Leveraging Pretrained Diffusion Models for Zero-Shot Part Assembly

Ruiyuan Zhang[1] , **Qi Wang**[2] , **Jiaxiang Liu**[1] , **Yuchi Huo**[1] and **Chao Wu**[1]
[1]Zhejiang University
[2]North China Electric Power University
{zhangruiyuan, zjljx, eehyc0, chao.wu}@zju.edu.cn, qiwang@ncepu.edu.cn

Abstract

3D part assembly aims to understand part relationships and predict their 6-DoF poses to construct realistic 3D shapes, addressing the growing demand for autonomous assembly, which is crucial for robots. Existing methods mainly estimate the transformation of each part by training neural networks under supervision, which requires a substantial quantity of manually labeled data. However, the high cost of data collection and the immense variability of real-world shapes and parts make traditional methods impractical for large-scale applications. In this paper, we propose first a zero-shot part assembly method that utilizes pre-trained point cloud diffusion models as discriminators in the assembly process, guiding the manipulation of parts to form realistic shapes. Specifically, we theoretically demonstrate that utilizing a diffusion model for zero-shot part assembly can be transformed into an Iterative Closest Point (ICP) process. Then, we propose a novel pushing-away strategy to address the overlap parts, thereby further enhancing the robustness of the method. To verify our work, we conduct extensive experiments and quantitative comparisons to several strong baseline methods, demonstrating the effectiveness of the proposed approach, which even surpasses the supervised learning method. The code has been released on https://github.com/Ruiyuan-Zhang/Zero-Shot-Assembly.

1 Introduction

3D part assembly autonomously assembles unordered 3D pieces into a realistic, complete object by predicting the rotations and translations of each piece. This research topic has drawn great attention in the field of robots in recent years, as it plays a crucial role in advancing robotic manipulation and automation [Chervinskii *et al.*, 2023; Ghasemipour *et al.*, 2022; Zhan *et al.*, 2020; Zhang *et al.*, 2022; Gao *et al.*, 2024].

3D part assembly is challenging because of the intricate geometries and various possible assembly combinations. The existing approach to 3D part assembly relies on training machine learning models with extensive manually annotated data, including rotations and scalings. However, the high cost of data collection makes it impractical to create datasets for each task, limiting supervised methods to well-resourced domains like common datasets. This question drove us to search for new methods to reduce reliance on manual labeling.

Diffusion models are a recent class of likelihood-based generative models that model data distributions through an iterative noising and denoising process [Ho *et al.*, 2020]. Following this, diffusion-based distillation models [Wang *et al.*, 2024b] have demonstrated significant high-fidelity 3D content generation capabilities, highlighting both their theoretical robustness and practical applicability in generating a large amount of complex 3D contents. Meanwhile, some studies have also demonstrated that a pre-trained diffusion model, leveraging its density estimates, can be transferred to handle various zero-shot tasks, including classification [Li *et al.*, 2023; Zhang *et al.*, 2024a], vision-language modeling [Liu *et al.*, 2025; Liu *et al.*, 2024; Liu *et al.*, 2023b], and singing voice synthesis [Zhang *et al.*, 2024d; Guo *et al.*, 2025; Zhang *et al.*, 2024c; Zhang *et al.*, 2024e]. Density estimation refers to the distribution of particles in space as they evolve over time during a diffusion process. These works further inspire us to explore how to distill the necessary pose transformations in assembly tasks using existing diffusion models.

