Leveraging SE(3) Equivariance for Learning 3D Geometric Shape Assembly

Ruihai Wu[1,3*] **Chenrui Tie**[2,1*] **Yushi Du**[2,1] **Yan Zhao**[1,3] **Hao Dong**[1,3†]
[1]CFCS, School of CS, PKU [2]School of EECS, PKU
[3]National Key Laboratory for Multimedia Information Processing, School of CS, PKU
{wuruihai,crtie,duyushi628,yan790,hao.dong}@pku.edu.cn

Abstract

*Shape assembly aims to reassemble parts (or fragments) into a complete object, which is a common task in our daily life. Different from the semantic part assembly (*e.g.*, assembling a chair's semantic parts like legs into a whole chair), geometric part assembly (*e.g.*, assembling bowl fragments into a complete bowl) is an emerging task in computer vision and robotics. Instead of semantic information, this task focuses on geometric information of parts. As the both geometric and pose space of fractured parts are exceptionally large, shape pose disentanglement of part representations is beneficial to geometric shape assembly. In our paper, we propose to leverage SE(3) equivariance for such shape pose disentanglement. Moreover, while previous works in vision and robotics only consider SE(3) equivariance for the representations of single objects, we move a step forward and propose leveraging SE(3) equivariance for representations considering multi-part correlations, which further boosts the performance of the multi-part assembly. Experiments demonstrate the significance of SE(3) equivariance and our proposed method for geometric shape assembly. Project page: https://crtie.github.io/SE-3-part-assembly/*

1. Introduction

Shape assembly aims to compose the parts or fragments of an object into a complete shape. It is a common task in the human-built world, from furniture assembly [16, 36] (*e.g.*, assemble chair parts like legs and handles into a whole chair) to fractured object reassembly [5, 24] (*e.g.*, assemble bowl fractures into a whole bowl) . When trying to complete an object from parts, we will focus on their ***geometric*** and ***semantic*** information.

There is a vast literature in both the computer vision and robotics fields studying the shape assembly problem, especially for the application purposes like furniture assembly and object assembly [1, 16, 19, 36]. Imagine we want to assemble a simple table with four wooden sticks and a flat board, we can infer that the sticks are the table legs so they should be vertically placed, while the board is the table top and should be horizontally placed. Here, we not only use geometric clues to infer the parts' functions but also use semantic information to predict the parts' poses.

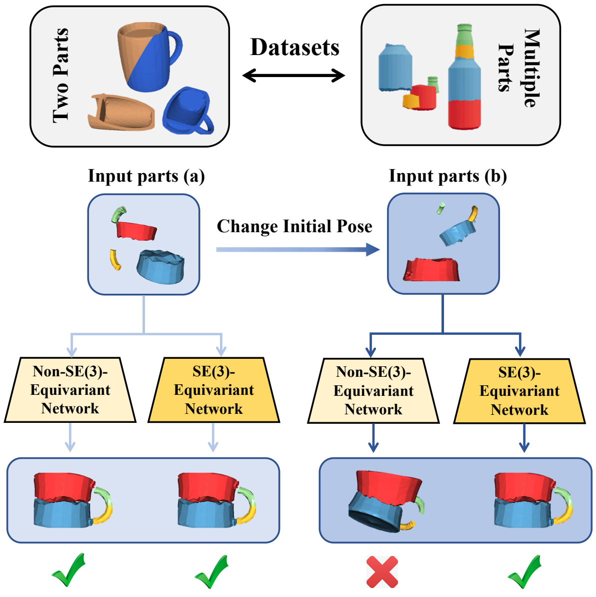
Two Parts
Datasets
Multiple Parts
Input parts (a)
Input parts (b)
Change Initial Pose
Non-SE(3)-Equivariant Network
SE(3)-Equivariant Network
Non-SE(3)-Equivariant Network
SE(3)-Equivariant Network
✓
✓
✗
✓

Figure 1. **Geometric Shape Assembly** aims to assemble different fractured parts into a whole shape. We propose to leverage **SE(3) Equivariance** for learning Geometric Shape Assembly, which disentangles poses and shapes of fractured parts, and performs better than networks without SE(3)-equivariant representations.

Recently, a two-part geometric mating dataset is proposed in NSM [5], which considers shape assembly from a pure geometric perspective, without relying on semantic information. This work randomly cuts an object into two pairs, and studies how to mate the fragment pairs into the original shape. Such design is practical in some applications such as object kitting [7, 18], form fitting [35], and pro-

[*] Equal contribution, order determined by coin flip.
[†] Corresponding author

tein binding [27]. In these tasks, the semantic information can hardly be acquired from the fragment shapes, and thus it is nearly impossible to predict fragments' poses relying on semantic information (e.g. part acts as a leg should be horizontally placed). Instead, such geometric mating tasks should be accomplished by relying on geometric cues.

Furthermore, the pairwise assembly task can be extended to the multi-part assembly task, and thus the pose space will grow much larger. Recent work [24] proposes a large-scale dataset named Breaking Bad, which models the destruction process of how an object breaks into fragments. For each object, there are multiple broken fragments with various and complex geometry, making it much more challenging for geometric shape understanding and assembly. Therefore, how to reduce the pose space and effectively assembly multiple fragments that are non-semantic but with diverse geometry still remains a problem.

Compared to furniture assembly, which relies on both part semantics and geometry, geometric assembly that assembles diverse fractures mainly focuses on geometric information, while the space of part pose and geometry are much larger in this task. Therefore, **shape pose disentanglement** plays a significant role in boosting the performance of geometric shape assembly.

