Compositionally Generalizable 3D Structure Prediction

Songfang Han
UC San Diego

Jiayuan Gu
UC San Diego

Kaichun Mo
Stanford University

Li Yi
Google Research

Siyu Hu
USTC

Xuejin Chen
USTC

Hao Su
UC San Diego

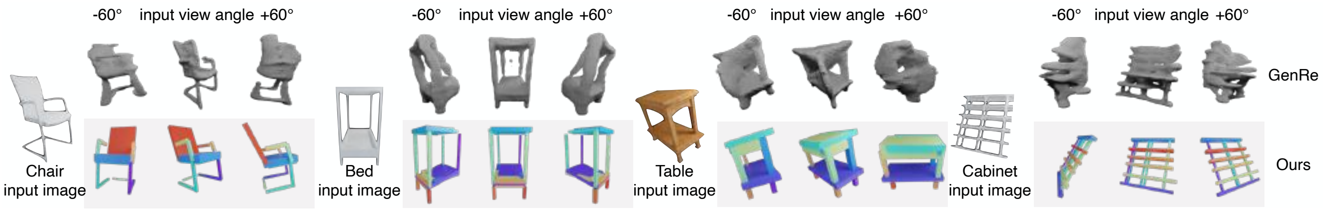
-60° input view angle +60°
-60° input view angle +60°
-60° input view angle +60°
-60° input view angle +60°
GenRe
Ours
Chair
input image
Bed
input image
Table
input image
Cabinet
input image

Figure 1: 3D structure predictions using models trained on *chair* shapes, by GenRe [62] and our approach (visualized at three views). Our method can generalize to novel unseen categories: bed, table, and cabinet. We achieve superior reconstruction results than GenRe.

Abstract

*Single-image 3D shape reconstruction is an important and long-standing problem in computer vision. A plethora of existing works is constantly pushing the state-of-the-art performance in the deep learning era. However, there remains a much more difficult and under-explored issue on how to generalize the learned skills over unseen object categories that have very different shape geometry distributions. In this paper, we bring in the concept of compositional generalizability and propose a novel framework that could better generalize to these unseen categories. We factorize the 3D shape reconstruction problem into proper sub-problems, each of which is tackled by a carefully designed neural sub-module with generalizability concerns. The intuition behind our formulation is that object parts (*e.g. *slates and cylindrical parts), their relationships (*e.g. *adjacency and translation symmetry) and shape substructures (*e.g. *T-junctions and a symmetric group of parts) are mostly shared across object categories, even though object geometries may look very different (*e.g. *chairs and cabinets). Experiments on PartNet show that we achieve superior performance than state of the art. This validates our problem factorization and network designs.*

1. Introduction

Single-image 3D reconstruction is a critical and long-standing problem in computer vision. Existing works in the literature of 3D deep learning [4, 7, 52, 3, 29, 36] mostly focus on designing better network architectures for decoding 3D geometry from 2D image inputs, assuming that the training and test data come from the same or similar categories.

We are interested in a much harder setting, where the training and testing data may be sampled from very different object categories. An example would be to reconstruct beds after seeing training data of chairs. While such cross-category generalizability is under-explored in the 3D deep learning literature, it is a fundamental capability of humans — as we explore 2D product images online, we constantly imagine 3D geometry of objects from unknown categories yet seldom running into difficulties.

Very few works investigated this cross-category generalization setup [62, 51]. Inspired by David Marr's seminal idea on 2.5D representations, GenRe [62] and its follow-up work [51] hypothesize that depth map in image space leads to a better cross-category generalizability than predicting 3D shapes directly. While it does generalize better than previous works, its basis, generalizable depth prediction across categories, is indeed an unsolved and difficult problem. Besides, merely using depth representation fails to leverage the underlying structure of objects, which is critical and natural to human understanding of 3D shapes. For example, humans can reconstruct a chair as a composite of several similar chair legs. Fig. 1 presents exemplar comparisons between GenRe [62] and our method on the cross-category 3D reconstruction, where we clearly observe that GenRe suffers from imperfect depth estimation, while our method can recover the object structures more faithfully.

In this work, we explore a very distinctive path toward cross-category generalizability based on the following observation: while training and test objects from different categories might follow very different whole-object level shape distribution, they often share similar distribution in terms of "substructures". For example, among all man-made objects, slates and cylindrical parts are quite common. And among all possible configurations of parts, parallel and T-junction relationships are more frequently observed. Taking the Bayesian's view, the joint probability $\Pr(p_1, p_2, \ldots, p_n)$ of all substructures p_i could be very different across training and test sets. However, if we factorize the joint probability as products of low-order potentials $\phi(p_i)$ and $\phi(p_i, p_j)$, these potentials are often much better aligned. This motivates us to factorize the problem of shape reconstruction into sub-problems of substructure prediction, which has the potential of being more generalizable across categories. We refer this factorization-enabled generalizability as *compositional generalizability* [34, 27].

To this end, we interpret a 3D shape as a composite of parts. Specifically, we use **oriented bounding boxes** to represent parts and parameterize a 3D shape by *connectivity*, *orientation*, and *size* of part bounding boxes. Instead of focusing on fine geometry, this representation encodes abstract structure, which facilitates many applications that require structure-level reasoning [48, 31, 32].

To predict such structured shapes, we build a system composed of independent modular networks. We first extract part segmentation from an input image. The part mask provides an explicit attention map. The network is encouraged to learn local structures, by keeping only the part masks of interest in the input. Given part masks, we further factorize shape reconstruction into a series of sub-problems, that are carefully designed and demonstrated with strong transferability. In particular, we predict the part orientation for each part mask and then regress the size in a group-wise fashion. After obtaining a set of part bounding boxes, we finally learn to assemble these part bounding boxes into a 3D shape. To tackle non-trivial challenges for independent modules, we also devise several novel techniques.

For the part orientation prediction module, we observe that the ground truth rotation annotation suffers from unavoidable ambiguity, which is caused by the ambiguity of principal axes and by part geometric symmetry. To address this issue, our approach predicts principal axes of oriented bounding boxes in an orderless fashion. Additionally, a mixture of experts is applied to predict multiple rotations and the confidence to select each. The association of multiple experts further improves the performance. From our experiments, the proposed method accelerates training and yields a much smaller angle error.

To address the challenge of ill-posed size prediction from a single image, we estimate the part size in a group-based manner. We first group translational symmetric parts and then aggregate part features in each group to predict a unified size. Such group-level information sharing allows better reasoning and more accurate predictions, especially when perspective distortion exists.

For the position prediction, we infer the relative position between two parts instead of the absolution position. Compared to the absolute translation of each part used in [23], this relative relationship shows stronger cross-category transferability.

In summary, the key contributions in this paper are listed as follows. First, we carefully design a modularized pipeline to solve each of these sub-problems in a transferrable manner. Second, we introduce a novel orientation prediction module to address the intrinsic symmetry ambiguity issue with a mixture of experts. Third, a group-based size prediction module is introduced to mitigate the size ambiguity caused by image projection. Fourth, we demonstrate significant improvement over prior arts and provide extensive analysis to justify our designs.

2. Related Work

We review previous works on reconstructing 3D shape from single images and part-based shape modeling, along with discussions and comparisons to our approach.

Single-image 3D Shape Reconstruction. Estimating 3D geometry from 2D single images is a long-standing problem in computer vision. Due to its ill-posed nature, learning-based methods [4, 46, 53, 40, 7, 25, 14, 52, 54, 11, 3, 29, 36, 41, 60] are proven effective recently since they can learn shape priors from training data. However, most of these works assume the alignment of the training and testing data distributions, *e.g.* either from the same object category or among multiple similar categories. Even though some works, such as [7], show that their methods, if trained in a category-agnostic manner over a large set of object categories, can reconstruct 3D shapes from novel categories, there are no explicit designs to guarantee such desired properties. Tatarchenko et al. [46] demonstrated that state-of-the-art single-image 3D shape reconstruction methods tend to memorize the training shapes and perform retrieval at the testing time. In our work, we design explicit network modules to discover part relationships and shape sub-structures that are shared across different object categories to gain better reconstruction generalization across category gaps.

It is an under-explored problem to extrapolate the reconstruction capabilities over out-of-the-distribution novel categories. GenRe [62] proposes to factorize the problem into a sequence of sub-tasks – depth estimation, depth spherical map inpainting, voxel-based 3D refinement, and observes

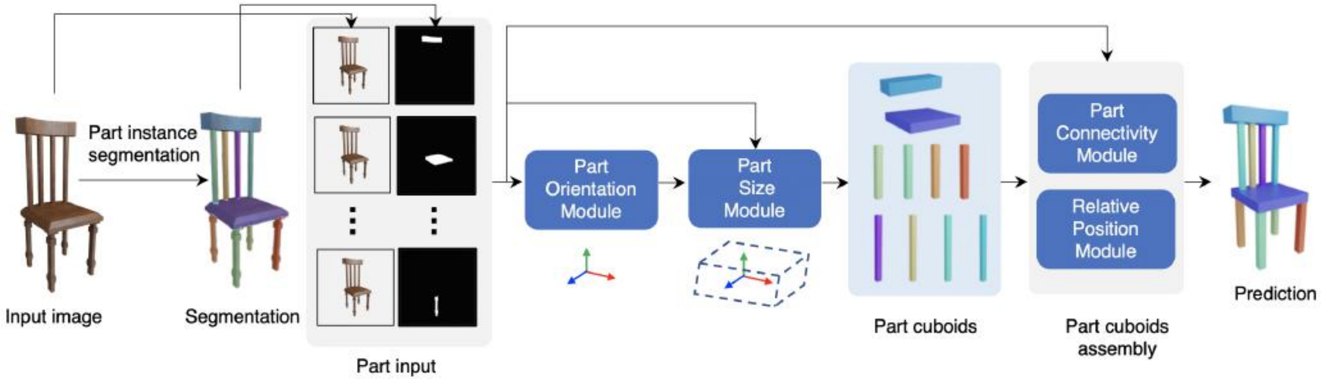

Figure 2: **Our single-view 3D structure prediction pipeline.** To extract the 3D structure from a single input image, we first apply MaskRCNN to extract the part instance masks. Then we predict each part cuboid attributes including orientation and size. Finally, we predict the pairwise connectivity and assemble the part cuboid together through relative position. Note that we heavily use part masks as module inputs to induce a focus on local regions.

better generalization capability to unseen testing object categories. GSIR [51] further extends the GenRe approach by bringing in the part structure information. Different from these two works that leverage depth as intermediate steps for generability, our approach learns to discover part relationships and shape sub-structures that are commonly shared among man-made object categories. Wallace et al. [49] and Michalkiewicz et al. [30] propose few-shot learning frameworks to quickly adapt the learned shape priors to novel categories. Our method uses a zero-shot setting that no data from the testing categories is needed at all.

Part-based Shape Structure Modeling. Most 3D shapes are composed of sub-structures and parts. While most previous works mentioned above reconstruct 3D shape geometry as a whole entity, understanding shape structures and parts is a crucial perception task for robotic manipulation [19, 28, 58], shape generation [22, 6, 42, 55, 8, 31, 21, 17, 47] and editing [20, 61, 32, 45].

There are many recent works that learn to abstract shape parts using primitive shapes, such as cuboids [48, 64, 43], superquadratics [38, 37], mixtures of Gaussians [9, 10], convex hulls [5], etc. However, these works do not explicitly consider part relationships, such as the part adjacency and symmetry constraints. In this paper, we explicitly discover and model such part relationships not only for better reconstruction results, but also for generalizing to novel categories that share similar part structures.

Previous works studied various ways for modeling part relationships and constraints. Early works [2, 18, 16] studied pairwise part relationships using probabilistic graphical models. More recently, GRASS [22], Im2Struct [35] and StructureNet [31] organize parts in tree structures and also model part adjacency and symmetry constraints among sibling parts. SAGNet [57] and SDM-Net [8] also design explicit network modules to learn pairwise part relationships for shape generation. Recent papers [24, 13] learns part relationships for part assembly via graph neural networks. Though working well within a collection of shapes from the same object category or very similar categories, most of these works failed to generalize to novel categories, which is the central goal of our work.

3. Problem Statement

Given an image I of an object O, our goal is to predict the 3D structure of O, which is represented as an assembly of geometric primitives $\{p_i\}$. In our case, we use the oriented bounding box as primitive following StructureNet [31], *i.e.*, $p_i = [c_x, c_y, c_z, s_x, s_y, s_z, q]$, where (c_x, c_y, c_z) denotes the center location, (s_x, s_y, s_z) represents the length of each edge, and $q \in \mathbb{R}^4$ corresponds to the quaternion representation of the rotation. Especially, we focus on the relative position between a pair of parts $(c_x^i - c_x^j, c_y^i - c_y^j, c_z^i - c_z^j)$. In our case, an object-level shift of center location is allowed, as long as the relative positions between pairs keep.

The learning of all the modules in our work are supervised. During training, the groundtruth part instance segmentation masks and the ground-truth parameters of the oriented bounding boxes are provided. In practice, the images are rendered from the CAD models in ShapeNet [1], and the ground-truth part decomposition and part parameters are generated according to 3D labels in PartNet [33]. The oriented bounding box of a part are computed using Pinciple Component Analysis (PCA) according to its part geometry.

4. Method

Compared to previous methods, our proposed algorithm is trained only on the chair category, yet can generalize to unseen categories, such as bed, storage furniture, and table. The key observation is that shapes across categories share similar parts and local part relationships, supporting compositional generalizability.