In this paper, we propose a new algorithm for aligning density estimates to pose transformations. Specifically, we first introduce noise to a shape that has not been correctly positioned. This perturbed shape is then input into the diffusion model. The objective at this stage is to transform the disordered components into a distribution that is suitable for the diffusion model. By utilizing the denoising process, we can obtain a new point cloud that is closer to an accurate chair shape. However, it is important to note that this new point cloud does not represent a rigid transformation compared to the previous point cloud. To address this issue, we employ the ICP algorithm to align each part as closely as possible. By iteratively repeating this process, we can utilize the diffusion model to convert disordered parts into a complete shape, thereby accomplishing the entire assembly process. Since this process is performed explicitly, it allows us to apply direct pull-in or push-away operations for overlapping or distant parts, which is nearly impossible to achieve with other methods. To validate our method, we employed four network architectures to predict rotational and translational transformations of parts. These baselines rely on the Shape Chamfer

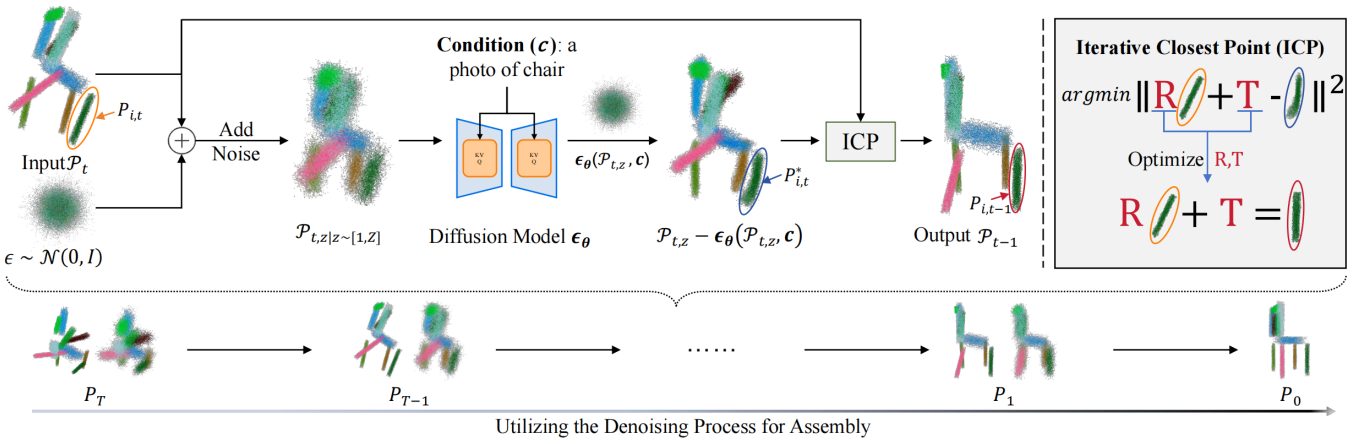

Figure 1: **The overall architecture of our algorithm.** Given the misaligned input clouds $\mathcal{P}_t$, we introduce noise to the shape, which helps the diffusion model recognize the data. The diffusion process then refines the input, generating a point cloud closer to the target chair shape. To achieve rigid transformation, we apply the ICP method for alignment, producing updated pose vectors. By iterating this process over T steps, the algorithm effectively assembles the disordered parts into the final coherent structure.

Distance (SCD) for supervised learning, aiming to approximate ground truth derived from reference samples generated by the diffusion model. Quantitative and qualitative results indicate that our method not only outperforms all baseline approaches in zero-shot settings but also surpasses some supervised techniques, underscoring its potential for practical applications.

The contributions of our paper can be summarized as follows:

- We propose the first zero-shot assembly method that utilizes density estimates from a diffusion model to achieve continuous and smooth transformations of parts, thereby coherently assembling multiple parts. Theoretical analysis within the paper supports the efficacy of this approach.
- We additionally introduce a push-away strategy to mitigate collisions between parts.
- Results show that our method outperforms all baselines in zero-shot settings and even some supervised approaches, highlighting its practical potential.