Recently, achieving SE(3) equivariance for object representations is arousing much attention in 3D computer vision and robotics. Many works have studied SE(3)-equivariant architectures [3, 4, 6, 8, 13, 28, 29, 30, 38] and leveraged SE(3) equivariance in object pose estimation [17, 21] or robotic object manipulation [14, 23, 25, 26, 32]. SE(3) equivariance is suitable for the disentangling of shapes and poses of parts in geometric shape assembly. Specifically, like previous works [5, 24], we formulate the shape assembly task as a pose prediction problem, and the target is to predict the canonical SE(3) pose for each given fragment to compose a whole shape. For every single fragment, the predicted pose transformation should be *equivariant* to its original pose, while being *invariant* to other fragments' poses. Accordingly, the learned representations have two main features: *consistency* and *stability*. *Consistency* means that parts with the same geometry but different poses should have *equivariant* representations, while *stability* means the representation of a specific part should be *invariant* to all other parts' poses and only related to their geometry characteristics. Leveraging such properties, the network can reduce the large pose space of the complex geometric shape assembly task and thus focus on the fragments' geometric information for shape assembly.

While most previous works in vision and robotics only leverage SE(3) equivariance representations on a single shape, there exist multiple complex fractured parts in our geometric shape assembly task, and extracting other parts' geometric information is essential to a successful reassembly. How to leverage SE(3)-equivariant representations for multi-parts shape assembly is not a trivial problem, as learned part representations should not only consider the certain part, but also consider correlations with other parts (*e.g.*, whether the notches of two parts match each other), while keeping the equivariance property. We propose to utilize both equivariant and invariant representations of single parts to compose the equivariant part representations including part correlations. To the best of our knowledge, we are the first to leverage the SE(3) equivariance property among multiple objects.

In summary, we make the following contributions:

- We propose to leverage SE(3) equivariance that disentangles shapes and poses of fractured parts for geometric shape assembly.
- Utilizing both SE(3)-equivariant and -invariant representations, we learn SE(3)-equivariant part representations with part correlations for multi-part assembly.
- Experiments on representative benchmarks, including both two-part and multi-part 3D geometric shape assembly, demonstrate the superiority of SE(3) equivariance and our proposed method.

2. Related Works

2.1. 3D Shape Assembly

Shape assembly is a long-standing problem with a rich literature. Many works have been investigating how to construct a complete shape from given parts [5, 9, 12, 16, 19, 22, 31, 33, 36], especially in application-specific domains. Based on PartNet, a large-scale dataset that contains diverse 3D objects with fine-grained part information, previous works propose a dynamic graph learning method [36] to predict 6-DoF poses for each input part (*e.g.*, the back, legs and bars of a chair) and then assemble them into a single shape as output, or study how to assemble 3D shape given a single image depicting the complete shape [19]. Besides, many works study the shape assembly problem for different applications like furniture assembly [16], or unique needs of CAD workflow [12].

However, most previous works rely deeply on the semantic information of object parts, sometimes bypassing the geometric cues. As for the geometric cues, a recent work, NSM [5], tries to solve the two-part mating problem by mainly focusing on shape geometries without particular semantic information. Besides, a new dataset, Breaking Bad [24], raises a new challenge about how to assemble multiple non-semantic fragments into a complete shape. This work demonstrates that fractured shape reassembly is still a quite open problem. Following these two works, we focus on studying the geometric information and tackling the pure geometric shape assembly problem.

2.2. SE(3)-Equivariant Representations

Recently, achieving SE(3) equivariance has attracted a lot of attention, and many SE(3)-equivariant architectures have emerged [3, 4, 8, 13, 28, 29, 30, 38]. Thomas *et al.* [28] propose a tensor field neural network that uses filters built from spherical, and Deng *et al.* [6] introduce Vector Neurons that can facilitate rotation equivariant neural networks by lifting standard neural network representations to 3D space. We follow Vector Neuron [6] and apply the vector neuron version of DGCNN [29] model in our pipeline.

Meanwhile, many recent works have utilized equivariant models for point cloud registration [20], object detection [34], pose estimation [17, 21], robotic manipulation [14, 23, 25, 26, 32], and demonstrate that such equivariant models can significantly improve the sample efficiency and generalization ability. In this paper, we leverage SE(3)-equivariant representations for geometric shape assembly to disentangle the shape and the pose.

3. Problem Formulation

Imagine an object has been broken into N fractured parts (*e.g.*, a broken porcelain vase found during archaeological work), we obtain the point cloud of each part, which forms $\mathcal{P} = \{P_i\}_{i=1}^N$. Our goal is to assemble these parts together and recover the complete object.

Formally, our framework takes all parts' point cloud $\mathcal{P}$ as the input and predicts the canonical 3D pose of each part. We denote the predicted SE(3) pose of the i-th fractured part as $(R_i,\ T_i)$, where $R_i \in \mathbb{R}^{3\times3}$ is the predicted rotation matrix and $T_i \in \mathbb{R}^3$ is the predicted translation vector. Then, we apply the predicted pose to transform the point cloud of each part and get the i-th part's predicted point cloud $P'_i = P_iR_i + T_i$. The union of all the transformed point clouds $P'_{whole} = \bigcup_i P'_i$ is our predicted assembly result.

4. Method

Our method leverages SE(3)-equivariant representations for geometric shape assembly. We start the Method Section by describing how to leverage SE(3)-equivariant for a single part as a basis (Sec. 4.1). Then, as geometric shape assembly requires each part to consider its correlations with other parts, we describe extending the leverage of SE(3) equivariance from single-part representations to part representations considering correlations with other parts (Sec. 4.2). Based on the learned equivariant representations and apart from predicting the pose of each fractured part, to further ensure all the re-posed parts compose a whole object, we propose translation embedding (Sec. 4.3) for geometric assembly and use adversarial learning (Sec. 4.4). Finally, we describe the loss functions (Sec. 4.5).

4.1. Leveraging SE(3) Equivariance for Single Parts

For the brevity of description, we first introduce a simple version (as a basis of our whole method): leveraging SE(3) equivariance in single parts' representations, without considering the correlations between multiple parts.

Specifically, in this section, we start by revisiting Vector Neurons Networks (VNN) [6], a general framework for SO(3)-equivariant (rotation equivariant) network. Leveraging VNN, we introduce how we leverage rotation equivariance and translation equivariance for single parts.