The overall framework is summarized in Fig. 2. Given an image, we first learn to predict part masks using MaskR-CNN [12], a well-established object instance segmentation approach (Sec 4.1). In Sec 4.2 and Sec 4.3, we introduce how to predict the direction and size of the oriented bounding box for each part. Finally, in Sec 4.4, we introduce how to assemble the predicted boxes into a complete shape.

4.1. Part Instance Segmentation

The first step of our pipeline is to obtain the mask of each part in the input image. The goal of this step is exactly the same as the classical instance-level object segmentation problem. Due to the popularity of research into instance-level segmentation, we do not treat this step as the focus of our work. Instead, we borrow existing methods to obtain the part masks. Particularly, we choose the well-established MaskR-CNN [12] and train it using the finest part masks rendered from PartNet. Please refer to Sec D.1 in the supplementary material for details.

4.2. Part Orientation Prediction

Fig 3 illustrates our part orientation prediction module, which consumes an RGB image and a binary part mask to estimate the part orientation. Specifically, a ResNet is employed to extract a global feature from the concatenation of an image and a binary mask. Then this latent feature is fed to a multi-layer perceptron which regresses the orientation.

Ambiguity of GT labels. Since it is difficult to define a canonical pose for various parts, most existing part-level datasets, like PartNet, do not provide oriented bounding box annotations. They are usually computed by the PCA given GT part segmentation. Thus, such generated labels are inevitably imperfect due to ambiguity of principal component axes and underlying part symmetry. For example, there exist 24 equivalent rotations for an equant part (cube). In addition, the PCA is also sensitive to noise in GT segmentation labels.

Orderless Principal Axes Prediction. Naively supervising the network with ambiguous and inconsistent ground truth would hinder the training seriously as illustrated in Table 4. A straightforward solution is to introduce symmetry-aware losses [50] to deal with ambiguity caused by symmetry. However, due to lack of GT symmetry annotations, heuristics are needed to distinguish different cases (equant, prolate, oblate, bladed), which is shown to be suboptimal in Sec 5.6.

To this end, we propose a novel approach to get rid of the orientation ambiguity by estimating the set of principal axes without differentiating the order and the signs. The output of our module is a rotation matrix[1], each column of which is treated as a principal axis. We consider two rotation matrices to be equivalent if the corresponding set of principal axes are the same without differentiating the signs. We supervise the network predictions through a Min-of-N MSE loss with all equivalent rotation matrices of the GT one which form a GT set. The loss is designed as the minimal mean squared error between the prediction and each rotation matrix in the GT set, denoted by $D(\hat{R}, \{R\})$. Here $\hat{R}$ is the prediction, and $\{R\}$ is the GT set. In Sec 4.3, we will discuss how to predict part sizes given a set of principal axes.

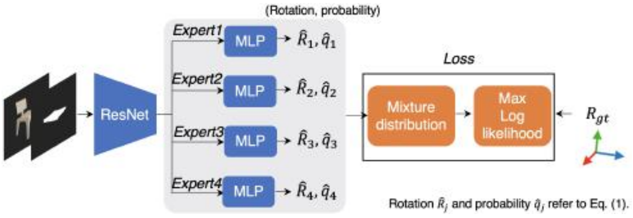

Figure 3: **Part orientation prediction module.** Given an RGB image and a part mask, we extract the global feature, from which the rotation is predicted. Specially, we predict 4 possible rotations as well as their probabilities with the Mixture of Experts. At the training stage, we maximize the log-likelihood under the rotation distribution. During inference, the rotation with the highest probability is selected.

Mixture of Experts for Rotation Symmetry and Multi-Covering of $\mathbb{SO}(3)$. However, the orderless principal axes prediction is not enough to solve the problem nicely in practice. Also, according to [59], it is impossible to find a continuous map from Euclidean space to $\mathbb{SO}(3)$ when symmetry exists. To this end, we introduce *Mixture of Experts* (MoE) [15]. Concretely, given the features extracted by the backbone, multiple branches (a.k.a. "experts") are applied to predict multiple rotations $\{\hat{R}_j\}$ as well as probabilities $\{\hat{q}_j\}$ to select certain branches. The loss is then updated as:

$$L_1 = \min_j D(\hat{R}_j, \{R\}).$$

Besides, we also maximize the log-likelihood of the predictions under a mixture of Laplacian distributions, which is formulated as:

$$L_2 = \log \sum_j \hat{q}_j \frac{1}{2b} e^{-\frac{D(\hat{R}_j, \{R\})}{b}}. \tag{1}$$

The overall training loss is $L_1 + \lambda L_2$. During inference, we select the branch with the maximum probability $\hat{q}_j$ to predict $\hat{R}_j$. We summarize the innovations of our design here and include details in Sec D.3 of the supplementary material. We provide qualitative results of rotation prediction in supplementary material Fig. 19.

[1] The network outputs a quaternion, which is converted to a rotation matrix. Other representations [63], are also tried and show similar results.

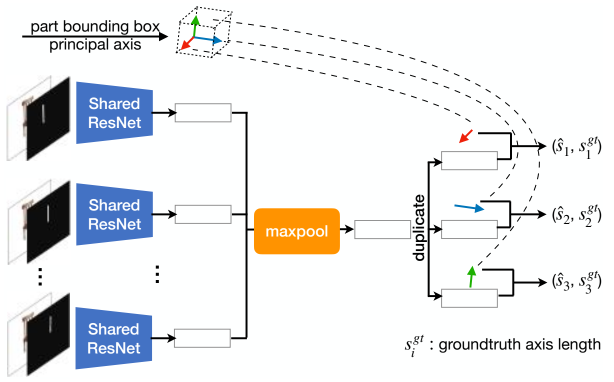

Figure 4: **Part size prediction module.** The module is designed to predict the size (edge length of bounding box) of parts in each translational symmetric group. We first extract a per-part feature with a shared ResNet and then aggregate part features through the max-pooling operation. A shared size is predicted given the aggregated group feature.

4.3. Part Size Prediction

Since a set of principal axes (order-invariant) is estimated by our part orientation prediction module, we predict the part size in an orderless fashion accordingly. Concretely, the edge length of the bounding box along each principal axis is predicted. In addition, we also propose group-based size prediction to mitigate the size ambiguity of a single image caused by perspective projection.

Group-based Size Prediction. In this work, we form part groups by considering translational symmetry. Specifically, all the parts are grouped into several clusters, so that parts in the same cluster can be translated to each other. The clustering scheme will allow us to predict a unified size for all the member parts in each cluster. Predicting part sizes in clusters brings at least three advantages: (1) Compared with predicting part sizes individually, we have more evidences from the input. Particularly, it significantly helps reduce the ambiguity of object sizes under perspective projection; (2) Some parts may not be fully visible due to occlusions, and predicting their sizes individually is very difficult; (3) Having a unified size for translational symmetric parts is visually much more pleasing for humans. The benefit of group-based prediction over unary size one is shown in Table 5.

The architecture of our group-based size prediction is illustrated in Fig. 4. We first use a shared ResNet to extract features for each part in a cluster independently. The input of each part is a concatenation of an image and a part mask. The mask induces the network to focus on the masked region. We then use a max-pooling layer to aggregate features of individual parts. Finally, the aggregated feature is duplicated three times and coupled with each principal axis predicted by the orientation prediction module (Sec 4.2). To output the length along each axis, which should be invariant to the order of axes, we follow the segmentation network in PointNet [39]. During training, each principal axis is just acquired from the ground truth. The module is supervised by the mean squared error. We summarize our design here and include details in Sec D.4 of the supplementary material. We provide qualitative results in supplementary material Fig. 20.

Translational Symmetry Classifier. We train a translational symmetry classifier to help form clusters. The classifier takes an image and a pair of part masks, to predict whether these two parts highlighted by the input masks have translational symmetry. At test time, we build clusters of parts so that the symmetry score of any part pair in the same cluster is above a predefined threshold.

4.4. Part Assembly

Having computed part mask, orientation, and size of individual parts, this module learns to assemble them into a whole shape. We predict the connectivity relationship and assemble adjacent parts by predicting the relative position between them.

Our relative position based assembly pipeline is significantly different from most work in the 3D deep learning literature. Most previous work estimates absolute location in the camera space, the world space, or a canonical object space. However, absolute locations are sensitive to the shape scale as well as the translation along the optical axis, significantly hurting the performance even for the simpler within-category prediction setup. While shape-level canonicalization is often conducted to mitigate the issue in literature, for unknown object categories, any prediction involving the undefined "canonical object space" is ill-posed.

Connectivity-based Part Tree. We train a connectivity classifier to assess whether a pair of parts touch each other in its original 3D shape. Based upon the prediction of connectivity scores between all part pairs, we heuristically sample a part tree from the connected parts. Specifically, we choose the largest part (by predicted volume) as the root node. Next, we iteratively add more parts by selecting the largest remaining part in the 1-hop neighborhood of the current tree.

Joint-based Relative Position. We assemble the adjacent parts by predicting the relative position. The input includes the reference image and the information about each individual part in the pair.

Instead of directly predicting relative center positions, we use a contact point based approach, which incorporates stronger relative position prior. Specifically, the contact point between touching part pairs must lie within the oriented bounding box of each part. In the following, we formulate how to infer the relative center position from contact point predictions (read Fig. 5 for better understanding):

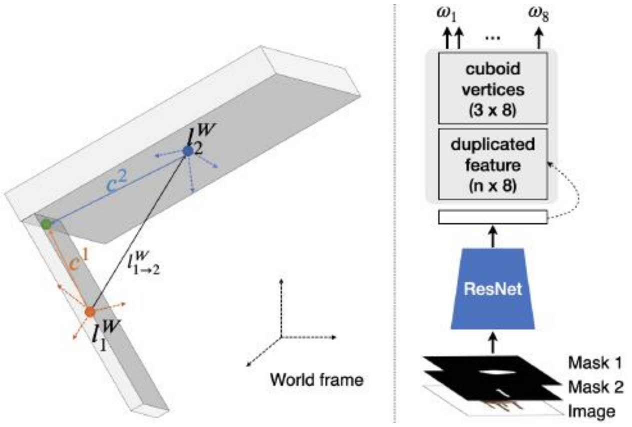

Figure 5: **Relative position prediction.** Left: Parameterize relative position between part centers (orange and blue spheres) by contact points (green sphere). Right: Network architecture illustration.

- **Contact Point.** We use superscript to denote frame and subscript to denote part indices. Given two touching parts p_1 and p_2, we denote the contact point in a part frame sitting at the center of p_1 by c^1, and in a different part frame sitting at the center of p_2 by c^2.[2] We denote the part centers in the world frame as l_1^W and l_2^W. Because the contact point is the same, we have the following constraint:

$$l_1^W + c^1 = l_2^W + c^2 \tag{2}$$

The relative center location $l_{1\to 2}^W$ can be inferred as:

$$l_{1\to 2}^W = l_2^W - l_1^W = c^1 - c^2. \tag{3}$$

Next we discuss how to estimate the contact point c^i in part frame.

- **Contact Point Estimation.** As the contact should be lying on the part surface or inside, we denote the contact points c^i as a linear interpolation of the oriented bounding box vertices $v_{i,\cdot}$: $c^i = \sum_{j=1}^{8} \omega_{i,j} \cdot v_{i,j}$, where $\sum_{j=1}^{8} \omega_{i,j} = 1$ and $\omega_{i,j} \geq 0$. We use a neural network to predict $\omega_{i,j}$'s. This network uses a softmax layer to satisfy the above two constraints. Our network consumes a stack containing the reference image and two part mask images and outputs a feature vector. We pair the feature vector with each vertex position $v_{i,j}$ of the two predicted part boxes and use another network to predict $\omega_{i,j}$'s. To make the contact point estimation invariant to the ordering of 3D bounding box vertices, the architecture of the second network is similar to the PointNet segmentation network [39]. Please refer to Sec D.5 of the supplementary material for details. Qualitative results of relative position prediction are provided in supplementary material Fig 21.

5. Experiments

We conduct experiments on PartNet [33] and compare our method with GenRe [62] and two other baselines that predict structured shapes. Our method achieves superior performance over baselines on novel categories. Ablation studies are included to validate our design of key modules.

5.1. Dataset and Settings

We use PartNet as the main testbed. It provides fine-grained object part segmentation for 26,671 3D models, covering 24 object categories in ShapeNet [1]. We pick four most commonly used object categories: chair, table, cabinet and bed. For all experiments, our method as well as baselines, are trained on the chair category only and tested on the rest three categories.

To generate the input images for training, we randomly sample 12 flying views and randomly move the shape location for each shape. Our training set contains 2,576 chairs and 60,984 rendered images in total. For the test set, we render each shape with 6 randomly sampled flying views. The rendered image is of 256 × 256 resolution.

5.2. Training Details

Our proposed method contains several individual modules. At the training stage, we train each module separately. For example, to train the joint-based relative position module, our module takes the orientation and size from ground truth labels. For testing, we directly combine these modules. Empirically, we find our pipeline good enough even without further end-to-end joint fine-tuning.

5.3. Baseline Methods

In addition to GenRe [62], we provide two other baseline methods on predicting structured shapes: a naive encoder-decoder baseline and a graph convolutional network (GCN) baseline. Note that there are no previous works addressing the exact same settings as ours. We try our best to adapt two previous works, namely StructureNet [31] and Li et al. [24], to create the two baselines.

Naive Encoder-decoder Baseline. This baseline borrows the network design from StructureNet [31]. However, StructureNet assumes a shared canonical semantic part hierarchy, which is not suitable for cross-category generalization. Thus, we remove the hierarchical part decoding stage in StructureNet and directly decode 50 part bounding boxes.