2 Related Works

2.1 3D Assembly Modeling

Estimating object pose has been a key research focus for decades. In the early research, Yoon et al. [2003] used visual sensors and neural networks for robotic assembly. Later, graph models were employed to capture semantic and geometric relationships among shape components, enabling advancements in assembly-based shape modeling [Zhan *et al.*, 2020; Jaiswal *et al.*, 2016], while a progressive strategy leveraging the recurrent graph learning framework was explored in [Narayan *et al.*, 2022]. To explore the diversity of assembly outcomes, several authors propose treating parts' poses as a distribution and achieving part assembly through a diffusion process involving noising and denoising [Xu *et al.*, 2024; Cheng *et al.*, 2023]. Furthermore, innovations in network architecture have been advancing concurrently. For instance, Zhang et al. [2024b; 2025] leverage the Transformer framework [Vaswani, 2017] to model structural relationships. Building on this, Gao et al. [2024] introduces hierarchical assembly to tackle the challenges associated with managing numerous parts. Unlike the aforementioned works that rely on manual annotations of each part's rotation and translation, our study aims to explore a novel approach to extracting the necessary pose transformations for assembly tasks. Specifically, we investigate how existing diffusion models can be leveraged to achieve this goal, thereby reducing the dependency on labour-intensive manual labelling.

2.2 Diffusion Model

Diffusion models operate in two steps: adding noise to destroy data structure and reversing this noise to reconstruct it. This enables them to model target distributions and generate diverse content, including images [Saharia *et al.*, 2022], videos [Wang *et al.*, 2024a], 3D objects [Peebles and Xie, 2023], and audio [Liu *et al.*, 2023a]. Recent studies suggest that diffusion models encode semantic and grouping information, leading to two main research directions. The first research direction leverages the internal representations of diffusion models for various discriminative tasks, requiring minimal additional training. These tasks include zero-shot classification [Li *et al.*, 2023], label-efficient segmentation [Baranchuk *et al.*, 2021], and open-vocabulary segmentation [Karazija *et al.*, 2023]. The second research direction focuses on generative tasks, such as bridging 2D diffusion models and 3D generation through Score Distillation Sampling (SDS). Methods like DreamFusion [Poole *et al.*, 2022] align 3D representations with text prompts, while later works enhance visual fidelity using strategies like coarse-to-fine optimization [Lin *et al.*, 2023] and multi-view consistency [Hu *et al.*, 2024]. These advances highlight the versatility of diffusion models in blending discriminative and generative capabilities. Our work builds upon the generative approach, introducing a theoretically sound and interpretable method to

tackle the zero-shot assembly problem effectively.

3 Methodology

In this section, we will first provide a formal symbolic definition of diffusion models (Sec. 3.1). Next, we will introduce the zero-shot method proposed in this paper (Sec. 3.2). Finally, based on our method, we will present a new approach to mitigate part overlap (Sec. 3.3).

3.1 Diffusion Model Preliminaries

The diffusion model [Song and Ermon, 2019; Luo and Hu, 2021] is a likelihood-based generative model, designed to learn the data distributions. Starting from an underlying data distribution $q(x)$, the model applies a forward process that progressively adds noise to a data sample x, creating a sequence of latent variables $\{x_z\}_{z=1}^{Z}$ governed by Gaussian transition kernels $q(x_z|x_{z-1})$. At each time step z, the marginal distribution of x_z is defined as:

$$x_z \sim q(x_z|x) = \mathcal{N}(\alpha_z x, \sigma_z^2 \mathbf{I}), \tag{1}$$

where $\sigma_z^2 + \alpha_z^2 = 1$, with σ_z gradually increasing from 0 to 1. This ensures that $q(x_z)$ converges to a Gaussian prior distribution $\mathcal{N}(0, \mathbf{I})$ as z approaches Z. Thus, $q(x_z)$ converges to a Gaussian prior distribution $\mathcal{N}(0, \mathbf{I})$.

The reverse process, which corresponds to the generative process, is designed to reconstruct the original data from a sequence of noisy observations. The conditional distribution $p_\phi(x_{z-1}|x_z)$ at each time step z is modeled as a Gaussian with mean $\mu_\phi(x_z, z)$ and covariance variance $\Sigma_\phi(x_z, z)$:

$$p_\phi(x_{z-1}|x_z) := \mathcal{N}(x_{z-1}; \mu_\phi(x_z, z), \Sigma_\phi(x_z, z)) \tag{2}$$

To ensure that the model can accurately reconstruct the original signal as it approaches the end of the generation process, $\Sigma_\phi(x_z, z)$ is typically designed to decrease as z decreases. This reflects the intuition that the model's confidence in predicting the next state should increase as it gets closer to the original data point.