Vector Neurons Networks (VNN) is a general framework for SO(3)-equivariant networks. It extends neurons from 1D scalars to 3D vectors and provides various SO(3)-equivariant neural operations including linear layers (such as Conv and MLP), non-linear layers (such as Pooling and ReLU) and normalization layers. Besides, it also designs SO(3)-invariant layers to extract SO(3)-invariant representations. The above properties are mathematically rigorous.

Rotation Equivariance and Invariance. In geometric part assembly, we suppose the predicted rotation of a part is equivariant with its original orientation and invariant with other parts' orientation. Accordingly, the network should learn both equivariant and invariant representations for each part. Based on VNN, we build a DGCNN [29] encoder with a SO(3)-equivariant head $\mathcal{E}_{equiv}$ and a SO(3)-invariant encoder head $\mathcal{E}_{inv}$ to extract part features with corresponding properties. Specifically, given an input point cloud P, and a random rotation matrix R, the encoders $\mathcal{E}_{equiv}$ and $\mathcal{E}_{inv}$ respectively satisfy rotation equivariance and invariance:

$$\mathcal{E}_{equiv}(PR) = \mathcal{E}_{equiv}(P)R \tag{1}$$

$$\mathcal{E}_{inv}(PR) = \mathcal{E}_{inv}(P) \tag{2}$$

Translation Equivariance. To achieve translation equivariance in parts' pose prediction, we preprocess the raw point cloud of each part by posing its gravity center on the coordinate origin. That's to say, with an input point cloud $P = (p_1, p_2, ..., p_n), p_i \in \mathbb{R}^3$, where n is the number of points, we compute its gravity center $\hat{x} = (\sum_{i=1}^n p_i)/n$, and get the preprocessed point cloud $\tilde{P} = P - \hat{x}$, and then we use $\tilde{P}$ as the network input. In this way, our prediction is translation equivariant. Formally, let T_{pred} denote the predicted translation output, if the part's point cloud changes from P to $P + \Delta T$, we have:

$$T_{pred}(P + \Delta T) = T_{pred}(P) + \Delta T \tag{3}$$

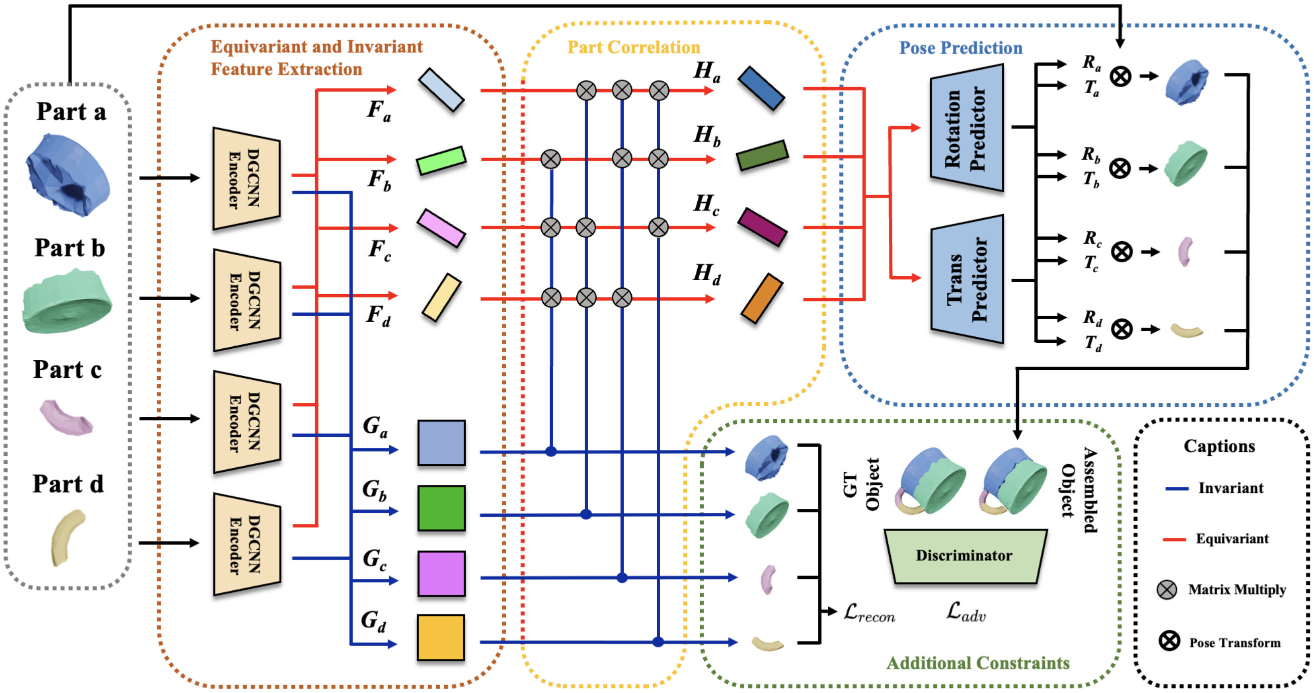

Figure 2. **Overview of our proposed framework.** Taking as input the point cloud of each part i, our framework first outputs the equivariant representation F_i and invariant representation G_i, computes the correlation between part i and each part j using the matrix multiplication of F_i and G_j, and thus gets each part's equivariant representation H_i with part correlations. The rotation decoder and the translation decoder respectively take H and decode the rotation and translation of each part. Additional constraints such as adversarial training and canonical point cloud reconstruction using G further improves the performance of our method.

4.2. Leveraging SE(3) Equivariance for Parts with Part Correlations

In the geometric shape assembly task, all the fractured parts should be reassembled together, with each edge matching up well with the others. Therefore, all parts' geometry information, especially the geometry of edges, is significant. Besides, it is necessary to analyze all parts' shape information together to infer the position of each individual part, otherwise, it would be nearly impossible to predict accurate positions for those parts. Therefore, the correlations between parts are essential in the geometric assembly task, and we propose a correlation module to aggregate the information of multiple parts, while keeping leveraging SE(3) equivariance.

Note that the translation equivariance in part pose predictions can be achieved using the same approach in Sec. 4.1, so in this section we mainly describe how to leverage rotation equivariance when taking multiple parts.