GCN Baseline. For this baseline, we use a network architecture similar to Li et al. [24], which builds part-graphs and uses the GCN to propagate information among adjacent part pairs. Concretely, for each part, we combine RGB input images and one additional channel for a 2D part mask to form a 4-channel input. To train the GCN, we use the ground truth part masks and adjacent edges.

[2] The orientation of the two frames can be arbitrarily chosen.

5.4. Evaluation Metric

To quantitatively evaluate the reconstruction results, we use the Earth-Mover Distance (EMD) [7]. It is a widely-used metric for comparing shape quality in 3D space. To measure the shape distance against the GT point cloud, we generate a point cloud based on our part bounding box representation. First, 1024 points are uniformly sampled from each estimated part bounding box. Then, 1024 points are sampled from the union of sampled part points in the previous step by the farthest point sampling.

5.5. Results and Analysis

In this section, we provide both qualitative and quantitative comparisons with baseline methods. Table 1 and Table 2 compare our results to the GCN baseline and the naive encoder-decoder baseline respectively. We observe that all methods work well on the training category – chair, while our method achieves significantly better results on novel test categories – table, cabinet, and bed.

Furthermore, the comparison with the GCN baseline demonstrates the effectiveness of our modular pipeline. For the comparison with the GCN, both the baseline and our method take the ground truth part mask and part adjacent relationship. Even with the ground truth part masks, the GCN fails to generalize to a novel category, which implies that explicit part inputs can not guarantee the compositional generalizability. With our carefully designed modules that encourage focusing on local structures, significantly better results are achieved in novel categories. We illustrate this finding in a qualitative comparison in Fig. 7. On the chair example, baseline methods achieve very competitive results compared to our method. However, the two baseline methods fail to generate satisfying results on novel categories (tables, beds, and cabinets), while our method is still able to do high-fidelity structure prediction.

Besides, we provide a qualitative (Fig. 8) and quantitative (Table 3) comparison to GenRe [62]. Note GenRe has a different setting from our method. GenRe outputs a mesh surface, while we output the part bounding boxes. Nonetheless, we provide comparison based on EMD in Table 3 using point cloud sampled from surfaces, which do not favor our structured part bounding boxes. Thanks to the structure prior, our method outperforms GenRe by a large margin. The qualitative comparison to GenRe in Fig. 8 shows that our method recovers the shape structure more faithfully.

5.6. Ablation Studies

We conduct several ablation experiments to demonstrate the effectiveness of our key components.

Part Orientation Prediction. Table 4 compares different losses w.r.t the average geodesic error between GT and predicted rotation matrices. *MSE* stands for mean squared error, which does not consider rotation ambiguity. It performs much worse than other losses, which indicates that rotation ambiguity hurts the learning process. With *MinN-MSE*, we achieve significantly better rotation prediction. Besides, *MOE-MinN-MSE* outperforms *MinN-MSE* on all categories, indicating that multiple predictions facilitate the network to escape poor local minima. For *MOE-Prior-MSE*, we use a heuristic algorithm to label part symmetry axes. Compared to our *MOE-MinN-MSE*, the performance is worse.

	Chair ↓	Table ↓	Cabinet ↓	Bed ↓
GCN	0.026	0.073	0.080	0.053
Ours (GT mask)	**0.017**	**0.048**	**0.053**	**0.033**

Table 1: **Quantitative comparisons with the GCN baseline.** (EMD on point clouds sampled from part bounding boxes.)

	Chair ↓	Table ↓	Cabinet ↓	Bed ↓
Naive	**0.026**	0.091	0.095	0.057
Ours (predicted mask)	0.028	**0.054**	**0.060**	**0.042**

Table 2: **Quantitative comparison with the naive encoder-decoder baseline.** (EMD on point clouds sampled from part bounding boxes.)

	Chair ↓	Table ↓	Cabinet ↓	Bed ↓
GenRe [62]	0.056	0.099	0.081	0.075
Ours (predicted mask)	**0.032**	**0.053**	**0.053**	**0.041**

Table 3: **Quantitative comparison with GenRe [62].** (*EMD on surface point clouds.*)

Loss	Chair ↓	Table ↓	Cabinet ↓	Bed ↓
MSE	38.03	39.43	39.82	38.98
MinN-MSE	9.21	13.74	8.29	10.81
MOE-Prior-MSE	9.85	14.10	9.75	11.35
MOE-MinN-MSE	**7.11**	**10.31**	**3.42**	**6.43**

Table 4: **Ablation experiments with MoE-MinN orientation prediction.** Comparison of average geodesic errors (angular difference in degrees) between ground truth and predicted rotation matrices over different losses.

Group-based Size Prediction. Table 5 compares unary and group-based size predictions. Our group-based size prediction shows consistently superior performance, compared with the unary counterpart. It implies that the group-based size prediction might benefit from information aggregated from multiple parts in the group.

Joint-based Relative Position Prediction. Table 6 compares our joint-based relative position prediction with absolute position prediction. Both approaches take images, part masks, part orientations, and part sizes as input. We achieve much more accurate results across all categories. As abso-

	Chair ↓	Table ↓	Cabinet ↓	Bed ↓
unary size	0.054	0.134	0.183	0.136
group size	**0.050**	**0.115**	**0.175**	**0.130**

Table 5: **Ablation experiments on group-based size prediction.** We report the number of average L1 distance.

	Chair ↓	Table ↓	Cabinet ↓	Bed ↓
absolute	0.035	0.080	0.082	0.067
relative (ours)	**0.023**	**0.038**	**0.045**	**0.042**

Table 6: **Ablation experiments on relative position prediction.** Comparison of our relative position prediction to absolute position prediction with L1 distance.

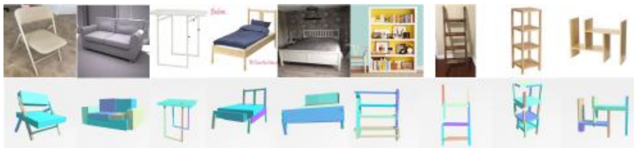

Figure 6: **Qualitative examples on real images** (best viewed zoom-in). Note that the object is masked out by the GT silhouette.

lute locations are sensitive to shape scales and translations, predicting absolute location is a more difficult problem than relative position.

5.7. Real World Data

Examples of 3D reconstruction results on real world images are shown in Fig 6. We use images from Pix3D [44] and the Internet. The object is masked out by the ground truth silhouette. Our model shows satisfactory results on real world images, even though only trained on synthetic ones.

6. Conclusion

In this paper, we first bring the concept of compositional generalizability to the task of estimating 3D shape structures from a single 2D image input and demonstrate much better reconstruction capabilities that generalize to the shapes from unseen object categories. At the core of our design, we factorize the whole problem into sub-problems that explicitly discover the commonly shared part relationships and shape substructure priors across different object categories. We carefully design the network module for each of the sub-problem, which uses localized contexts for the generalizability guarantee. Experiments show that our method is effective on generalizing to novel categories.

References

[1] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 3, 6

[2] Siddhartha Chaudhuri, Evangelos Kalogerakis, Leonidas J. Guibas, and Vladlen Koltun. Probabilistic reasoning for assembly-based 3d modeling. In *ACM Trans. Graph.*, 2011. 3

[3] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1, 2

[4] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3D-R2N2: A unified approach for single and multi-view 3d object reconstruction. In *Eur. Conf. Comput. Vis.*, 2016. 1, 2

[5] Boyang Deng, Kyle Genova, Soroosh Yazdani, Sofien Bouaziz, Geoffrey Hinton, and Andrea Tagliasacchi. Cvxnet: Learnable convex decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 31–44, 2020. 3

[6] Anastasia Dubrovina, Fei Xia, Panos Achlioptas, Mira Shalah, Raphaël Groscot, and Leonidas J Guibas. Composite shape modeling via latent space factorization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8140–8149, 2019. 3

[7] Haoqiang Fan, Hao Su, and Leonidas Guibas. A point set generation network for 3d object reconstruction from a single image. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 1, 2, 7

[8] Lin Gao, Jie Yang, Tong Wu, Yu-Jie Yuan, Hongbo Fu, Yu-Kun Lai, and Hao Zhang. SDM-NET: Deep generative network for structured deformable mesh. In *ACM Trans. Graph.*, 2019. 3

[9] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3d shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4857–4866, 2020. 3

[10] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7154–7164, 2019. 3

[11] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan C. Russell, and Mathieu Aubry. AtlasNet: A papier-mâché approach to learning 3d surface generation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 2

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 4, 14

[13] Jialei Huang, Guanqi Zhan, Qingnan Fan, Kaichun Mo, Lin Shao, Baoquan Chen, Leonidas Guibas, and Hao Dong. Generative 3d part assembly via dynamic graph learning. In *The IEEE Conference on Neural Information Processing Systems (NeurIPS)*, 2020. 3

[14] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *Adv. Neural Inform. Process. Syst.*, 2018. 2

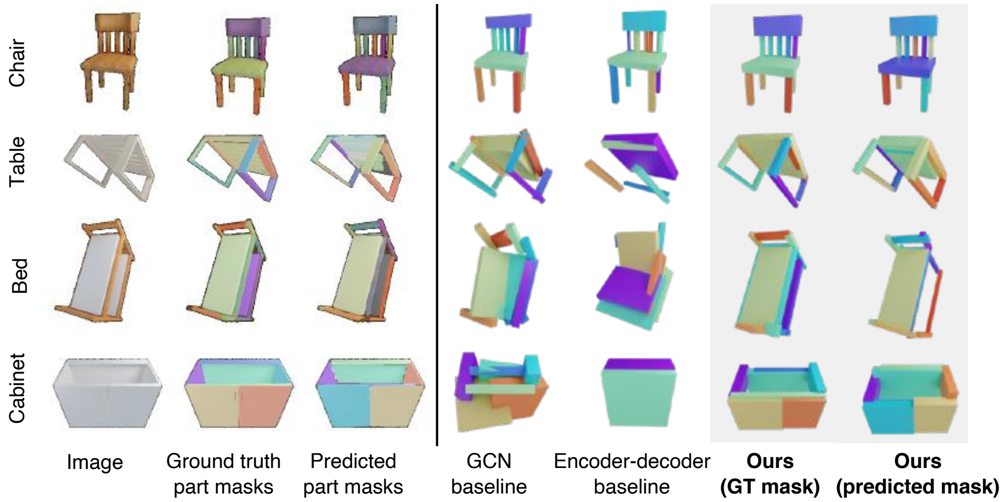

Figure 7: **Qualitative comparisons to GCN baseline and Encoder-decoder baseline.** We visualize our 2D part mask predictions and the 3D part cuboids predictions. Notice that all models are trained on chairs only. The bottom three rows show shape reconstruction from novel unseen test categories: bed, cabinet, and table.

[15] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991. 4

[16] Prakhar Jaiswal, Jinmiao Huang, and Rahul Rai. Assembly-based conceptual 3d modeling with unlabeled components using probabilistic factor graph. *Computer-Aided Design*, 2016. 3

[17] R Kenny Jones, Theresa Barton, Xianghao Xu, Kai Wang, Ellen Jiang, Paul Guerrero, Niloy J Mitra, and Daniel Ritchie. Shapeassembly: Learning to generate programs for 3d shape structure synthesis. *arXiv preprint arXiv:2009.08026*, 2020. 3

[18] Evangelos Kalogerakis, Siddhartha Chaudhuri, Daphne Koller, and Vladlen Koltun. A probabilistic model for component-based shape synthesis. *ACM Transactions on Graphics (TOG)*, 31(4):1–11, 2012. 3

[19] Dov Katz and Oliver Brock. Manipulating articulated objects with interactive perception. In *2008 IEEE International Conference on Robotics and Automation*, pages 272–277. IEEE, 2008. 3

[20] Vladislav Kraevoy, Alla Sheffer, Ariel Shamir, and Daniel Cohen-Or. Non-homogeneous resizing of complex models. *ACM Transactions on Graphics (TOG)*, 27(5):1–9, 2008. 3

[21] Jun Li, Chengjie Niu, and Kai Xu. Learning part generation and assembly for structure-aware shape synthesis. In *AAAI*, 2020. 3

[22] Jun Li, Kai Xu, Siddhartha Chaudhuri, Ersin Yumer, Hao Zhang, and Leonidas Guibas. GRASS: Generative recursive autoencoders for shape structures. In *ACM Trans. Graph.*, 2019. 3

[23] Yichen Li, Kaichun Mo, Lin Shao, Minhyuk Sung, and Leonidas Guibas. Learning 3d part assembly from a single image. *arXiv preprint arXiv:2003.09754*, 2020. 2

[24] Yichen Li, Kaichun Mo, Lin Shao, Minghyuk Sung, and Leonidas Guibas. Learning 3d part assembly from a single image. *European conference on computer vision (ECCV 2020)*, 2020. 3, 6

[25] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning efficient point cloud generation for dense 3d object reconstruction. In *AAAI*, 2018. 2

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 14

[27] Joao Loula, Marco Baroni, and Brenden M Lake. Rearranging the familiar: Testing compositional generalization in recurrent networks. *arXiv preprint arXiv:1807.07545*, 2018. 2

[28] Roberto Martín-Martín, Clemens Eppner, and Oliver Brock. The rbo dataset of articulated objects and interactions. *The International Journal of Robotics Research*, 38(9):1013–1019, 2019. 3