Specifically, $\Sigma_\phi(x_z, z)$ can be parameterized or fixed according to a schedule that depends on the time step z. In practice, this variance term may be simplified to depend only on z, for instance, by setting it proportional to the pre-defined noise scale σ_z^2:

$$\Sigma_\phi(x_z, z) = \sigma_z^2 \mathbf{I} \tag{3}$$

where σ_z^2 is part of a predefined noise schedule that increases over time during the forward diffusion process and consequently decreases during the reverse generative process. A linear noise schedule could be defined as:

$$\sigma_z^2 = \frac{\sigma_Z^2}{Z} z \tag{4}$$

As such, when z is small, indicating that we are close to the final generation step, σ_z^2 is also small, leading to a smaller $\Sigma_\phi(x_z, z)$. This design choice ensures that the model exhibits higher certainty in its predictions as it nears the reconstruction of the original data, thereby enhancing the stability and quality of the generated samples.

3.2 Diffusion Based Iterative Zero-Shot Assembler

Denote the input point clouds as $\mathcal{P} = \{P_i \mid i = 1, \ldots, N\}$, where $P_i \in \mathbb{R}^{d\times 3}$ corresponds to the i-th part of the 3D shape, consisting of d points in the 3D space. In zero-shot task, each part of the point cloud P_i has a corresponding rigid transformation, described by a quaternion $\text{quat}_i \in \mathbb{R}^4$ and a translation vector $\text{trans}_i \in \mathbb{R}^3$, which represent the rotation and translation of the part. The goal of this task is to predict the pose parameters (quaternion quat and translation vector trans) of the test samples **without** pose information during training.

To match the current shape to the diffusion models's requirements, we introduce Gaussian noise to the current shape:

$$\mathcal{P}_{t,z} = \mathcal{N}(\alpha_z \mathcal{P}_t, \sigma_z^2 \mathbf{I}), \tag{5}$$

where t is the iterative step of our method, z is the time step in diffusion model ϵ.

By utilizing the denoising process, we can obtain a new point cloud that is closer to the shape's distribution:

$$\mathcal{P}_t^* = \mathcal{P}_{t,z} - \epsilon_\theta(\mathcal{P}_{t,z}, c), \tag{6}$$

where c corresponds to the prompt label associated with the input sample. Subsequently, to satisfy the requirements of rigid transformations, we employ ICP to obtain the vector of rotation and translation. We then apply the transformations to the input point cloud $\mathcal{P}_t$ to obtain the updated poses, which are then utilized as the input for the next iteration.

The theoretical justification for using ICP is detailed in Section 4. By iterating the above process, we can utilize the diffusion model ϵ to convert disordered parts into a complete shape, thereby accomplishing the entire assembly process.

3.3 Collision detection and handling

Given the explicit nature of our method, it facilitates the direct application of pull-in or push-away operations for either overlapping or distant parts. This strategy is very difficult to implement in existing methods due to their poses being implicitly generated by the model. To describe the pushing behavior of $\mathcal{P}_i$ in a point cloud $\mathcal{P}$ to reduce overlap with $\mathcal{P}_j$ ($i \neq j$), the overlap is quantified using $\mathcal{C}(\mathcal{P}_i, \mathcal{P}_j)$, which counts coincident points. The indicator function $\mathcal{I}(\mathcal{C}(\mathcal{P}_i, \mathcal{P}_j) < \text{threshold})$ determines whether the overlap is below a predefined threshold. The centroids of $\mathcal{P}_i$ and their intersection region are denoted as $\bar{C}_{\mathcal{P}_i}$ and $\bar{C}_{\text{intersect}}$, respectively. The displacement required to separate $\mathcal{P}_i$ is given by:

$$\Delta_i = \mathcal{I}(\mathcal{C}(\mathcal{P}_i, \mathcal{P}_j) < \text{threshold}) \cdot \left(\bar{C}_{\mathcal{P}_i} - \bar{C}_{\text{intersect}}\right) \cdot s.$$

Here, s specifies the sign of the movement. This approach computes the necessary displacement direction and distance to reduce the overlap between $\mathcal{P}_i$ and $\mathcal{P}_j$.