Rotation Equivariance in Multi-Part Representations. To predict the final pose of part i, we should consider its correlations with other parts. In fact, what really matters for this part's pose prediction is other parts' shape geometry instead of their initial poses. In other words, changed initial poses of other parts may not affect the predicted pose of part i. Therefore, it comes naturally that we leverage the rotation-invariant representations of other parts (which are invariant to their initial poses) to extract their geometric features and further compute their correlations with part i.

Specifically, given the point cloud $P_i \in \mathbb{R}^{n\times 3}$ of the i-th part, we pass it through rotation-equivariant and -invariant encoders $\mathcal{E}_{equiv}$ and $\mathcal{E}_{inv}$ and get corresponding features (shown in the **Equivariant and Invariant Feature Extraction** module in Figure 2):

$$\begin{aligned} F_i = \mathcal{E}_{equiv}(P_i), \quad & F_i \in \mathbb{R}^{f\times 3} \\ G_i = \mathcal{E}_{inv}(P_i), \quad & G_i \in \mathbb{R}^{f\times f} \end{aligned} \tag{4}$$

As shown in the **Part Correlation** module in Figure 2, to extract the correlation feature $C_{i,j}$ between part i and part $j, (j \neq i)$, we use matrix multiplication between G_j and F_i:

$$C_{i,j} = G_j \cdot F_i, \quad C_{i,j} \in \mathbb{R}^{f\times 3} \tag{5}$$

where · denotes **matrix multiplication**.

As G_j is invariant to P_j, and F_i is equivariant to P_i, the matrix multiplication $C_{i,j}$ is thus equivariant to P_i with the geometry correlation between P_i and P_j.

Furthermore, to get the part representation H_i considering correlations with all other parts while maintaining equivariance with P_i, we define H_i as:

$$H_i = \frac{1}{N-1} \sum_{j=1,\, j \neq i}^{N} G_j \cdot F_i, \quad H_i \in \mathbb{R}^{f \times 3} \tag{6}$$

As G_j is invariant with P_i and F_i is equivariant with P_i, it's easy to verify that, H_i is equivariant with P_i, and invariant with $P_j (j \neq i)$. *i.e.*, for any rotation matrix R applied on part i or any other part j:

$$\begin{aligned} &H_i(P_1, ..., P_i R, ... P_N) = H_i(P_1, ..., P_i, ..., P_N) R \\ &H_i(P_1, ..., P_j R, ... P_N) = H_i(P_1, ..., P_j, ..., P_N), (j \neq i) \end{aligned} \tag{7}$$

Pose Prediction. As shown in the **Pose Prediction** module in Figure 2, given the equivariant representation H_i of part i with part correlations, we use a pose regressor $\mathcal{R}$ to predict its rotation $R_{pred,\, i}$ and translation $T_{pred,\, i}$:

$$\begin{gathered} R_{pred,\, i},\, T_{pred,\, i} = \mathcal{R}(H_i), \\ R_{pred,\, i} \in \mathbb{R}^{3 \times 3}, \quad T_{pred,\, i} \in \mathbb{R}^3 \end{gathered} \tag{8}$$

Canonical Part Reconstruction. To ensure that the rotation invariant feature G_i encodes geometric information of P_i with any initial pose, we use a point cloud decoder $\mathcal{D}$ and expect $\mathcal{D}$ to decode point cloud of i-th part in the canonical view when receiving G_i (shown in the **Additional Constraint** module in Figure 2):

$$P^*_{pred,\, i} = \mathcal{D}(G_i), \quad P^*_{pred,\, i} \in \mathbb{R}^{n \times 3} \tag{9}$$

Let $P^*_{gt,i}$ denote the canonical point cloud of P_i and $P^*_{pred,\, i}$ denote the prediction, we minimize the Chamfer Distance between $P^*_{pred,\, i}$ and $P^*_{gt,\, i}$.

4.3. Translation Embeddings for Part Representations

Since the reassembled whole object is the composition of multiple re-posed parts, although the above described designs learn the pose of each part, the framework lacks leveraging the property that the representations of all parts could compose the whole object.

Inspired by Visual Translation Embedding (VTransE) [11, 37] that maps different objects' features into a space where the relations between objects can be the feature translation, we propose a similar Translation Embedding where the representations of parts can be added up to the representation of the whole shape.

Formally, denoting the point cloud of the whole object at canonical pose as P^*_{gt}, we pass it through our rotation equivariant encoder to get $F^*_{gt} = \mathcal{E}_{equiv}(P^*_{gt})$, and minimize:

$$\mathcal{L}2(\sum_i H_i,\, F^*_{gt}) \tag{10}$$

where H_i is the rotation equivariant feature of i-th fracture.

Through this procedure, rotation equivariant representations of parts would be more interpretable as a whole.

4.4. Adversarial Learning

The above described designs use proposed representations to learn the pose of each part, lacking the evaluation that all re-posed parts visually make up a whole shape. Following the design of [5], we employ a discriminator $\mathcal{M}$ and use adversarial learning to make the re-posed parts visually look like those of a whole object, as shown in the **Additional Constraint** module in Figure 2.

Our discriminator $\mathcal{M}$ takes as input the predicted reassembly shape P'_{whole} (defined in Sec. 3) and the ground truth point cloud of the whole object P_{whole}, and distinguishes whether the input point clouds look visually plausible like a complete object. To achieve this, we define a loss term $\mathcal{L}_\mathrm{G}$ for training the generator ($i.e.$ encoders $\mathcal{E}$ and pose regressor $\mathcal{R}$), which is defined as:

$$\mathcal{L}_G = \mathbb{E}\left[\|\mathcal{M}(P'_{whole}) - 1\|\right], \tag{11}$$

and an adversarial loss $\mathcal{L}_D$ for training the discriminator $\mathcal{M}$, which is defined as:

$$\mathcal{L}_D = \mathbb{E}\left[\|\mathcal{M}(P'_{whole})\|\right] + \mathbb{E}\left[\|\mathcal{M}(P_{whole}) - 1\|\right] \tag{12}$$

Through the adversarial training procedure, the reassembled shapes become more plausible as a whole.

