994. Qualitative examples are shown in Fig. 8.

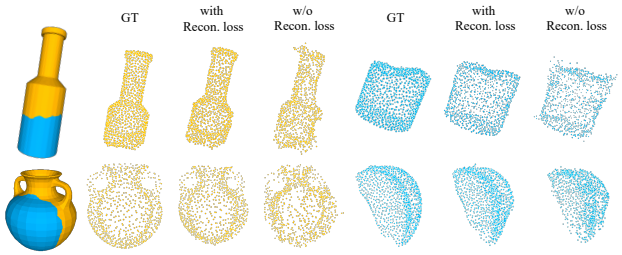

Figure 8. Ablation study on complementary shape prediction.

4.9. Failure case

Our method may fail in extreme cases with very small fractured surfaces (see Fig. 10), where the prediction of

Method	Canonical Dataset		Non-canonical Dataset	
	$E_r(\times 10^{-3}) \downarrow$	$E_t(\times 10^{-3}) \downarrow$	$E_r(\times 10^{-3}) \downarrow$	$E_t(\times 10^{-3}) \downarrow$
w/o RANSAC	380.81	80.32	391.20	92.12
w/o point resample	120.50	55.01	142.14	67.05
w/o ICP, w/o point resample	86.49	25.44	94.84	41.26
w/ full components	8.95	18.04	9.47	20.42

Table 3. Quantitative ablation study results of the geometric registration module.

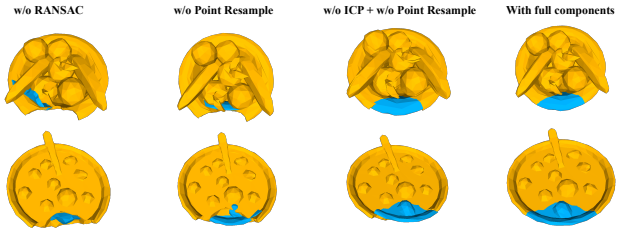

Figure 9. Qualitative examples of the ablation study.

fractured points is very challenging due to lacking samples and/or features within the limited region.

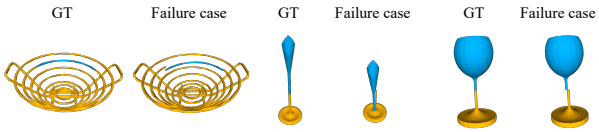

Figure 10. Example of failed cases.

5. Conclusion

We present a novel hybrid approach for the accurate reassembly of fractured objects with two fragments. Given two fragments with arbitrary poses as inputs, our neural network module first predicts the complementary shapes of the input fragments and the fracture face points. Next, we leverage the inferred geometric information for a subsequent geometric registration module to effectively initialize and accurately optimize the alignment of the two fragments, resulting in seamless and precise reassembly results. The experimental results demonstrate that our method significantly outperforms prior works.

Limitations and future work. Albeit the effectiveness of the geometric registration module, our approach is not based on an end-to-end architecture. In the future, we would like to devise an end-to-end model by making the geometric processing module differentiable and integrating it with the neural network module. Also, a promising research direction would be to apply our pairwise fragment assembly method to multiple fragments. Multiple fragments reassembling can be broken into multiple pairwise problems by adding pieces incrementally. But we are more interested in a 'comprehensive' approach where multi-piece features can be simultaneously learned for a 'global' assembly, as a future work.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (62220106003), the Research Grant of Beijing Higher Institution Engineering Research Center, Tsinghua-Tencent Joint Laboratory for Internet Innovation Technology, and UKRI grant CAMERA (No. EP/T022523/1).