[29] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1, 2

[30] Mateusz Michalkiewicz, Sarah Parisot, Stavros Tsogkas, Mahsa Baktashmotlagh, Anders Eriksson, and Eugene Belilovsky. Few-shot single-view 3-d object reconstruction with compositional priors. *arXiv preprint arXiv:2004.06302*, 2020. 3

[31] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas Guibas. Structurenet: Hierarchical graph networks for 3d shape generation. *ACM Transactions on Graphics (TOG), Siggraph Asia 2019*, 38(6):Article 242, 2019. 2, 3, 6, 16

[32] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy J Mitra, and Leonidas J Guibas. Structedit: Learning structural shape variations. In *Proceedings of the IEEE/CVF*

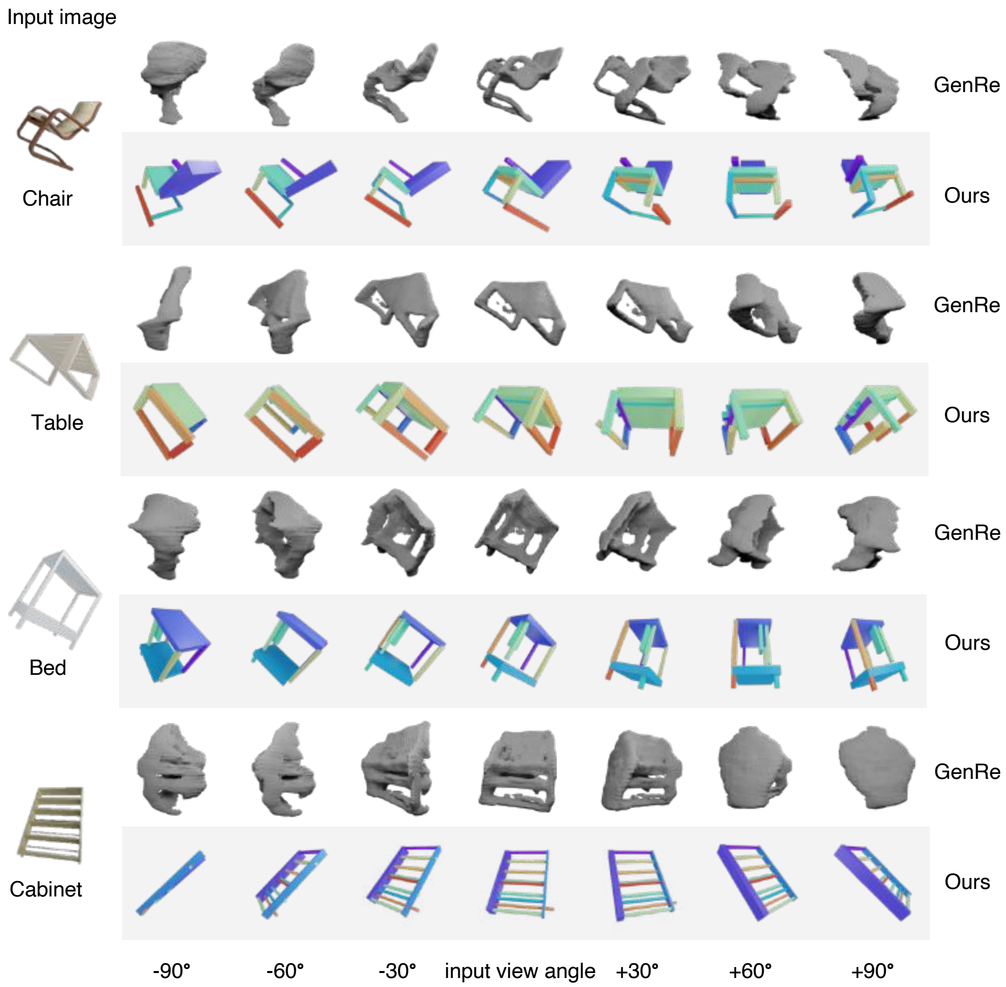

Figure 8: **Qualitative comparisons to GenRe [62].** We compare our results with predicted mask to GenRe and present each result from multiple views to clearly show the 3D reconstruction. While the results from GenRe look good from the input image views (the fifth column), the 3D reconstruction of GenRe is actually worse than ours as compared from other views.

Conference on Computer Vision and Pattern Recognition, pages 8859–8868, 2020. 2, 3

[33] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 909–918, 2019. 3, 6, 14

[34] Tongzhou Mu, Jiayuan Gu, Zhiwei Jia, Hao Tang, and Hao Su. Refactoring policy for compositional generalizability using self-supervised object proposals. *Advances in Neural Information Processing Systems*, 33, 2020. 2

[35] Chengjie Niu, Jun Li, and Kai Xu. Im2struct: Recovering 3d shape structure from a single rgb image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4521–4529, 2018. 3

[36] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2019. 1, 2

[37] Despoina Paschalidou, Luc Van Gool, and Andreas Geiger. Learning unsupervised hierarchical part decomposition of 3d objects from a single rgb image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1060–1070, 2020. 3

[38] Despoina Paschalidou, Ali Osman Ulusoy, and Andreas Geiger. Superquadrics revisited: Learning 3d shape parsing beyond cuboids. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10344–10353, 2019. 3

[39] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 5, 6, 14, 15

[40] Stephan R Richter and Stefan Roth. Matryoshka networks:

Predicting 3D geometry via nested shape layers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018. 2

[41] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. PIFu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Int. Conf. Comput. Vis.*, 2019. 2

[42] Nadav Schor, Oren Katzir, Hao Zhang, and Daniel Cohen-Or. CompoNet: Learning to generate the unseen by part synthesis and composition. In *Int. Conf. Comput. Vis.*, 2019. 3

[43] Chun-Yu Sun, Qian-Fang Zou, Xin Tong, and Yang Liu. Learning adaptive hierarchical cuboid abstractions of 3d shape collections. *ACM Transactions on Graphics (TOG)*, 38(6):1–13, 2019. 3

[44] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 8

[45] Minhyuk Sung, Zhenyu Jiang, Panos Achlioptas, Niloy J Mitra, and Leonidas J Guibas. Deformsyncnet: Deformation transfer via synchronized shape deformation spaces. *arXiv preprint arXiv:2009.01456*, 2020. 3

[46] M. Tatarchenko, A. Dosovitskiy, and T. Brox. Octree generating networks: Efficient convolutional architectures for high-resolution 3d outputs. In *Int. Conf. Comput. Vis.*, 2017. 2

[47] Yonglong Tian, Andrew Luo, Xingyuan Sun, Kevin Ellis, William T Freeman, Joshua B Tenenbaum, and Jiajun Wu. Learning to infer and execute 3d shape programs. *arXiv preprint arXiv:1901.02875*, 2019. 3

[48] Shubham Tulsiani, Hao Su, Leonidas J Guibas, Alexei A Efros, and Jitendra Malik. Learning shape abstractions by assembling volumetric primitives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2635–2643, 2017. 2, 3

[49] Bram Wallace and Bharath Hariharan. Few-shot generalization for single-image 3d reconstruction via priors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3818–3827, 2019. 3

[50] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2642–2651, 2019. 4

[51] Jianren Wang and Zhaoyuan Fang. Gsir: Generalizable 3d shape interpretation and reconstruction. *European conference on computer vision (ECCV 2020)*, 2020. 1, 3

[52] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Wei Liu, and Yu-Gang Jiang. Pixel2mesh: Generating 3d mesh models from single rgb images. In *Eur. Conf. Comput. Vis.*, 2018. 1, 2

[53] Peng-Shuai Wang, Yang Liu, Yu-Xiao Guo, Chun-Yu Sun, and Xin Tong. Adaptive O-CNN: A Patch-based Deep Representation of 3D Shapes. In *ACM Trans. Graph.*, 2018. 2

[54] Chao Wen, Yinda Zhang, Zhuwen Li, and Yanwei Fu. Pixel2Mesh++: Multi-view 3d mesh generation via deformation. In *Int. Conf. Comput. Vis.*, 2019. 2

[55] Rundi Wu, Yixin Zhuang, Kai Xu, Hao Zhang, and Baoquan Chen. PQ-NET: A generative part seq2seq network for 3d shapes, 2019. 3

[56] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019. 14

[57] Zhijie Wu, Xiang Wang, Di Lin, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. SAGNet: Structure-aware generative network for 3d-shape modeling. In *ACM Trans. Graph.*, 2019. 3

[58] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. Sapien: A simulated part-based interactive environment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11097–11107, 2020. 3

[59] Sitao Xiang and Hao Li. Revisiting the continuity of rotation representations in neural networks. *arXiv preprint arXiv:2006.06234*, 2020. 4

[60] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. DISN: Deep implicit surface network for high-quality single-view 3D reconstruction. In *Adv. Neural Inform. Process. Syst.*, 2019. 2

[61] Weiwei Xu, Jun Wang, KangKang Yin, Kun Zhou, Michiel Van De Panne, Falai Chen, and Baining Guo. Joint-aware manipulation of deformable models. *ACM Transactions on Graphics (TOG)*, 28(3):1–9, 2009. 3

[62] Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Josh Tenenbaum, Bill Freeman, and Jiajun Wu. Learning to reconstruct shapes from unseen classes. In *Advances in neural information processing systems*, pages 2257–2268, 2018. 1, 2, 6, 7, 10, 12, 22

[63] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. 4

[64] Chuhang Zou, Ersin Yumer, Jimei Yang, Duygu Ceylan, and Derek Hoiem. 3d-prnn: Generating shape primitives with recurrent neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 900–909, 2017. 3

Supplementary Materials

Contents

A. Overview **12**

B. More Qualitative Results **12**

C. Evaluations of Individual Modules **12**
- C.1. Part Instance Segmentation 12
- C.2. Part Orientation Prediction 12
- C.3. Part Size Prediction 12
- C.4. Part Adjacency Relationship Prediction . . . 13
- C.5. Shape Assembly Process 14

D. Network Architecture Details **14**
- D.1. Part Instance Segmentation 14
- D.2. Part Relationship Prediction 14
- D.3. Part Orientation Prediction 14
- D.4. Part Size Prediction 14
- D.5. Part Relative Position Prediction 15

E. Additional Ablation Studies **15**
- E.1. Part Orientation and Size: Joint or Sequential Predictions? 15
- E.2. Part Relative Position Prediction: Center Offset or Contact Point? 17

A. Overview

This document accompanies the main paper and provides additional qualitative results for the entire pipeline, extensive evaluations for each individual sub-module in our pipeline, elaborations on the network architecture designs, and more ablation studies. We will release our code, data, and pretrained models upon paper acceptance.

Sec. B provides more qualitative results of our method and comparisons to the three baseline methods. In Sec. C, we provide more detailed evaluations of each individual network module. Sec. D presents more details about our network architecture designs and the implementation. Sec. E includes more ablation studies, besides the three shown in the main paper, to further analyze the effectiveness of our design choices in each module.

B. More Qualitative Results

We show more qualitative results comparing our methods against the naive encoder-decoder baseline and the graph-convolution-network baseline. Figures 14, 15, 16, and 17 show more chair, table, bed, and cabinet results respectively. Notice that all methods are trained on chairs only. We see that our method achieves much better reconstruction generalizability than the baseline methods. We also present more qualitative comparisons to GenRe [62] in Fig. 18, where we can clearly see that our method reconstructs the shape structure more faithfully.

	Chair	Table	Cabinet	Bed
box AP@0.5	83.18	72.03	34.40	55.31
mask AP@0.5	79.11	66.53	25.08	47.89

Table 7: **Results of Our Part Instance Segmentation.** The metrics include bounding box AP (average precision) and mask AP with the IoU threshold 0.5.

C. Evaluations of Individual Modules

Our pipeline has many sub-network modules. Due to the page limit of the main paper, we cannot present detailed result visualization and discussion on each sub-network module. In this section, we provide qualitative and quantitative evaluations for several key network modules to illustrate the predicted results at each step in our pipeline.

Concretely, we measure the quantitative accuracy of the part instance segmentation module and part adjacency relationship module. Then, we show qualitative results and analysis on the part orientation module, the part size module, and the part relative position module.

C.1. Part Instance Segmentation

In Table 7, we evaluate our 2D part mask instance segmentation module using the standard 2D object detection metrics: bounding box average precision (AP) and mask AP. We find that, though trained on chairs only, our part instance segmentation module generalizes well to novel categories in a zero-shot fashion. Figures 14, 15, 16, and 17 (left three columns) visualizes example predictions on the four object categories: chairs (training category), beds, tables, and cabinets (test categories). We can clearly see that the 2D part mask predictions are accurate enough to support the later 3D reasoning stages of our pipeline.

C.2. Part Orientation Prediction

Fig. 19 shows qualitative results of our predicted part orientation. To visualize each part oriented bounding box (OBB), we take the estimated part rotation and use ground truth part size and position. As our orientation module predicts principal axes of OBB in an orderless fashion, we re-order the ground truth length along each axis based on the predicted axes and ground truth axes. While our model is only trained on chairs, we achieve superior results on table, bed, and cabinet categories.