4.5. Losses

Our loss function consists of the following terms:

$$\begin{aligned} \mathcal{L} = &\lambda_{rot}\mathcal{L}_{rot} + \lambda_{trans}\mathcal{L}_{trans} + \lambda_{point}\mathcal{L}_{point} \\ &+ \lambda_{recon}\mathcal{L}_{recon} + \lambda_{embed}\mathcal{L}_{embed} + \lambda_{adv}\mathcal{L}_{adv} \end{aligned} \tag{13}$$

For an input broken object, we sample point clouds from every fractured part and form $\mathcal{P} = \{P_i\}_{i=1}^N$. For the i-th part, we denote its ground truth rotation matrix and translation as $R_{gt,\, i}$ and $T_{gt,\, i}$, and the predicted rotation matrix and translation as $R_{pred,\, i}$ and $T_{pred,\, i}$.

For rotation, we use geodesic distance (GD) between R_{gt} and R_{pred} as our rotation loss:

$$\mathcal{L}_{rot} = arccos \frac{tr(R_{gt} R^T_{pred}) - 1}{2} \tag{14}$$

For translation, we use $\mathcal{L}2$ loss between T_{gt} and T_{pred} as the translation prediction loss:

$$\mathcal{L}_{trans} = \mathcal{L}2(T_{pred},\, T_{gt}) \tag{15}$$

Following [24], we use Chamfer Distance to further jointly supervise the predicted translation and rotation by supervising the predicted re-posed point cloud:

$$\mathcal{L}_{point} = Chamfer(PR_{pred} + T_{pred},\ PR_{gt} + T_{gt}) \quad (16)$$

As mentioned in Sec. 4.2, we also use Chamfer Distance as the reconstruction loss to supervise the invariant representation G_i can be further decoded to a canonical point cloud $P^*_{pred,i}$, ensuring G_i encodes geometric information with any initial pose:

$$\mathcal{L}_{recon} = Chamfer(P^*_{pred,i},\ P_i R_{gt,i} + T_{gt,i}) \quad (17)$$

From Sec. 4.3, we design translation embedding loss to supervise that the representations of all fractured parts can be added up to the representation of the complete shape:

$$\mathcal{L}_{embed} = \mathcal{L}2(\sum_i H_i,\ F^*_{gt}) \quad (18)$$

From Sec. 4.4, the adversarial loss is defined as:

$$\mathcal{L}_{adv} = \mathbb{1}_D \mathcal{L}_D + \mathbb{1}_G \mathcal{L}_G \quad (19)$$

where $\mathbb{1}_D = 1$ only if we're updating discriminator, $\mathbb{1}_G = 1$ only if updating generator.

5. Experiments

5.1. Datasets, Settings and Metrics

Datasets. We use two benchmark datasets for evaluation:

- Geometric Shape Mating dataset [5] for **two-part assembly (mating)**. Objects in this dataset are cut into two parts by the randomly generated heightfields that can be parameterized by different functions. Specifically, we employ 4 kinds of cut types (planar, sine, parabolic and square functions) on 5 categories of objects (Bag, Bowl, Jar, Mug and Sofa) in ShapeNet [2]. We employ the official data collection code, collect 41,000 cuts for training and 3,100 cuts for testing.

- Breaking Bad dataset's commonly used "everyday" object subset [24] for **multi-part assembly**. Compared with Geometric Shape Mating dataset, this dataset is much more challenging, as the objects are irregularly broken into multiple fragments by physical plausible decomposition , resulting in more parts with much more complex geometries. Our study focuses more on this multi-part geometric assembly problem.

On both datasets, we train all methods in all categories, and test them on unseen objects in the same categories.

Metrics. Following the evaluation metrics of the two datasets [5, 24], we import geodesic distance (GD) to measure the difference between predicted rotation and ground truth rotation. To further evaluate both the rotation and translation prediction, we compute the root mean squared error RMSE (R) between the predicted rotation R and the corresponding ground truth values, and the root mean squared error RMSE (T) between the predicted translation T and the corresponding ground truth values. Here we use Euler angle to represent rotation.

Besides, we follow the evaluation protocol in [19, 24] and adopt part accuracy (PA) as an evaluation metric. This metric measures the portion of 'correctly placed' parts. We first use predicted rotation and translation to transform the input point cloud, and then compute the Chamfer Distance between the transformed point cloud and the ground truth. If the distance is smaller than a threshold, we count this part as 'correctly placed'.

Hyper-parameters. We set batch size to be 32 for Breaking Bad, 48 for Geometric Shape Mating, and the initial learning rate of Adam Optimizer [15] to be 0.0001. We train the model 80 and 120 epochs respectively for Geometric Shape Mating and Breaking Bad.

5.2. Baselines

For the Geometric Shape Mating dataset, the two-part geometric shape assembly task, we compare our method with NSM [5], the state-of-the-art method for two-part mating. For the Breaking Bad dataset, the multi-part geometric shape assembly task, we modified the official code of the NSM [5] from two-part geometric shape assembly to multi-part geometric assembly by predicting the pose of each input part. We also compare our method with DGL [36] and LSTM [10] following the Breaking Bad benchmark [24]. All baseline implementations use the official code in two benchmarks [5, 24]. The baselines are described as follows:

- **NSM** [5] extracts part features using transformer and predicts their poses for mating, achieving state-of-the-art performance in two-part mating.

- **DGL** [24, 36] uses graph neural networks to encode and aggregate part features, and predicts the pose of each part. Following [24], we remove the node aggregation procedure as there does not exist parts with the same geometric appearance in geometric assembly.

- **LSTM** [24, 36, 10] uses bi-directional LSTM to take part features as input and sequentially predicts the pose of each part. This method assembles the decision-making method of humans when faced with geometric shape assembly problems.