4 The Theory of Zero-Shot Assembly

As mentioned previously, part assembly seeks to optimize the rotation q_i and translation t_i of each part P_i to transform the unordered input into a coherent realistic object. Let the q_i and t_i be represented by an optimizable transformation matrix $\mathbf{A_i}$, then the assembly process can be formulated as:

$$\mathcal{P}_{out} = g(\mathbf{A}), \tag{7}$$

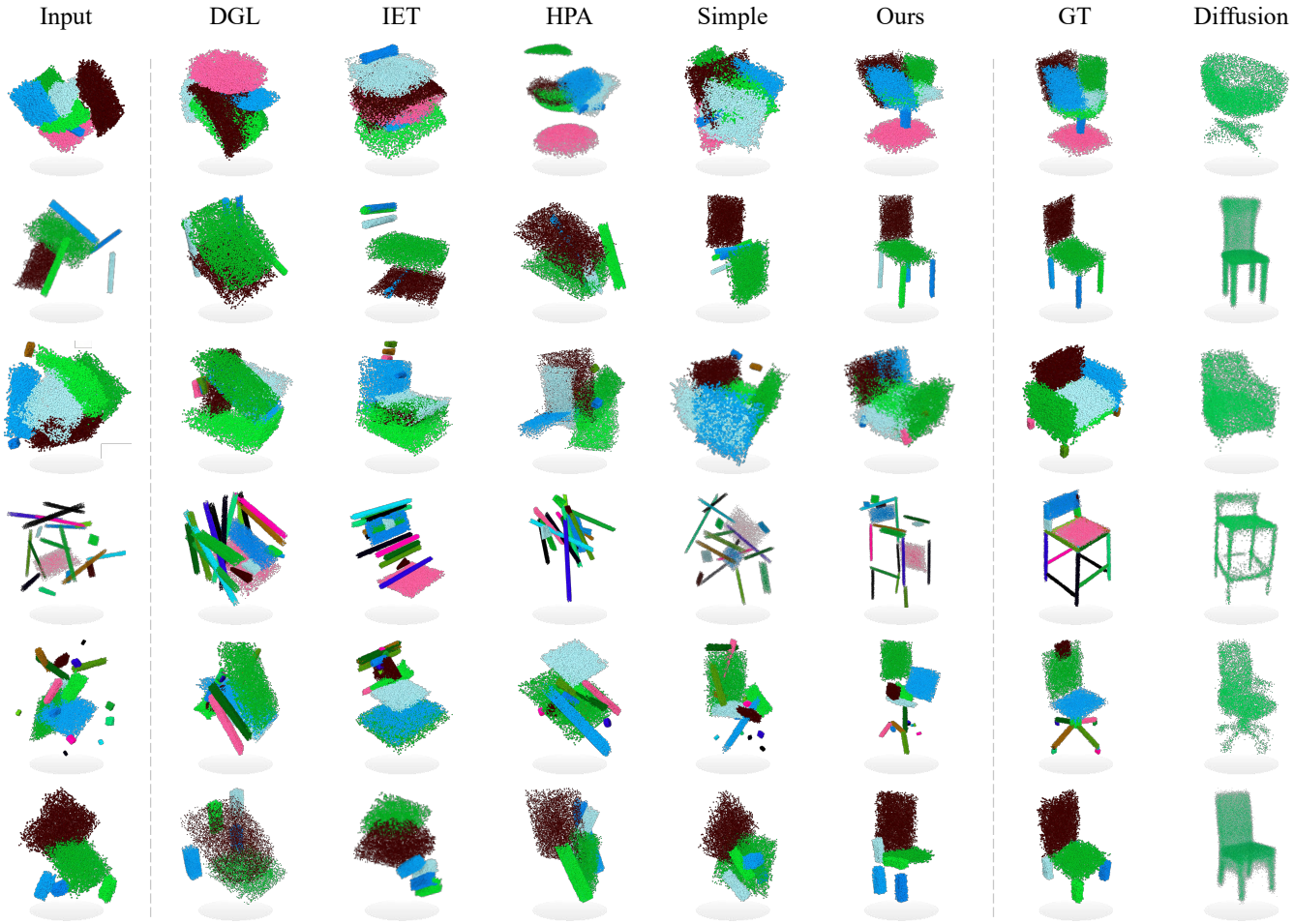

Figure 2: **Visual comparisons demonstrating our superior assembly performance over baseline methods on PartNet.** The first column shows our input at the Excessive level, while the last column presents reference samples obtained through diffusion sampling.

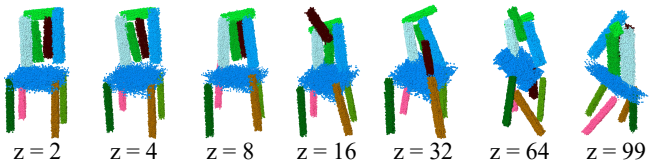

Figure 3: **Different z in our experiments.**

where $\mathbf{A} = \{\mathbf{A_i} \mid i = 1, ..., N\}$, $g(*)$ denotes the matrix multiplication with the unordered point cloud. For previously supervised part assemblers, the optimization of $\mathbf{A}$ is quite straightforward:

$$\arg\min_{A} \mathbb{E}\left[g(\mathbf{A}) - \mathcal{P}_{gt}\right], \tag{8}$$

where $\mathbb{E}$ denotes a set of distance functions and $\mathcal{P}_{gt}$ is the ground truth. However, optimizing $\mathbf{A}$ is non-trivial in our case, where no supervised data is available. Therefore, instead of forcing $g(\mathbf{A})$ to fit a determined object, **we tend to make the generation of $g(\mathbf{A})$ looks like a realistic object, i.e. a sample from the distribution of the real object.