C.3. Part Size Prediction

Fig. 20 presents qualitative example results of our group-based size prediction. To evaluate solely the performance of

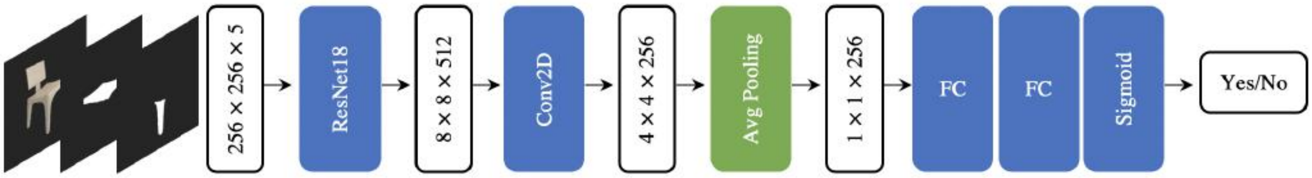

Figure 9: **Network Architecture of the Part Relationship Prediction Module.** This module is formulated as a binary classification network, which detects translational symmetry and adjacency of two parts from the image and their masks. Our part relationship module takes one image and two part masks as inputs, extract image global feature with ResNet18 without pre-training, a convolution layer, and an average-pooling layer. The global feature is then forwarded to two fully-connected layers and activated by a Sigmoid function. The output is the probability of the existence of a certain part relationship.

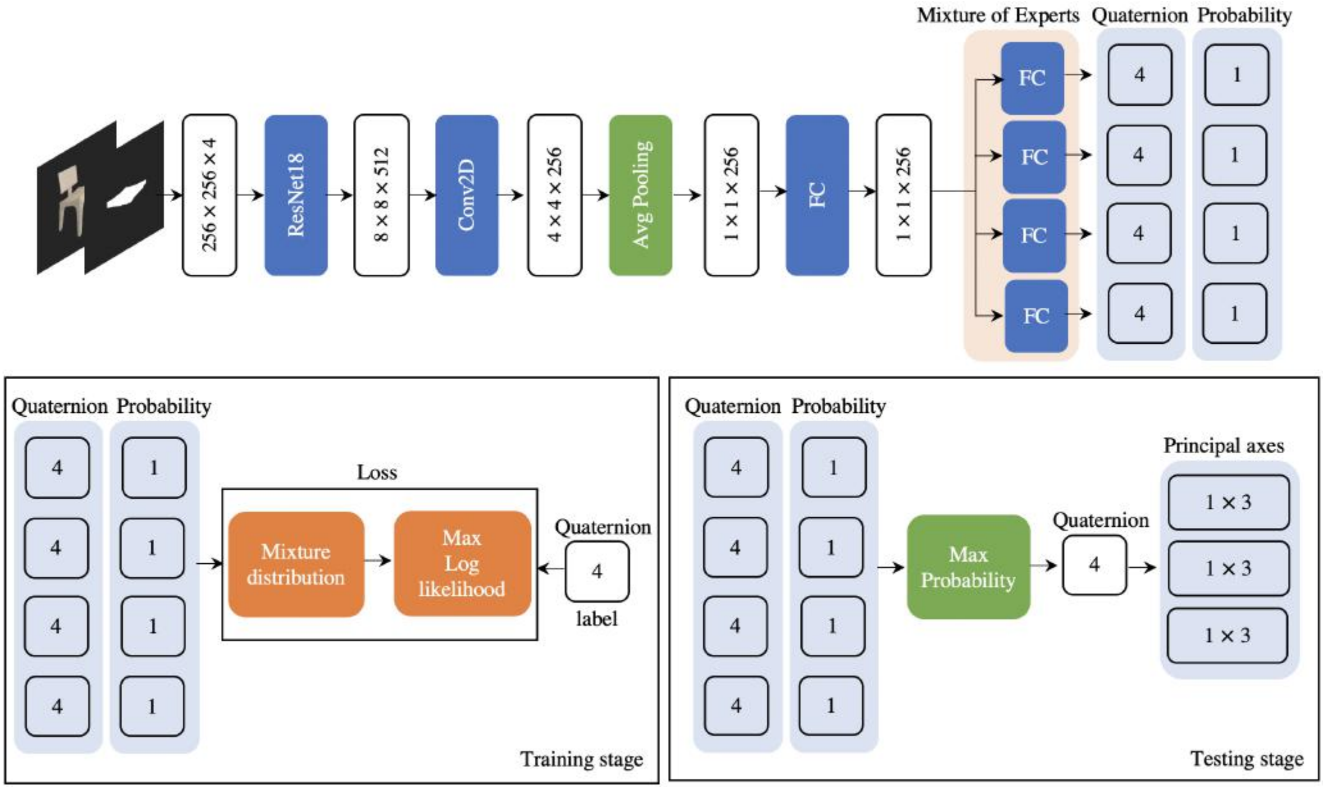

Figure 10: **Network Architecture of the Part Orientation Prediction Module.** This network takes one image and one part mask as input, and extracts a global image feature through a ResNet18, followed by a convolution layer and an average pooling layer. Then, we regress the quaternion outputs from the global feature with fully connected layers. We predict 4 rotations as well as the corresponding selection probability for the proposed Mixture-of-Experts approach. At training stage, we maximize the log likelihood under the rotation distribution. At testing stage, we pick the rotation with the maximal probability.

the group-based size module, we take ground truth orientation as input. We visualize each part with the estimated size and the ground truth orientation and position. High quality results on table, bed and cabinet demonstrate that our part size module can generalize well to the unseen test categories.

C.4. Part Adjacency Relationship Prediction

Our shape assembly process depends on a part-graph structure to form a sequential part assembly tree. We leverage part adjacency relationships as the pairwise edges connecting the neighboring parts in the part-graph. We train a network that predicts whether two parts are adjacent or not to construct such part-graph structure. Table 8 shows

	Chair	Table	Cabinet	Bed
Accuracy	0.943	0.907	0.913	0.873

Table 8: **Prediction Accuracy of Estimated Part Adjacency Relationships.** We see that the part adjacency relationships can be easily learned and generalized to unseen test categories

the quantitative evaluation of our part adjacency relationship predictions, where we see that the part adjacency relationships can be easily learned and generalized to unseen test categories.

C.5. Shape Assembly Process

Fig. 21 shows some example results for our joint-based part relative position estimation. The results here take the ground truth part orientation and size as input. Each part is visualized with the estimated position and the ground-truth orientation and size. Though our model is trained on chairs only, our position estimation module can work well on very different shapes from the cabinet, bed, and table categories.

D. Network Architecture Details

In this section, we describe more network architecture and implementation details for our network modules.

D.1. Part Instance Segmentation

We use Mask-RCNN [12] implemented in Detectron2 [56] to do part instance segmentation, which is originally designed for COCO [26] object instance segmentation. Since the aspect ratios of PartNet [33] parts follow a different distribution from those of COCO objects, we add two additional aspect ratios (1:4 and 4:1) to each anchor (predefined sliding windows). The input image is resized to 800×800, and flipped randomly as data augmentation. The detector is a ResNet50-based FPN, pretrained on COCO. It is fine-tuned on the chair category for 80,000 iterations by SGD. The batch size is 4. The initial learning rate is 0.005, and is divided by 10 after 48,000 and 64,000 iterations. Other hyperparameters follow the default setting of Detectron2.

D.2. Part Relationship Prediction

In our method, we detect translational symmetry and adjacency part relationships. The detection of relationship is modeled as a binary classification problem. The architecture of the relationship detection module is visualized in Fig. 9. The input to the network is the concatenation of an image and two part masks. A ResNet18 without pre-training followed by a convolution layer and an average-pooling layer is used to extract a global feature. The global feature is forwarded to two fully-connected layers and activated by a sigmoid function. The final output is the probability of the existence of a certain relationship between input parts. In our experiments, we use threshold 0.7 for adjacency relationship and threshold 0.9 for translation symmetry relationship. The module is trained on the chair category for 10 epochs by the Adam optimizer. The batch size is 64. The initial learning rate is 0.001, and decays by 0.9 every 2 epochs.

D.3. Part Orientation Prediction

Fig. 10 illustrates the network architecture of our part orientation prediction module. The input to the network is the concatenation of an image and a part mask. A ResNet18 without pre-training followed by a convolution layer and an average-pooling layer is used to extract a part image feature. The extracted image feature is forwarded to two fully-connected layers but with 4 different branches. This illustrates the Mixture-of-Experts (MOE) method described in main paper Sec. 4.2 (Line 366-374), through which we hope to predict a distribution of rotations instead of a single output. The final outputs of each branch are the quaternion and the probability to select this branch.

The module is trained on the chair category for 20 epochs by the Adam optimizer. The batch size is 64. The initial learning rate is 0.001, and decays by 0.7 every 2 epochs.

D.4. Part Size Prediction

For simplicity, we begin with the network architecture of single part size module which processes each part individually, and then describe how to extend it to the group-based size prediction.

Fig. 11 shows the network architecture of the single part size prediction module. The goal is to predict the edge length of the bounding box along each axis from an image, a part mask, and principal axes. First, an image feature of a part is extracted with a 2D backbone, which is composed of a ResNet-18 without pre-training, a convolution layer, and an average-pooling layer. As our orientation module predicts principal axes of the part bounding box in an orderless fashion, our size module is designed to predict an edge length for each axis. Inspired by PointNet [39], we consider each axis a 'point', and estimate its attribute (edge length) in a point-wise segmentation fashion. Such a design is permutation-equivariant to the order of axes. Concretely, we first concatenate the aggregated image feature and each axis as the input. Then, we apply a shared multilayer perceptron (MLP) to extract the feature of each axis. Following PointNet, a global feature is also extracted through the share MLP and concatenated with each local feature to enable the information communication between different axes. Finally, a scalar edge length is estimated from the feature of each edge direction.

Fig. 12 illustrates the network architecture of the extension to the group-based prediction. Given multiple parts that

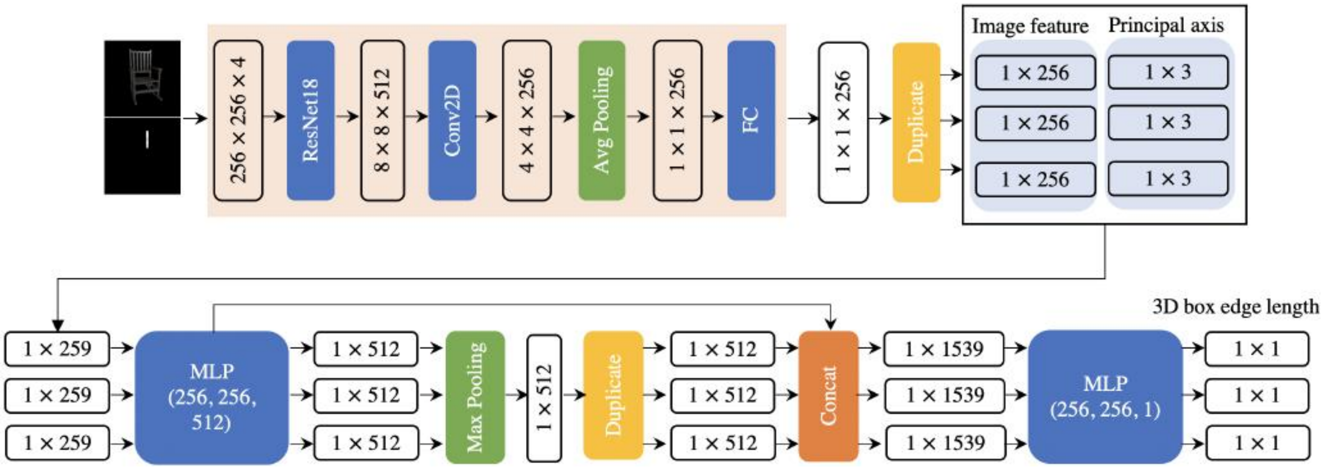

Figure 11: **Network Architecture of the Single Part Size Prediction Module.** This network takes an image, a part mask, and part principal axes (orientation) as input. The goal is to predict the edge length of part bounding box along each axis. Our size prediction module adopts a similar architecture to the PointNet segmentation network to maintain the property that edge lengths should be permutation equivariant to the order of principal axes. We consider each axis a 'point', and estimate its attribute (edge length) in a point-wise segmentation fashion. we first concatenate the image feature and each axis as the input. Then, we apply a shared MLP to extract the feature of each axis. Following PointNet, a global feature is also extracted through the share MLP and concatenated with each local feature to enable the information communication between different axes. Finally, a scalar edge length is estimated from the feature of each axis.

are considered translational symmetry, we use the 2D backbone of the single part size module to extract parts image features and then apply a max-pooling layer to aggregate all parts image features. Then we take the first part orientation as the uniform orientation for this translational symmetry part group. Again, we adopt the same architecture as the single part size module to predict a uniform size from the aggregated parts image feature and orientation. During training, we leverage the ground truth translational symmetry relationship to generate the group. During inference, it is based on the prediction of the part relationship module.

The module is trained on the chair category for 60 epochs by the Adam optimizer. The batch size is 64. The initial learning rate is 0.001 and decays by 0.9 every 2 epochs.

D.5. Part Relative Position Prediction

Fig. 13 shows the network architecture of our contact-point-based part relative position module. The network estimates the contact point of each part given the input image, part masks and the part bounding box shape, *i.e.* the orientation and size. This contact point representation is then converted to a relative position, as explained in the main paper. As the process of estimating contact point is the same for both parts in a contacting pair, we illustrate the process for one of them as an example.

The 2D input is the concatenation of an image and masks of two adjacent parts. Note the masks are ordered where the first one corresponds to the part being considered and the second one corresponds to its contacting part. A ResNet18 without pre-training followed by a convolution layer and an average-pooling layer is used to extract an image feature.

Then, the image feature is concatenated with each bounding box vertex of the part. Similar to PointNet [39] for semantic segmentation, the concatenated feature is processed by a shared MLP to obtain a weight for each bounding box vertex. By summing all the vertex positions with their weights, we can induce the contact point of the part. During training, the ground truth orientation and size are used to get the bounding box vertices. At testing stage, we take the predicted orientation and size from previous stages.