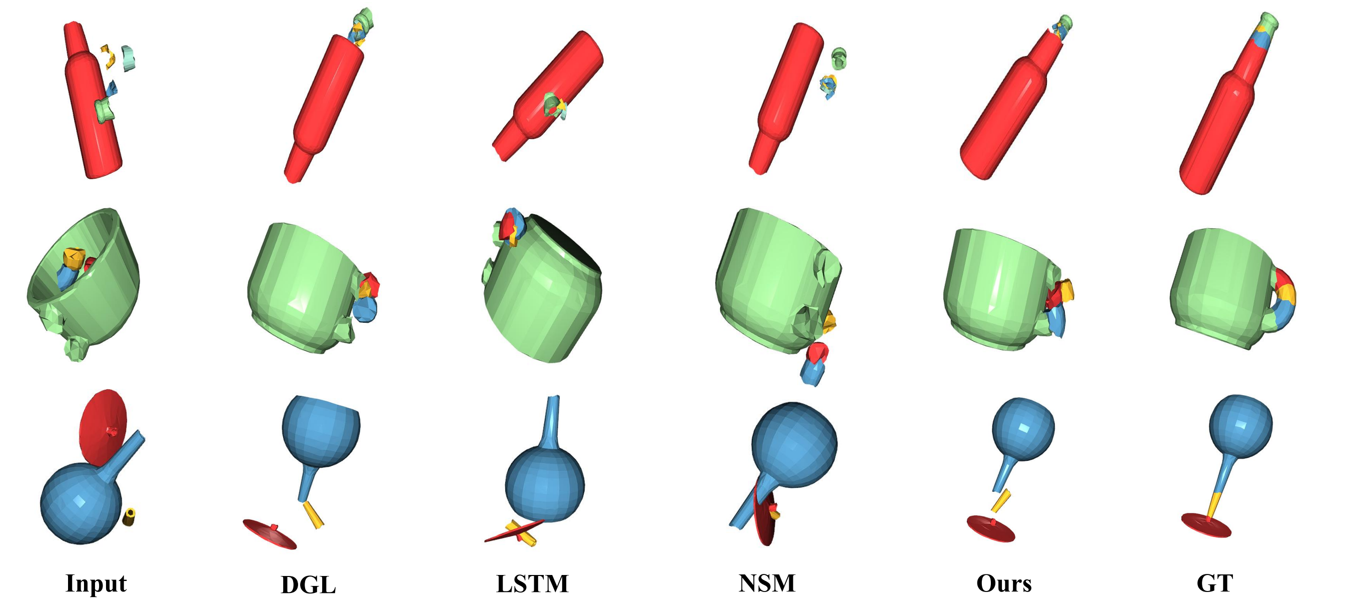

Figure 3. **Qualitative results on Breaking Bad dataset for multi-part geometry shape assembly.** We observe better rotation and translation predictions (especially rotation) than baseline methods.

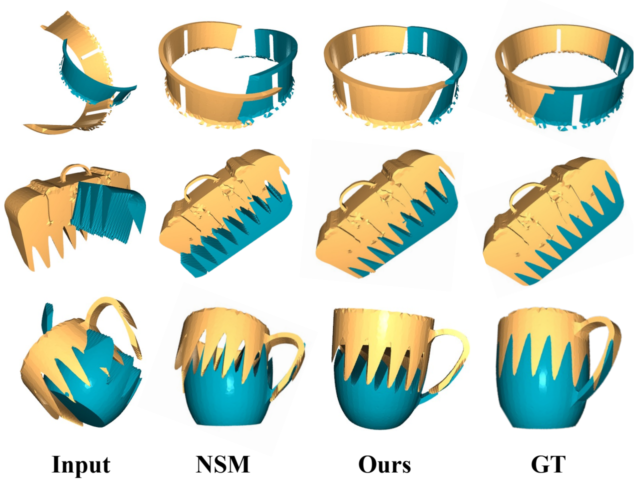

Figure 4. **Qualitative results on Geometric Shape Mating dataset for two-part geometric shape assembly.** We observe better pose predictions (especially rotation) than NSM.

Method	RMSE (R) ↓	GD (R) ↓	RMSE (T) ↓	PA ↑
	degree	rad	$\times 10^{-2}$	%
DGL	84.1	2.21	14.7	22.9
LSTM	87.6	2.24	16.3	13.4
NSM	85.6	2.21	15.7	16.0
Ours	**75.3**	**2.00**	**14.1**	**26.7**

Table 1. **Quantitative evaluation on Breaking Bad dataset for multi-part geometric assembly.** We report quantitative results of our method and three learning-based shape assembly baselines on the `everyday` object subset.

Method	RMSE (R) ↓	GD (R) ↓	RMSE (T) ↓	PA ↑
	degree	rad	$\times 10^{-2}$	%
NSM	21.3	0.52	2.9	79.1
Ours	**15.9**	**0.39**	**2.7**	**85.7**

Table 2. **Quantitative evaluation on Geometric Shape Mating dataset for two-part geometric assembly.** We report quantitative results of our method and the NSM baseline.

5.3. Experimental Results and Analysis

Table 1 and 2 show the quantitative performance of our method and baselines. The experimental results demonstrate that our method performs better than all baselines in both two-part and multi-part geometric shape assembly tasks over all evaluation metrics.

As discussed in [24], predicting rotations of multiple parts is pretty more difficult than translation. Table 1 and 2 show our method has a significant improvement in this aspect, and outperforms all baselines in the root mean squared error RMSE (R) metric and the geodesic distance (GD) metric. In particular, our rotation is around 10 degrees less than the baselines. For translation prediction, our RMSE (T) also outperforms all baselines on both datasets. In addition, our method also outperforms all baselines in part accuracy (PA), especially for the LSTM and NSM in the more challenging Breaking Bad dataset.

This may result from our SO(3) equivariant network that disentangles shape and pose information, reducing the difficulty of learning rotations of different posed parts with different geometries, thus allowing for better predictions.

Figure 4 shows qualitative examples of our method and NSM on the Geometric Shape Mating dataset. Although it is a comparatively simple dataset and the task is nearly solved by previous methods, our method still performs better, especially in rotation prediction.