** Inspired by Poole et al. [2022], we leverage a pre-trained diffusion model for 3D point cloud generation, which implicitly captures the distribution of point clouds in real-world objects. Then we optimize over $\mathbf{A}$ so that $g(\mathbf{A})$ looks like a sample from this frozen diffusion model. This is achieved through a Score Distillation Sampling (SDS) loss [Poole *et al.*, 2022]:

$$\nabla_{\mathbf{A}}\mathcal{L}_{\text{SDS}}(\theta, g(\mathbf{A})) \triangleq \mathbb{E}_{z,\epsilon}\Bigg[w(z)\left(\epsilon_\theta\left(\mathcal{P}_{t,z}; c, z\right) - \epsilon\right)\frac{\partial \mathcal{P}_{out}}{\partial \mathbf{A}}\Bigg]. \tag{9}$$

As shown in Fig.1:

$$\epsilon_\theta\left(\mathcal{P}_{t,z}; c, z\right) = \mathcal{P}_{t,z} - \mathcal{P}^*, \tag{10}$$

$$\epsilon = \mathcal{P}_{t,z} - \mathcal{P}_t. \tag{11}$$

Substituting Eq. 10 and Eq. 11 into Eq. 9, we get:

$$\nabla_{\mathbf{A}}\mathcal{L}_{\text{SDS}}(\theta, g(\mathbf{A})) \triangleq \mathbb{E}_z\left[w(z)\left(\mathcal{P}_t - \mathcal{P}^*\right)\frac{\partial \mathcal{P}^{\text{out}}}{\partial \mathbf{A}}\right]. \tag{12}$$

In the equation above (Eq. 12), since $g(*)$ represents matrix multiplication, $\frac{\partial \mathcal{P}_{out}}{\partial \mathbf{A}}$ corresponds to the coordinates of