The module is trained on the chair category for 60 epochs by the Adam optimizer. The batch size is 64. The initial learning rate is 0.001, and decays by 0.9 every 2 epochs.

E. Additional Ablation Studies

We provide additional ablation studies to validate the designs of some key network modules, complementing the two presented in the main paper.

E.1. Part Orientation and Size: Joint or Sequential Predictions?

Our network design disentangles the predictions of part 3D bounding box orientation and size. Note that the order of estimated principal axes of part bounding box is tightly coupled with the order of edge length predictions along

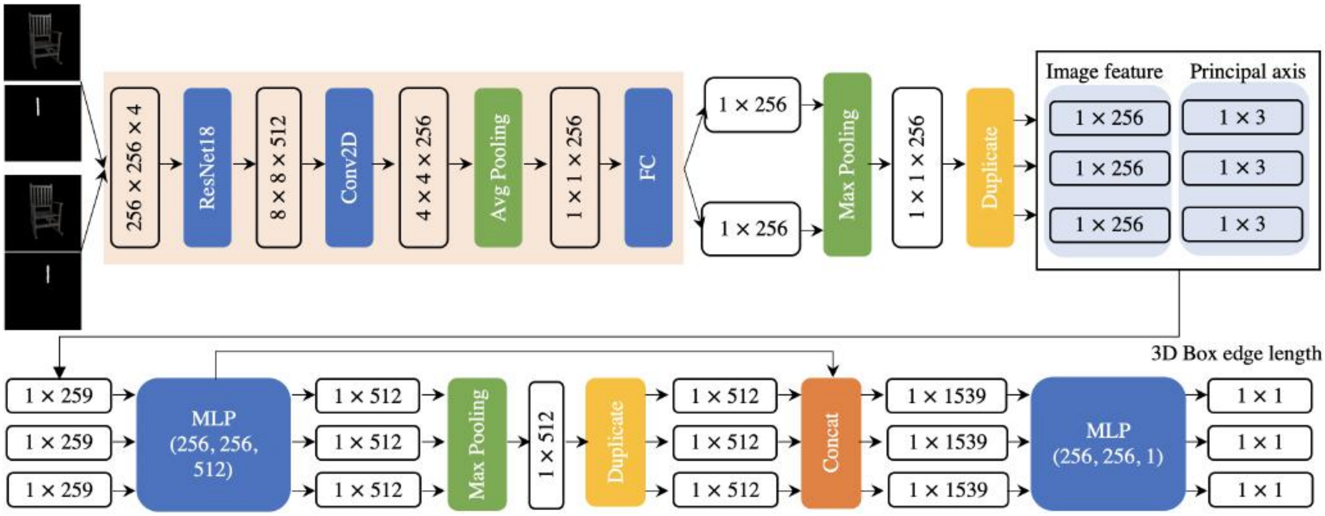

Figure 12: **Network Architecture of the Group-based Part Size Prediction Module.** The input to this module is a group of parts, which are considered to be translational symmetric. Single part image features are aggregated through max pooling, forming a grouped image feature. Then we predict edges length of the bounding box from this grouped image feature and part principal axes same as the single part size prediction.

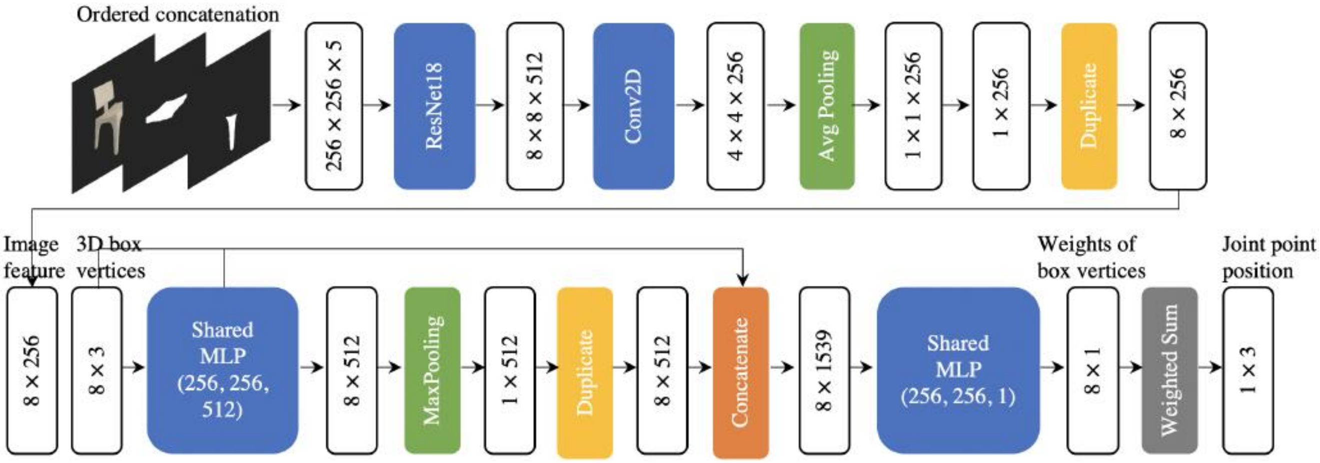

Figure 13: **Network Architecture of the Contact-point-based Part Relative Position Prediction Module.** This module estimates the contact point of both part given the image, part masks and part bounding box shape, *i.e*. orientation and size. After getting both parts contact point, we infer the part relative position to the other part, as described in main paper Sec. 4.4 (Line 512-537). This contact-point module takes image and two adjacent part masks and part bounding box attributes, i.e., orientation and size. The contact point is represented as a weighted sum of all vertex position of the part bounding box. As the weight of each vertex is order-invariant to 3D box vertices, we take a similar architecture with PointNet. Each box vertex is considered as a point. We first concatenate the aggregated image feature and each box vertex as the input. Then, we apply a shared MLP to extract the feature of each edge direction. Following PointNet,a global feature is also extracted through the share MLP and concatenated with each local feature. Finally, each vertex weight is estimated from the feature of each box vertex.

each axis. As a result, in our final pipeline, we predict part orientation and size in a sequential manner.

One alternative way is to estimate them jointly as proposed in [31]. The joint prediction is supervised by the Chamfer Distance between the ground-truth part bounding box and the bounding box scaled by the predicted size and transformed by the predicted rotation. Our experiments show that such joint prediction approach is more vulnerable to overfitting, since it allows more degrees of freedom in the output space and exploits less explicit constraints compared to our approach. Table 9 shows the quantitative comparison, where we clearly see the advantage of our sequential approach over the joint one.

	Chair	Table	Cabinet	Bed
Joint	0.026	0.146	0.161	0.179
Sequential (ours)	**0.019**	**0.105**	**0.129**	**0.089**

Table 9: **Comparison of joint and sequential predictions of part orientation and size.** We report the average Chamfer Distance between the ground truth and predicted part bounding boxes.

E.2. Part Relative Position Prediction: Center Offset or Contact Point?

One alternative to our contact-point-based part relative position prediction method, which regresses the contact point of two parts, is to directly regress an offset vector from the center of one part to that of the other. We conduct experiments to compare the performance of these two approaches. Table 10 shows the quantitative comparison w.r.t the L1 distance between the ground truth and predicted offset vectors. We argue that our contact-point-based part relative position prediction guarantees that two parts considered adjacent are contacted, which finally leads to superior performance.

	Chair	Table	Cabinet	Bed
Center Offset	0.027	0.042	0.051	0.046
Contact Point (ours)	**0.023**	**0.038**	**0.045**	**0.042**

Table 10: **Comparison of Our Contact-point-based approach to an Center-offset-based Alternative.** We report the average L1 distance between the ground truth and predicted offset vectors.

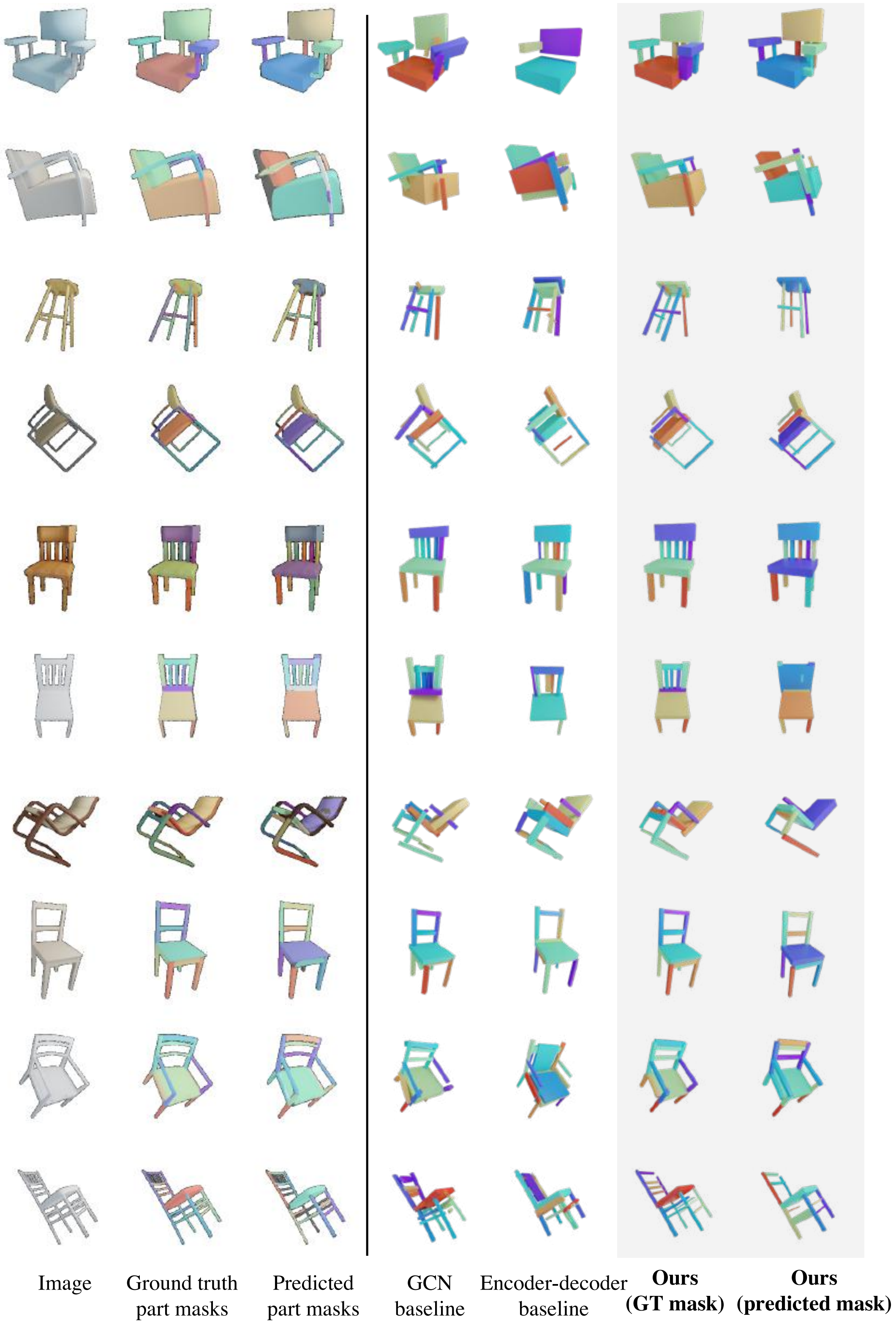

Figure 14: **More Qualitative Results and Comparisons on Chairs (training category).** From left to right, we present: input 2D image, GT 2D part mask, predicted 2D part mask, and 3D predictions of the GCN baseline, the naive encoder-decoder baseline, ours(GT mask), ours(predicted mask).