Figure 3 shows qualitative comparisons between our method and baselines on the more challenging and realistic Breaking Bad dataset. Although this task is highly difficult and all methods could not solve the task, our method can better predict the pose (especially the rotation) of each part.

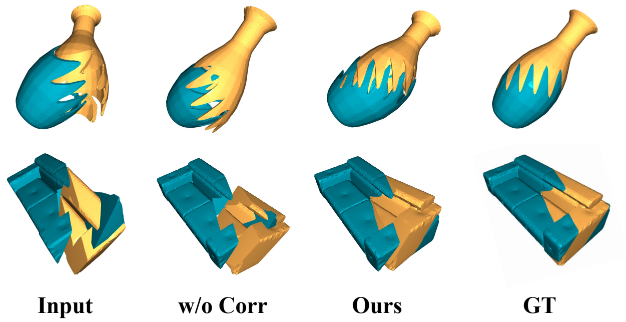

Figure 5. **Qualitative results of our method with and without part correlation on Geometric Shape Mating dataset.** The parts with representations considering part correlations match better.

5.4. Ablation Studies

To further evaluate the effectiveness of different components in our framework, we conduct ablation studies by comparing our method with the following ablated versions:

- **w/o Corr**: our method without considering part correlations in each part's equivariant representations.
- **w/o TE**: our method without translation embedding.
- **w/o Adv**: our method without adversarial learning.

As shown in Table 3 and 4, and Figure 5, the performance decline when removing part correlations in part representations demonstrate that, our proposed part correlations help in the geometric assembly of fractured parts, as it is significant to aggregate geometric information between parts for geometric shape assembly.

As shown in Table 3 and 4, the translation embedding and adversarial training help improve the performance of our method, as described in Section 4.3 and 4.4, translation embedding and adversarial learning and can serve as pose fine-tuners and improve pose predictions.

Method	RMSE (R) ↓	GD (R) ↓	RMSE (T) ↓	PA ↑
	degree	rad	$\times 10^{-2}$	%
w/o Corr	19.2	0.52	2.9	80.5
w/o TE	17.3	0.46	2.8	84.3
w/o Adv	16.7	0.43	2.8	82.6
Ours	**15.9**	**0.39**	**2.7**	**85.7**

Table 3. **Ablations on Geometric Shape Mating.** We compare with versions removing part correlations (w/o Corr), translation embedding (w/o TE) and adversarial learning (w/o Adv).

Method	RMSE (R) ↓	GD (R) ↓	RMSE (T) ↓	PA ↑
	degree	rad	$\times 10^{-2}$	%
w/o Corr	79.8	2.17	15.7	18.4
w/o TE	77.2	2.04	15.2	22.5
w/o Adv	77.6	2.02	14.3	23.7
Ours	**75.3**	**2.00**	**14.1**	**26.7**

Table 4. **Ablations on Breaking Bad.** We compare with versions removing part correlations (w/o Corr), translation embedding (w/o TE) and adversarial learning (w/o Adv).

6. Conclusion

In this paper, to tackle 3D geometric shape assembly tasks that rely on *geometric* information of fractured parts, we propose to leverage SE(3)-equivariant representations that disentangle shapes and poses to facilitate the task. Our method leverages SE(3) equivariance in part representations considering part correlations, by learning both SE(3)-equivariant and -invariant part representations and aggregating them into SE(3)-equivariant representations. To the best of our knowledge, we are the first to explore leveraging SE(3) equivariance on multiple objects in related fields. Experiments demonstrate the effectiveness of our method.

Limitations & Future Work In Breaking Bad, although we perform better than all baselines, this does not mean that we have solved the problem. When the number of fractures increases, the problem's complexity increases sharply, and most existing methods cannot perform well. To completely solve the problem, more additional designs need to be added, while leveraging SE(3) equivariance is orthogonal to many designs. For the whole framework, while the learned representations are equivariant to input part poses, the rotation regressor is non-equivariant, as it limits the degree-of-freedom in pose prediction and leads to worse results. Besides, it will take computing resources and time to train equivariant networks than ordinary networks.

7. Acknowledge

This work was supported by National Natural Science Foundation of China (No. 62136001).

References

[1] Maneesh Agrawala, Doantam Phan, Julie Heiser, John Haymaker, Jeff Klingner, Pat Hanrahan, and Barbara Tversky. Designing effective step-by-step assembly instructions. *ACM Transactions on Graphics (TOG)*, 22(3):828–837, 2003. 1

[2] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 6

[3] Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill. Equivariant point network for 3d point cloud analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14514–14523, 2021. 2, 3

[4] Yunlu Chen, Basura Fernando, Hakan Bilen, Matthias Nießner, and Efstratios Gavves. 3d equivariant graph implicit functions. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 485–502. Springer, 2022. 2, 3

[5] Yun-Chun Chen, Haoda Li, Dylan Turpin, Alec Jacobson, and Animesh Garg. Neural shape mating: Self-supervised object assembly with adversarial shape priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12724–12733, 2022. 1, 2, 5, 6

[6] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J Guibas. Vector neurons: A general framework for so (3)-equivariant networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12200–12209, 2021. 2, 3

[7] Shivin Devgon, Jeffrey Ichnowski, Michael Danielczuk, Daniel S Brown, Ashwin Balakrishna, Shirin Joshi, Eduardo MC Rocha, Eugen Solowjow, and Ken Goldberg. Kitnet: Self-supervised learning to kit novel 3d objects into novel 3d cavities. In *2021 IEEE 17th International Conference on Automation Science and Engineering (CASE)*, pages 1124–1131. IEEE, 2021. 1

[8] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in Neural Information Processing Systems*, 33:1970–1981, 2020. 2, 3

[9] Thomas Funkhouser, Hijung Shin, Corey Toler-Franklin, Antonio García Castañeda, Benedict Brown, David Dobkin, Szymon Rusinkiewicz, and Tim Weyrich. Learning how to match fresco fragments. *Journal on Computing and Cultural Heritage (JOCCH)*, 4(2):1–13, 2011. 2