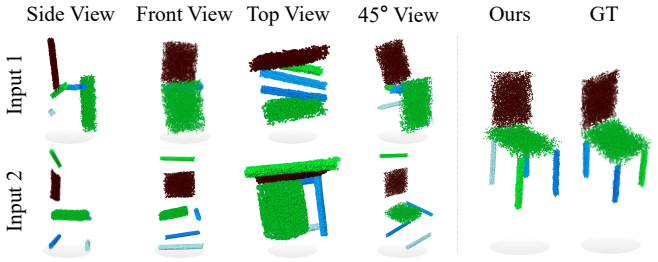

Figure 4: **Different Views from Baseline-Simple and Ours.** Baseline-Simple utilizes supervised learning on point clouds generated by a diffusion model, while our method employs density estimates. The results of the Simple are similar to set of point clouds from reference, but do not correspond to a chair shape.

the points in $\mathcal{P}_t$, which are constant three-dimensional vectors; $w(z)$ is a constant scalar in practice, which will be explained in the following section. Therefore, the remaining term $\mathcal{P}_t - \mathcal{P}^*$ governs the descent process of the $\mathcal{L}_{\text{SDS}}$. If we can accurately estimate the transformation from $\mathcal{P}_t$ to $\mathcal{P}^*$, the optimization process will converge directly.

In practice, we utilize the ICP algorithm to estimate the transformation between $\mathcal{P}_t$ and $\mathcal{P}^*$, as shown in Fig. 1. It is worth noting that the smaller the change in the shape of each part in $\mathcal{P}_t$ and $\mathcal{P}^*$, the more accurate the transformation obtained by the ICP algorithm. We minimize the shape variation between $\mathcal{P}_t$ and $\mathcal{P}^*$ by controlling the magnitude of noise added and removed during the forward and generation processes. Recall that the step size determines the noise (Eq. 1 and 3): (1) In the forward process, smaller step sizes reduce the Gaussian noise variance, bringing $\mathcal{P}_t$ closer to $\mathcal{P}_{t,z}$; (2) In the generation process, smaller step sizes reduce denoising variance, making $\mathcal{P}_{t,z}$ closer to $\mathcal{P}^*$. Therefore, we fixed the time step z to a small value, typically 2 or 4, to obtain more accurate ICP estimates. Fig. 3 shows the assembly result under different z with the same iterations, a smaller z significantly improves the realism of the results. We finally apply the transformations obtained from the ICP algorithm to $\mathcal{P}_t$ to generate the result of this iteration $\mathcal{P}_{t-1}$ and use it as the input for the next iteration. With each iteration, the diffusion model helps bring our results closer to real-world objects.

5 Experiments

5.1 Datasets, Baselines, and Metrics

Dataset. We conduct experiments on the Chair subset of PartNet [2019], a large-scale dataset with fine-grained part-level annotations. We follow the official train/val/test splits: the training set is used to train a diffusion model for 3D point cloud generation, and the test set is used for zero-shot assembly. Part counts range from 2 to 20. To evaluate robustness, we introduce four noise levels: *slight*, *moderate*, *substantial*, and *excessive* (see Fig. 6). The diffusion model is adopted from [Luo and Hu, 2021].

Baselines. We compared our approach with Complement [2017], DGL [2020], IET [2022], HPA [2024], and Simple. Among them, Simple is the Baseline we designed, which utilizes seven trainable parameters to represent the rotational and translational transformations of parts. Simple usually outperforms other baseline methods in practice.

Metrics. We evaluate using Part Accuracy (PA) and Shape Chamfer Distance (SCD) from Zhan et al.[2020], as well as Rotation and Translation RMSE (RMSE(R), RMSE(T)) from Sellán et al.[2022]. PA assesses per-part precision, SCD measures overall shape quality, and RMSE(R/T) quantify rotation and translation accuracy.