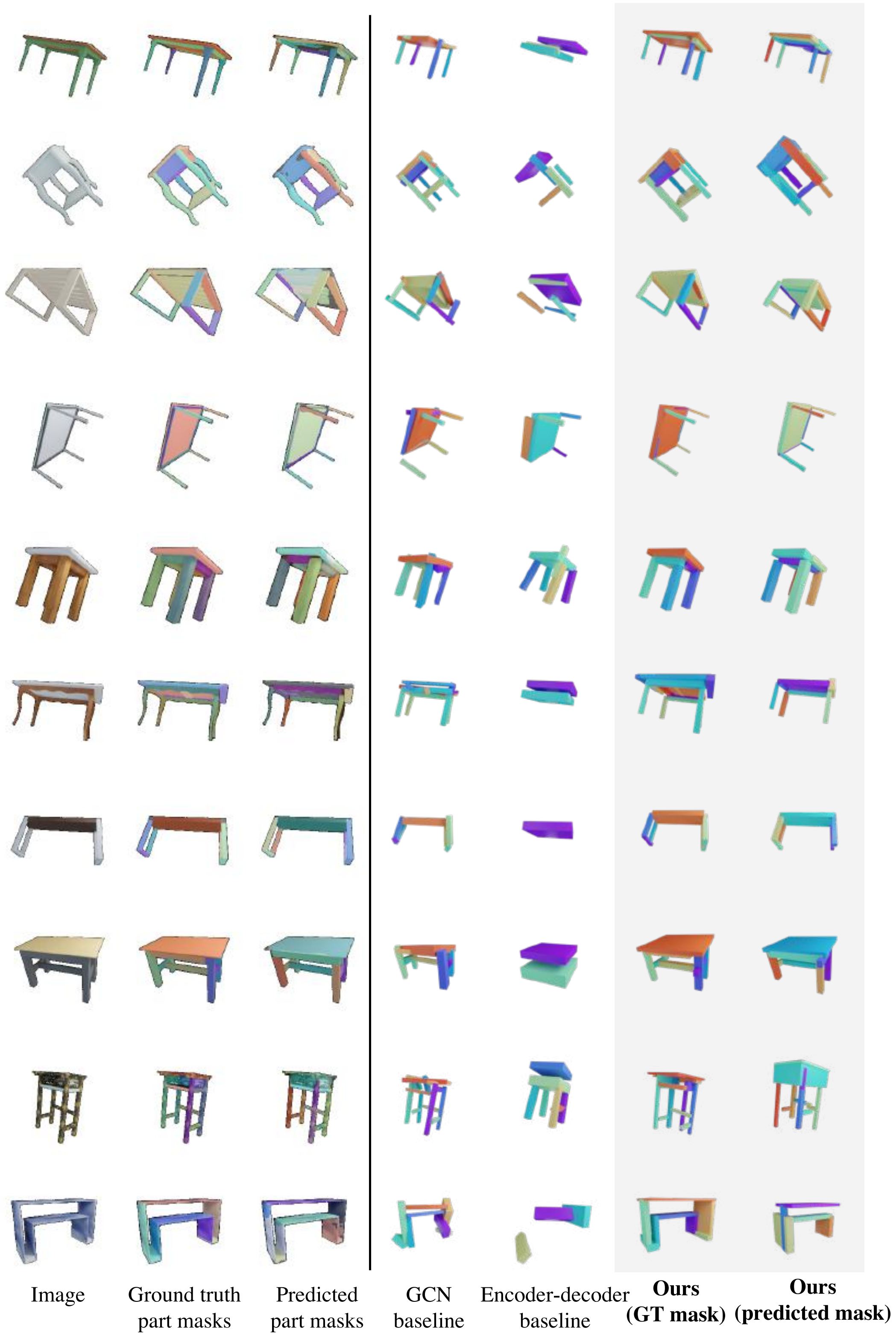

Figure 15: **More Qualitative Results and Comparisons on Tables (novel category).** From left to right, we present: input 2D image, GT 2D part mask, predicted 2D part mask, and 3D predictions of the GCN baseline, the naive encoder-decoder baseline, ours(GT mask), ours(predicted mask).

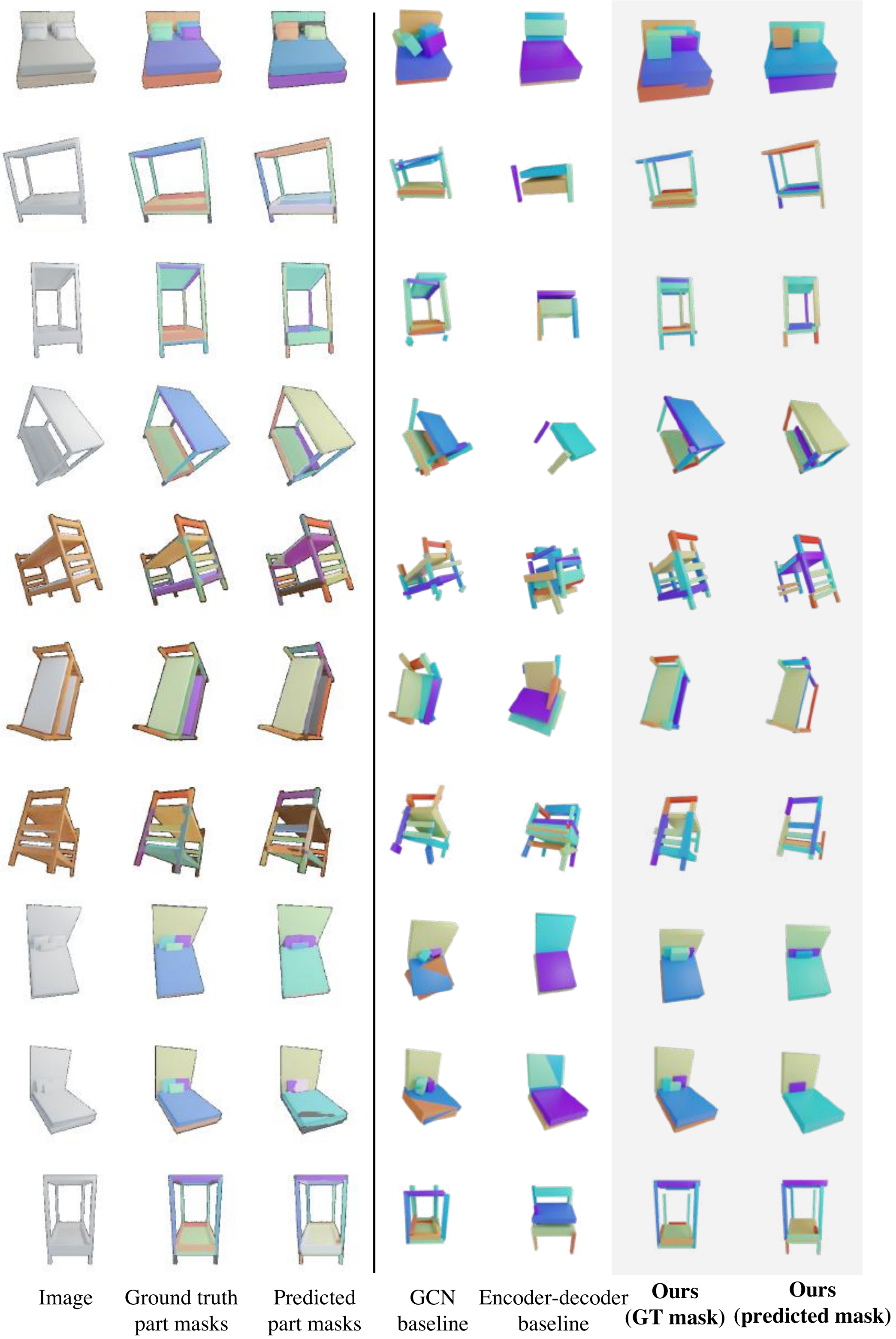

Figure 16: **More Qualitative Results and Comparisons on Beds (novel category).** From left to right, we present: input 2D image, GT 2D part mask, predicted 2D part mask, and 3D predictions of the GCN baseline, the naive encoder-decoder baseline, ours(GT mask), ours(predicted mask).

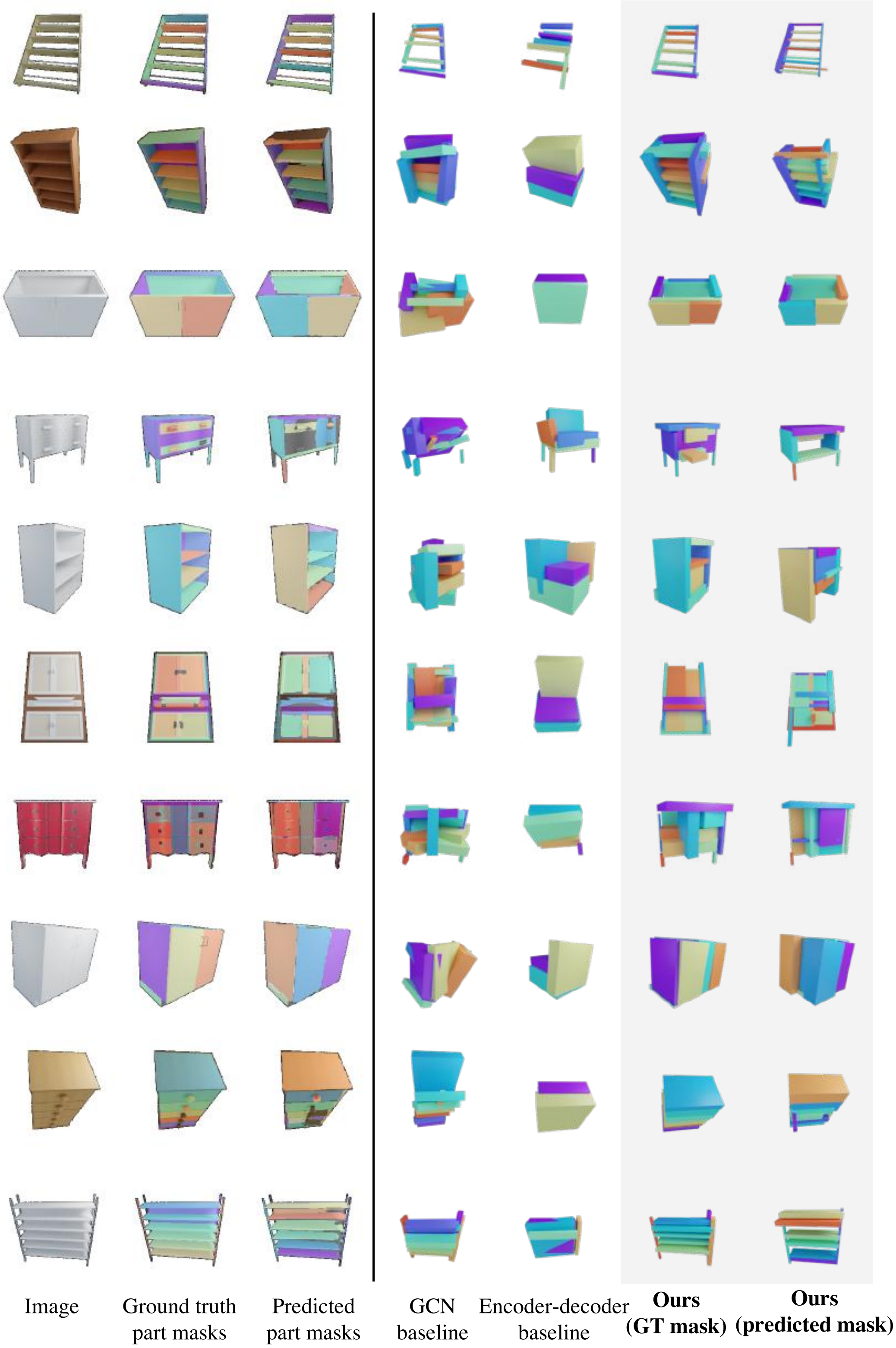

Figure 17: **More Qualitative Results and Comparisons on Cabinets (novel category).** From left to right, we present: input 2D image, GT 2D part mask, predicted 2D part mask, and 3D predictions of the GCN baseline, the naive encoder-decoder baseline, ours(GT mask), ours(predicted mask).

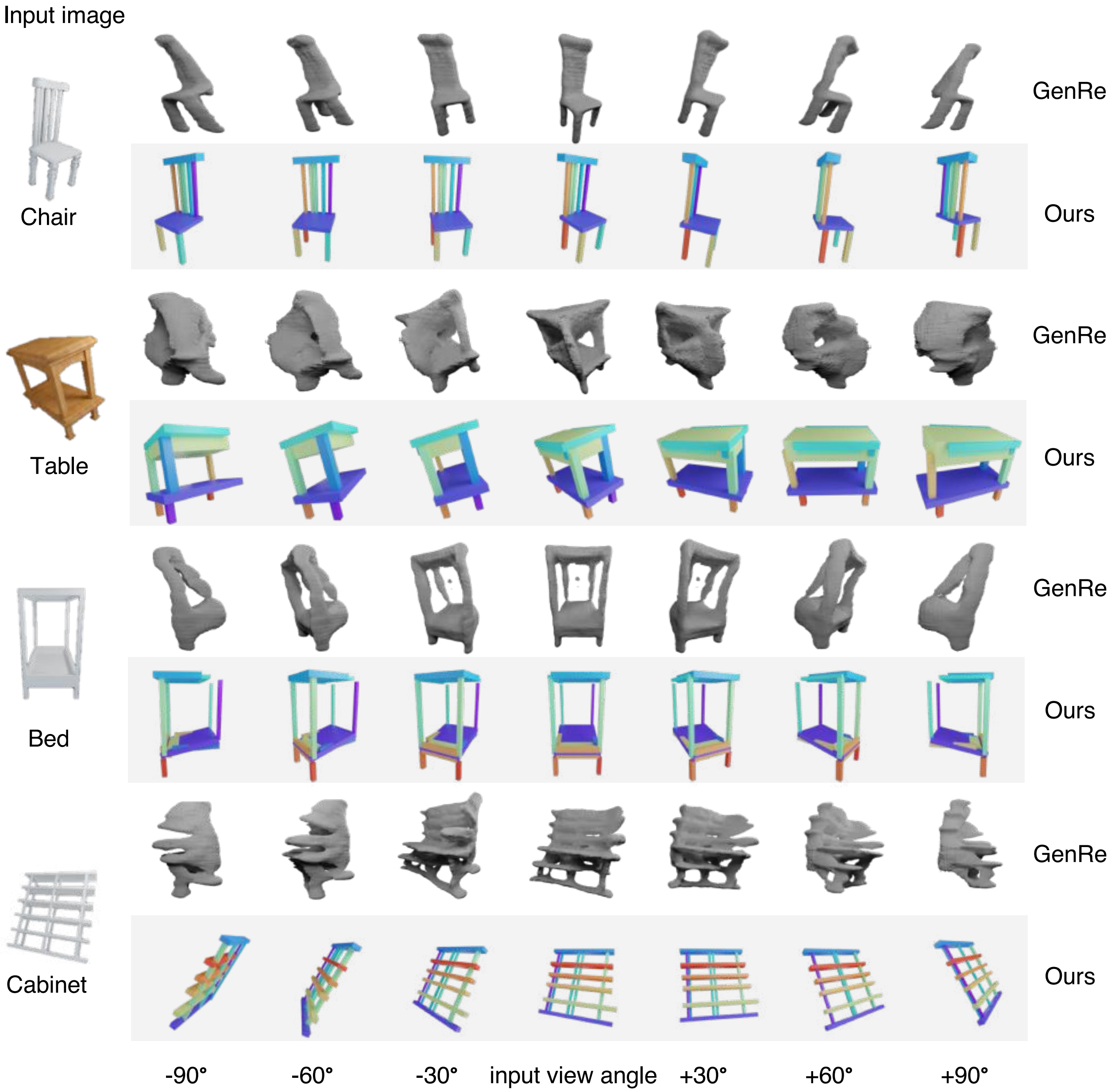

Figure 18: **More Comparisons to GenRe [62].** We compare our results to GenRe and present each result from multiple views to clearly show the 3D reconstruction. For each object category, we present three rows of results for GenRe, ours(GT mask), ours(predicted mask) respectively, with the first column shows the input image. While the result from GenRe seems to be good from the input image views (the fifth column), the 3D reconstruction of GenRe is actually worse than ours if we compare from the other views. We also see that the GenRe model trained on chairs tends to overfit to the chair shapes, while our method can faithfully reconstruct shapes from unseen test categories.