References

[1] D. Aiger, N. J. Mitra, and D. Cohen-Or. 4-points congruent sets for robust pairwise surface registration. *ACM Trans. Graph.*, 27(3):1–10, 2008. 3

[2] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018. 4

[3] P. Besl and N. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. 1, 3, 7

[4] S. Bouaziz, A. Tagliasacchi, and M. Pauly. Sparse iterative closest point. *Computer Graphics Forum*, 32(5):113–123, 2013. 3

[5] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. Dsac-differentiable ransac for camera localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6684–6692, 2017. 2, 3

[6] B. J. Brown, C. Toler-Franklin, D. Nehab, M. Burns, D. Dobkin, A. Vlachopoulos, C. Doumas, S. Rusinkiewicz, and T. Weyrich. A system for high-volume acquisition and matching of fresco fragments: Reassembling theran wall paintings. *ACM transactions on graphics (TOG)*, 27(3):1–9, 2008. 2

[7] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[8] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. In *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pages 2724–2729 vol.3, 1991. 3

[9] Y.-C. Chen, H. Li, D. Turpin, A. Jacobson, and A. 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, 4, 6, 7, 8, 9

[10] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 2, 3

[11] N. Gelfand, L. Ikemoto, S. Rusinkiewicz, and M. Levoy. Geometrically stable sampling for the icp algorithm. In *Fourth International Conference on 3-D Digital Imaging and Modeling, 2003. 3DIM 2003. Proceedings.*, pages 260–267, 2003. 3