[10] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 6

[11] Zih-Siou Hung, Arun Mallya, and Svetlana Lazebnik. Contextual translation embedding for visual relationship detection and scene graph generation. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):3820–3832, 2020. 5

[12] Benjamin Jones, Dalton Hildreth, Duowen Chen, Ilya Baran, Vladimir G Kim, and Adriana Schulz. Automate: A dataset and learning approach for automatic mating of cad assemblies. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021. 2

[13] Oren Katzir, Dani Lischinski, and Daniel Cohen-Or. Shape-pose disentanglement using se (3)-equivariant vector neurons. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part III*, pages 468–484. Springer, 2022. 2, 3

[14] Seungyeon Kim, Byeongdo Lim, Yonghyeon Lee, and Frank C Park. Se (2)-equivariant pushing dynamics models for tabletop object manipulations. In *6th Annual Conference on Robot Learning*. 2, 3

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *The 3rd International Conference for Learning Representations*, 2015. 6

[16] Youngwoon Lee, Edward S Hu, and Joseph J Lim. Ikea furniture assembly environment for long-horizon complex manipulation tasks. In *2021 ieee international conference on robotics and automation (icra)*, pages 6343–6349. IEEE, 2021. 1, 2

[17] Xiaolong Li, Yijia Weng, Li Yi, Leonidas J Guibas, A Abbott, Shuran Song, and He Wang. Leveraging se (3) equivariance for self-supervised category-level object pose estimation from point clouds. *Advances in Neural Information Processing Systems*, 34, 2021. 2, 3

[18] Yulong Li, Shubham Agrawal, Jen-Shuo Liu, Steven K Feiner, and Shuran Song. Scene editing as teleoperation: A case study in 6dof kit assembly. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4773–4780. IEEE, 2022. 1

[19] Yichen Li, Kaichun Mo, Lin Shao, Minhyuk Sung, and Leonidas Guibas. Learning 3d part assembly from a single image. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 664–682. Springer, 2020. 1, 2, 6

[20] Cheng-Wei Lin, Tung-I Chen, Hsin-Ying Lee, Wen-Chin Chen, and Winston H Hsu. Coarse-to-fine point cloud registration with se (3)-equivariant representations. *arXiv preprint arXiv:2210.02045*, 2022. 3

[21] Xueyi Liu, Ji Zhang, Ruizhen Hu, Haibin Huang, He Wang, and Li Yi. Self-supervised category-level articulated object pose estimation with part-level SE(3) equivariance. In *The Eleventh International Conference on Learning Representations*, 2023. 2, 3

[22] Abhinav Narayan, Rajendra Nagar, and Shanmuganathan 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

[23] Hyunwoo Ryu, Jeong-Hoon Lee, Hong-in Lee, and Jongeun Choi. Equivariant descriptor fields: Se (3)-equivariant energy-based models for end-to-end visual robotic manipulation learning. *arXiv preprint arXiv:2206.08321*, 2022. 2, 3

[24] Silvia Sellán, Yun-Chun Chen, Ziyi Wu, Animesh Garg, and Alec Jacobson. Breaking bad: A dataset for geometric fracture and reassembly. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 1, 2, 6, 7

[25] Anthony Simeonov, Yilun Du, Yen-Chen Lin, Alberto Rodriguez Garcia, Leslie Pack Kaelbling, Tomás Lozano-Pérez, and Pulkit Agrawal. SE(3)-equivariant relational rearrangement with neural descriptor fields. In *6th Annual Conference on Robot Learning*, 2022. 2, 3

[26] Anthony Simeonov, Yilun Du, Andrea Tagliasacchi, Joshua B Tenenbaum, Alberto Rodriguez, Pulkit Agrawal, and Vincent Sitzmann. Neural descriptor fields: Se (3)-equivariant object representations for manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 6394–6400. IEEE, 2022. 2, 3

[27] Freyr Sverrisson, Jean Feydy, Bruno E Correia, and Michael M Bronstein. Fast end-to-end learning on protein surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15272–15281, 2021. 2

[28] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018. 2, 3

[29] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019. 2, 3

[30] Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco S Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems*, 31, 2018. 2, 3

[31] Karl DD Willis, Pradeep Kumar Jayaraman, Hang Chu, Yunsheng Tian, Yifei Li, Daniele Grandi, Aditya Sanghi, Linh Tran, Joseph G Lambourne, Armando 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

[32] Zhengrong Xue, Zhecheng Yuan, Jiashun Wang, Xueqian Wang, Yang Gao, and Huazhe Xu. Useek: Unsupervised se (3)-equivariant 3d keypoints for generalizable manipulation. *arXiv preprint arXiv:2209.13864*, 2022. 2, 3

[33] Kangxue Yin, Zhiqin Chen, Siddhartha Chaudhuri, Matthew Fisher, Vladimir G Kim, and Hao Zhang. Coalesce: Component assembly by learning to synthesize connections. In *2020 International Conference on 3D Vision (3DV)*, pages 61–70. IEEE, 2020. 2

[34] Hong-Xing Yu, Jiajun Wu, and Li Yi. Rotationally equivariant 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1456–1464, 2022. 3

[35] Kevin Zakka, Andy Zeng, Johnny Lee, and Shuran Song. Form2fit: Learning shape priors for generalizable assembly from disassembly. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9404–9410. IEEE, 2020. 1

[36] Guanqi Zhan, Qingnan Fan, Kaichun Mo, Lin Shao, Baoquan Chen, Leonidas J Guibas, Hao Dong, et al. Generative 3d part assembly via dynamic graph learning. *Advances in Neural Information Processing Systems*, 33:6315–6326, 2020. 1, 2, 6

[37] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5532–5540, 2017. 5

[38] Yongheng Zhao, Tolga Birdal, Jan Eric Lenssen, Emanuele Menegatti, Leonidas Guibas, and Federico Tombari. Quaternion equivariant capsule networks for 3d point clouds. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 1–19. Springer, 2020. 2, 3