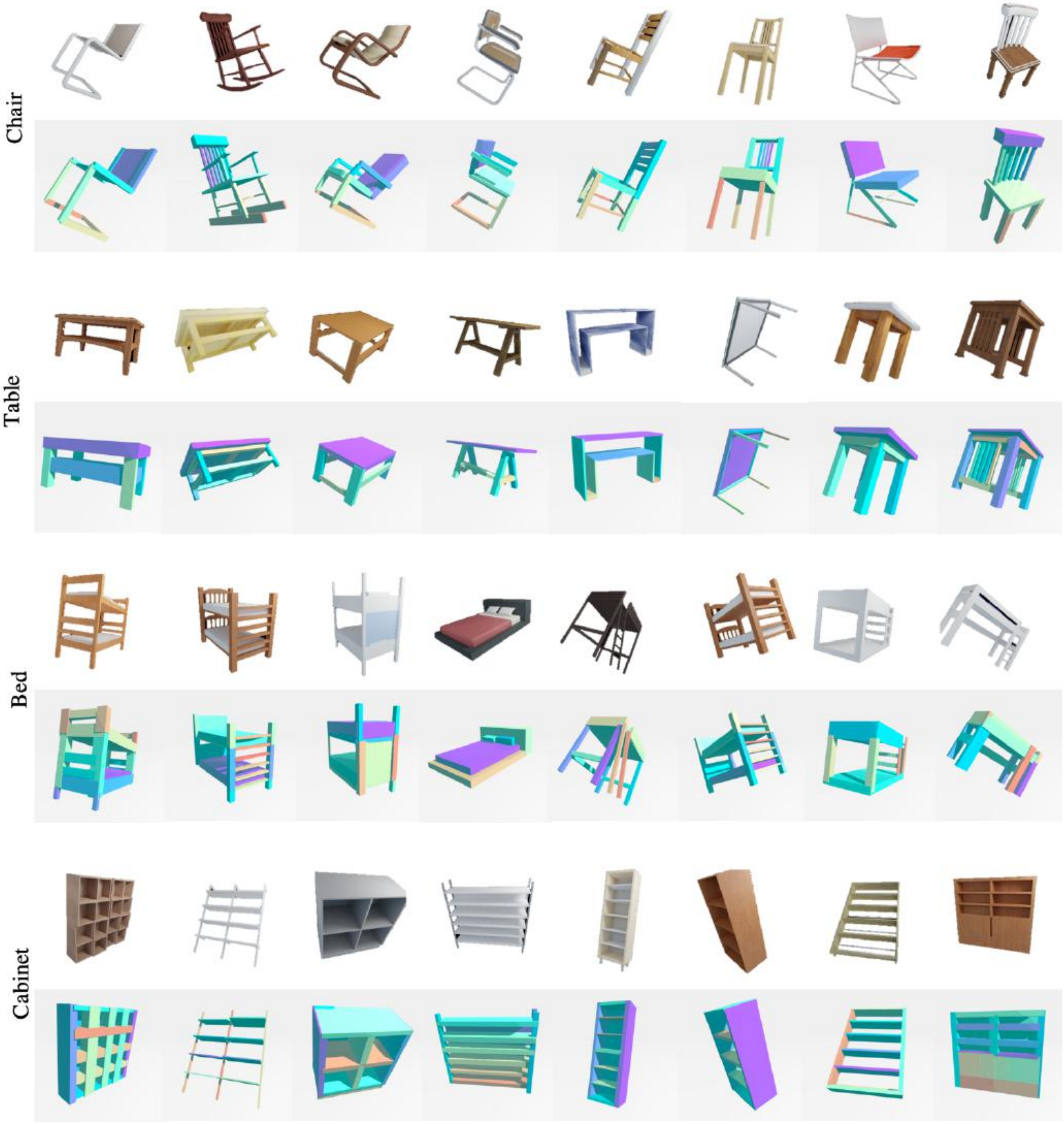

Figure 19: **Part Orientation Predictions.** For each part, we use the estimated orientation and the ground-truth size and center position. For each object category, the first row shows the input images and the second row presents the 3D part bounding boxes reconstructions.

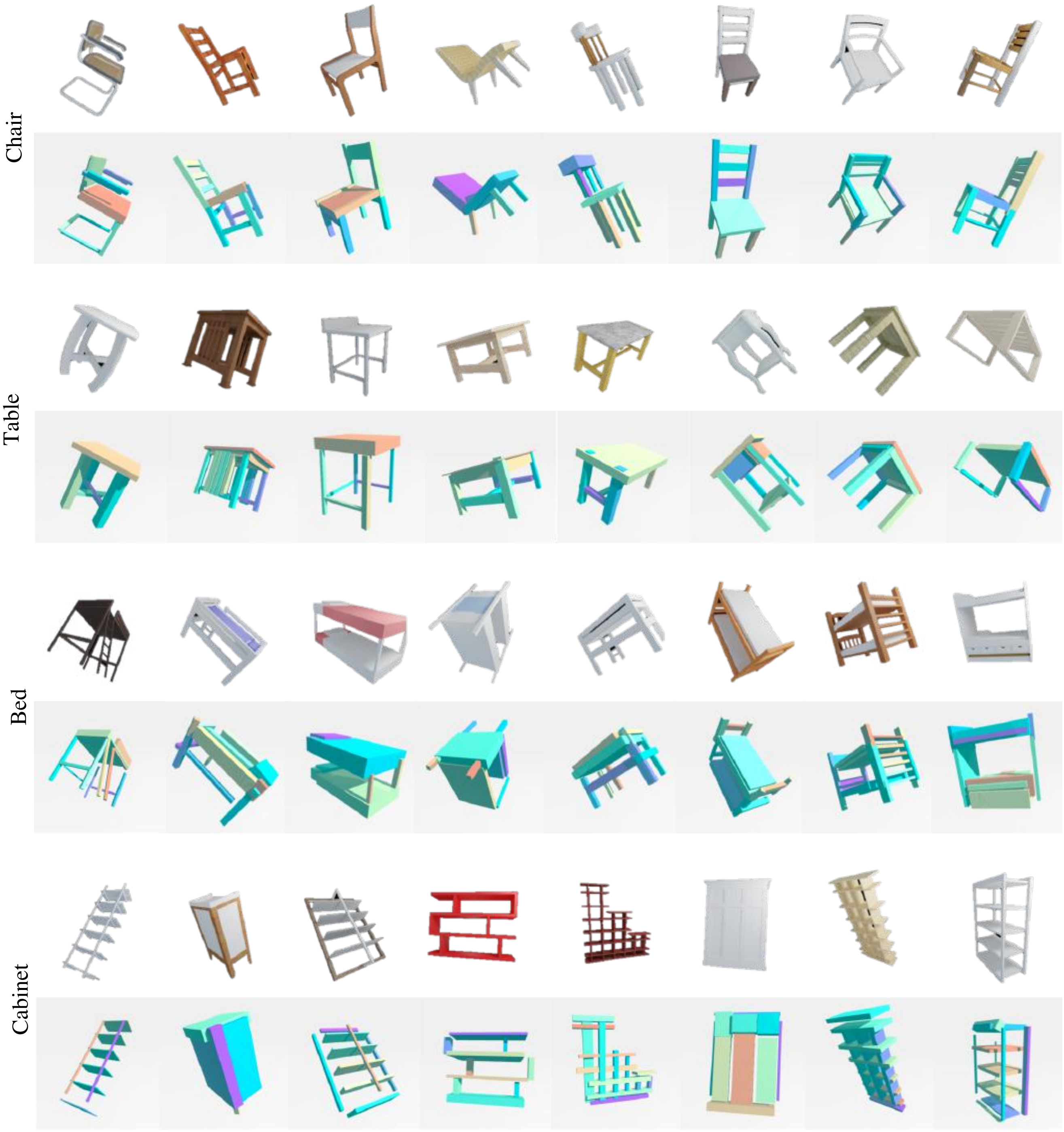

Figure 20: **Group-based Part Size Predictions.** To visualize each part, we use the estimated size and the ground truth part orientation and part position. For each object category, the first row shows the input images and the second row presents the 3D part bounding boxes reconstructions. Note that results may show disconnected parts since the part 3D box use the ground truth center positions. Our follow-up shape assembly process will connect adjacent parts.

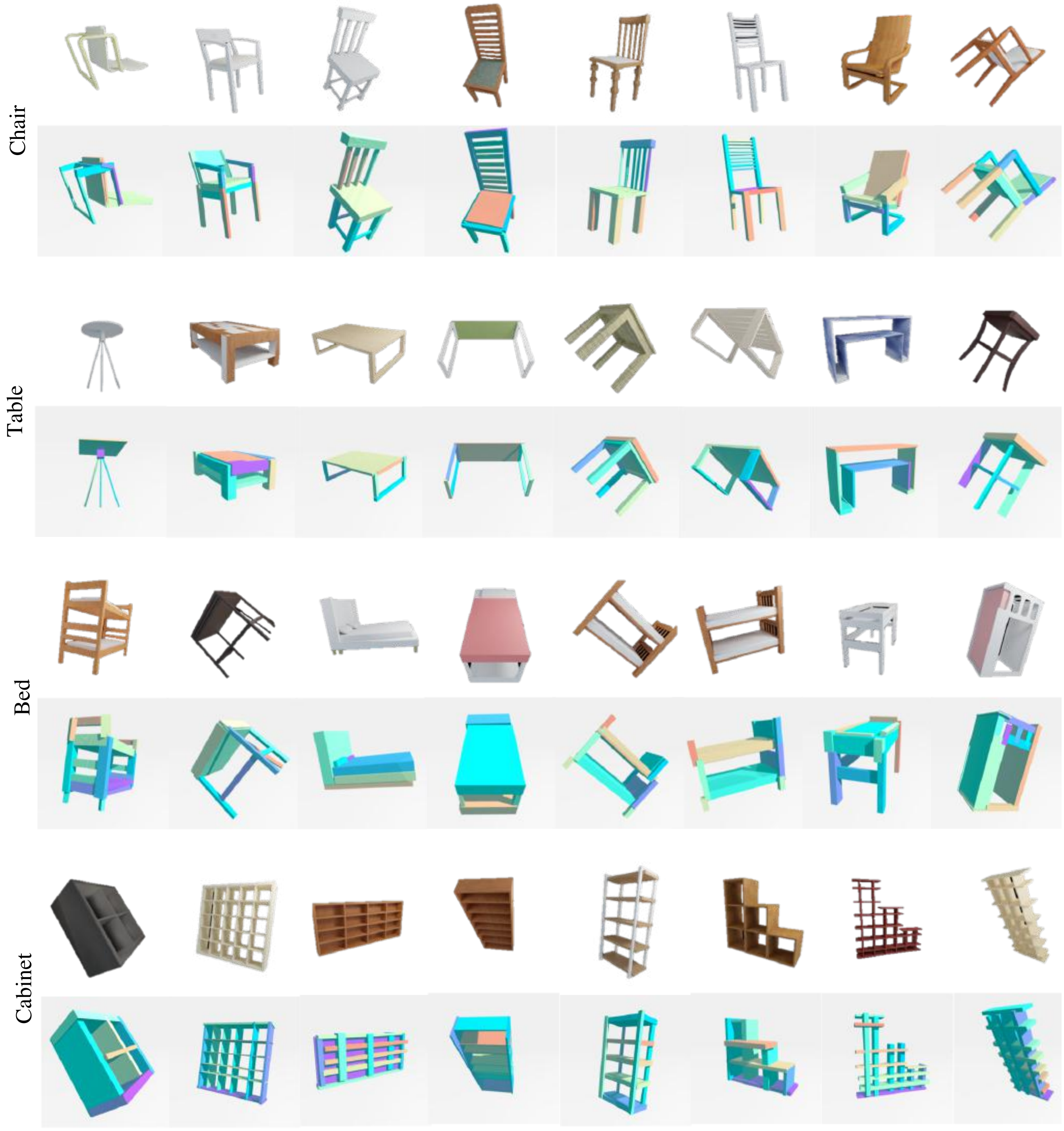

Figure 21: **Joint-based Part Relative Position Predictions.** To visualize each part, we use the estimated center position and the ground truth part orientation and length. For each object category, the first row shows the input images and the second row presents the 3D part box reconstructions.