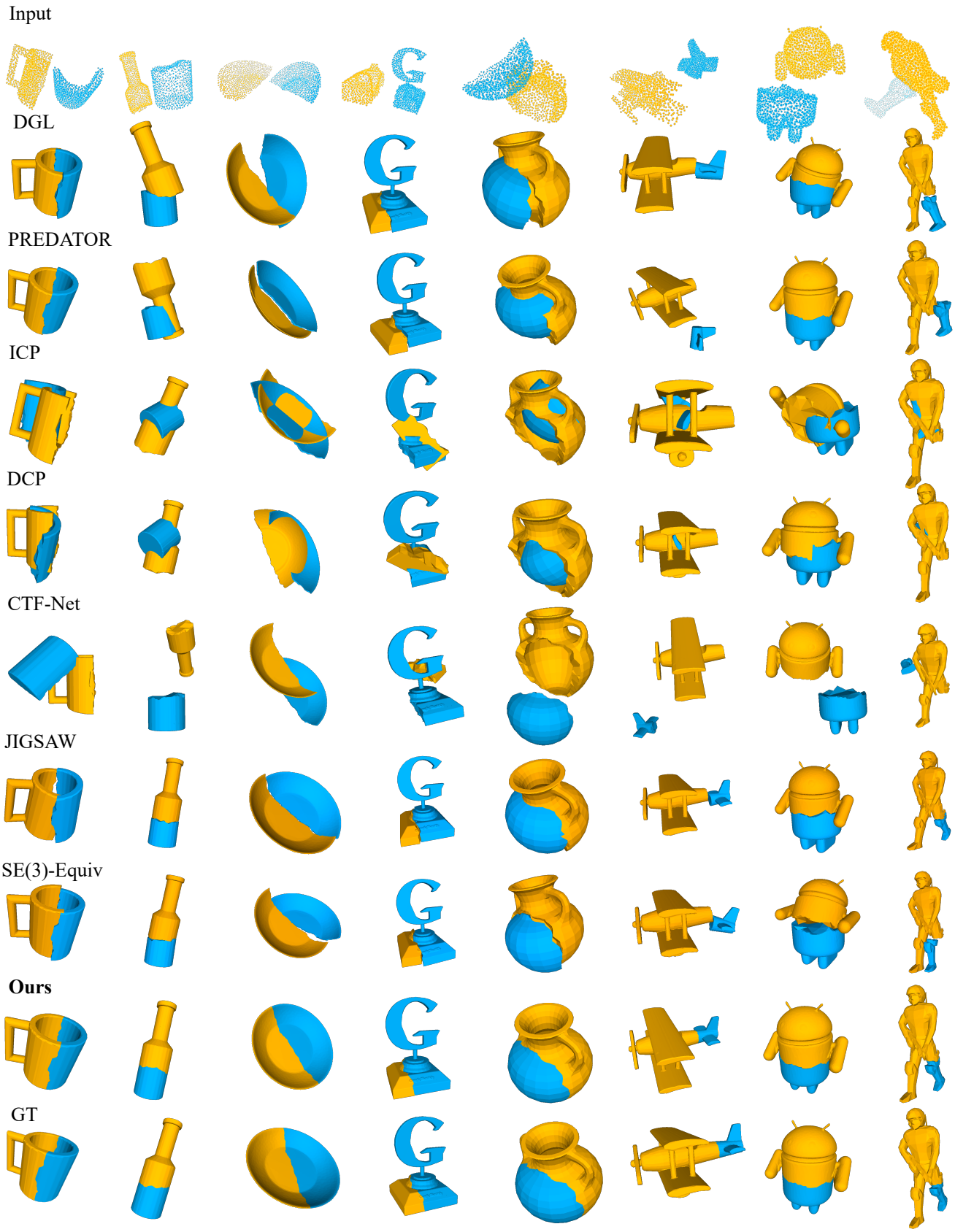

Figure 11. Qualitative comparisons between our method and SOTA methods.

[12] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann. Robust global registration. In M. Desbrun and H. Pottmann, editors, *Third Eurographics Symposium on Geometry Processing, Vienna, Austria, July 4-6, 2005*, volume 255 of *ACM International Conference Proceeding Series*, pages 197–206. Eurographics Association, 2005. 3

[13] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pages 1263–1272. PMLR, 2017. 4

[14] H. Guo, S. Liu, H. Pan, Y. Liu, X. Tong, and B. Guo. Complexgen: Cad reconstruction by b-rep chain complex generation. *ACM Transactions on Graphics (TOG)*, 41(4):1–18, 2022. 2

[15] S.-M. Hu, D. Liang, G.-Y. Yang, G.-W. Yang, and W.-Y. Zhou. Jittor: a novel deep learning framework with meta-operators and unified graph execution. *Science China Information Sciences*, 63:1–21, 2020. 6

[16] H. Huang, M. Gong, D. Cohen-Or, Y. Ouyang, F. Tan, and H. Zhang. Field-guided registration for feature-conforming shape composition. *ACM Trans. Graph.*, 31(6), nov 2012. 3

[17] Q. Huang, S. Flöry, N. Gelfand, M. Hofer, and H. Pottmann. Reassembling fractured objects by geometric matching. *ACM Trans. Graph.*, 25(3):569–578, 2006. 2

[18] S. Huang, Z. Gojcic, M. Usvyatsov, A. Wieser, and K. Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 4267–4276, 2021. 3, 7

[19] A. Johnson and M. Hebert. Using spin images for efficient object recognition in cluttered 3d scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):433–449, 1999. 3

[20] B. Jones, D. Hildreth, D. Chen, I. Baran, V. G. Kim, and A. Schulz. Automate: A dataset and learning approach for automatic mating of cad assemblies. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021. 2, 9

[21] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[22] D. Koller and M. Levoy. Computer-aided reconstruction and new matches in the forma urbis romae. *Computer-aided Reconstruction and new Matches in The Forma Urbis Romae*, pages 103–125, 2006. 2

[23] N. Lamb, S. Banerjee, and N. K. Banerjee. Deepjoin: Learning a joint occupancy, signed distance, and normal field function for shape repair. *ACM Transactions on Graphics (TOG)*, 41(6):1–10, 2022. 2

[24] N. Lamb, S. Banerjee, and N. K. Banerjee. Deepmend: Learning occupancy functions to represent shape for repair. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 433–450. Springer, 2022. 2

[25] N. Lamb, S. Banerjee, and N. K. Banerjee. Mendnet: Restoration of fractured shapes using learned occupancy functions. *Comput. Graph. Forum*, 41(5):65–78, 2022. 2

[26] M. Levoy and S. Rusinkiewicz. Efficient variants of the icp algorithm. In *3D Digital Imaging and Modeling, International Conference on*, page 145, 2001. 3

[27] S.-h. Liao, C. Xiong, S. Liu, Y.-q. Zhang, and C.-l. Peng. 3d object reassembly using region-pair-relation and balanced cluster tree. *Computer Methods and Programs in Biomedicine*, 197:105756, 2020. 2

[28] J. Lu, Y. Sun, and Q. Huang. Jigsaw: Learning to assemble multiple fractured objects. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. 2, 7, 8

[29] N. Mellado, D. Aiger, and N. J. Mitra. Super 4pcs fast global pointcloud registration via smart indexing. *Computer Graphics Forum*, 33(5):205–215, 2014. 3

[30] K. Mo, S. Zhu, A. X. Chang, L. Yi, S. Tripathi, L. J. Guibas, and H. 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

[31] A. Narayan, R. Nagar, and S. Raman. Rgl-net: A recurrent graph learning framework for progressive part assembly. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 78–87, 2022. 2, 9

[32] G. Papaioannou and E.-A. Karabassi. On the automatic assemblage of arbitrary broken solid artefacts. *Image and Vision Computing*, 21(5):401–412, 2003. 2

[33] G. Papaioannou, E.-A. Karabassi, and T. Theoharis. Virtual archaeologist: Assembling the past. *IEEE Computer Graphics and Applications*, 21(2):53–59, 2001. 2

[34] G. Papaioannou, T. Schreck, A. Andreadis, P. Mavridis, R. Gregor, I. Sipiran, and K. Vardis. From reassembly to object completion: A complete systems pipeline. *Journal on Computing and Cultural Heritage (JOCCH)*, 10(2):1–22, 2017. 2

[35] C. Raposo and J. P. Barreto. Using 2 point+normal sets for fast registration of point clouds with small overlap. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5652–5658, 2017. 3

[36] R. B. Rusu, N. Blodow, Z. C. Marton, and M. Beetz. Aligning point cloud views using persistent feature histograms. In *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3384–3391, 2008. 3

[37] P.-E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4938–4947, 2020. 3, 4

[38] G. Scarpellini, S. Fiorini, F. Giuliari, P. Morerio, and A. D. Bue. Diffassemble: A unified graph-diffusion model for 2d and 3d reassembly. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 28098–28108. IEEE, 2024. 3

[39] S. Sellán, Y.-C. Chen, Z. Wu, A. Garg, and A. Jacobson. Breaking bad: A dataset for geometric fracture and reassembly. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 2, 6, 9

[40] S. Sellán, J. Luong, L. M. D. Silva, A. Ramakrishnan, Y. Yang, and A. Jacobson. Breaking good: Fracture modes for realtime destruction. *ACM Transactions on Graphics*, 2022. 9

[41] G. K. Tam, Z.-Q. Cheng, Y.-K. Lai, F. C. Langbein, Y. Liu, D. Marshall, R. R. Martin, X.-F. Sun, and P. L. Rosin. Registration of 3d point clouds and meshes: A survey from rigid to nonrigid. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1199–1217, 2013. 3

[42] F. Tombari, S. Salti, and L. Di Stefano. Unique signatures of histograms for local surface description. In K. Daniilidis, P. Maragos, and N. Paragios, editors, *Computer Vision – ECCV 2010*, pages 356–369, 2010. 3

[43] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3, 4

[44] Y. Wang and J. M. Solomon. Deep closest point: Learning representations for point cloud registration. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3523–3532, 2019. 2, 3, 6, 7

[45] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019. 3, 4

[46] K. D. Willis, P. K. Jayaraman, H. Chu, Y. Tian, Y. Li, D. Grandi, A. Sanghi, L. Tran, J. G. Lambourne, A. Solar-Lezama, et al. Joinable: Learning bottom-up assembly of parametric cad joints. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15849–15860, 2022. 2

[47] R. Wu, C. Tie, Y. Du, Y. Zhao, and H. Dong. Leveraging SE(3) equivariance for learning 3d geometric shape assembly. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 14265–14274. IEEE, 2023. 2, 7, 8

[48] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015. 2

[49] Y.-P. Xiao, Y.-K. Lai, F.-L. Zhang, C. Li, and L. Gao. A survey on deep geometry learning: From a representation perspective. *Computational Visual Media*, 6(2):113–133, 2020. 2

[50] Z. Yan, Z. Yi, R. Hu, N. J. Mitra, D. Cohen-Or, and H. Huang. Consistent two-flow network for tele-registration of point clouds. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):4304–4318, 2021. 3, 7, 8

[51] G. Zhan, Q. Fan, K. Mo, L. Shao, B. Chen, L. J. Guibas, and H. Dong. Generative 3d part assembly via dynamic graph learning. *Advances in Neural Information Processing Systems*, 33:6315–6326, 2020. 2, 4, 7, 9