Fair Part Accuracy (fPA). Vanilla PA is determined by the Chamfer Distance between components with identical tensor indices. This metric serves as a criterion for evaluating assembly precision. However, certain components, such as stool legs, are permitted to be positioned in regions with inconsistent indices, as illustrated by GT and Ours in Figure. Therefore, we propose the concept of Fair Part Accuracy (fPA). Given two point clouds $\mathcal{P}_{\text{pred}} = \{p[i], i \in \{1, 2, \ldots, N\}\}$ and $\mathcal{P}_{\text{gt}} = \{g[i], i \in \{1, 2, \ldots, N\}\}$, $\mathcal{CD}$ represents Chamfer Distance calculation. we formally define:

$$j^* = \text{argmin}_j \, \mathcal{CD}(\mathcal{P}_{\text{pred}}[i], \mathcal{P}_{\text{gt}}[j]).$$

Next, we use $\mathcal{P}_{\text{fair gt}}$ to replace $\mathcal{P}_{\text{gt}}$.

$$\mathcal{P}_{\text{fair gt}}[i] = \mathcal{P}_{\text{gt}}[j^*], \text{ where } i \in \{1, 2, \ldots, N\}.$$

We define the accuracy as:

$$\text{fPA} = \frac{1}{N} \sum_{p=1}^{N} \mathbb{1} \left(\frac{1}{N} \sum_{p} \mathcal{CD}(\mathcal{P}_{pred}[p], \mathcal{P}_{fair\ gt}[p]) < \text{thre} \right),$$

where thre = 0.01, which is a parameter inherited from previous work [2020]. $\mathbb{1}$ denotes an indicator function that equals 1 if the condition inside is met, and 0 otherwise.

5.2 Experiments Results and Analysis

As demonstrated in the first four rows of Table 1, assembly performance declines with increasing noise intensity, thereby validating our noise level designations. The *slight* noise level evaluates the ability of our method to converge close to the ground truth. Conversely, the *excessive* noise level, characterized by randomly dispersed point clouds, tests the extremes of performance for both baselines and our method. The intermediate *moderate* and *substantial* levels further substantiate the efficacy and rationality of our noise addition strategy.

We evaluated our method against various baselines, as shown in Table 1, Fig. 2, and the Appendix. Our approach outperforms current methods in addressing the zero-shot challenge. All of our metrics outperform existing methods, except for SCD. This is expected since our method emphasizes density estimates from a diffusion model rather than sampling complete shapes. Therefore, our assembled outputs conceptually resemble chairs instead of precisely replicating chair-shaped point clouds. This distinction is illustrated in Fig. 4, which visualizes samples generated by Simple under two different random seeds. Furthermore, compared to IET, Simple performs worse on the SCD, highlighting the advantage of Transformer models in encoding complex structures. Additionally, we tested a carefully designed model, HPA, whose performance is significantly impaired when trained exclusively with the SCD. Without any prior information on part

Methods	Noise Level	SCD ↓ $\times 10^{-3}$	PA ↑ %	RMSE(Trans) ↓ $\times 10^{-2}$	RMSE(Rot) ↓	fPA ↑ %
Ours	Slight	7.7	68.59	7.56	7.18	68.91
Ours	Moderate	17.0	36.53	27.19	27.16	37.43
Ours	Substantial	34.8	15.54	26.89	32.70	17.90
Ours	Excessive	45.0	**9.0**	**28.52**	**31.02**	**12.3**
Simple	Excessive	<u>31.7</u>	<u>2.49</u>	48.13	<u>57.13</u>	<u>6.26</u>
HPA	Excessive	156.2	0.02	<u>42.69</u>	72.41	0.06
IET	Excessive	**12.94**	0.05	56.75	93.12	0.18
DGL	Excessive	165.2	0.03	66.80	69.35	0.08
Random Initial	Excessive	203.4	0.0	61.48	89.97	0.0

Table 1: **Quantitative evaluation on zero-shot scenario.** <u>Underline</u>/**bold** fonts highlight the suboptimal/best approach. Our approach outperforms current methods in addressing the zero-shot challenge.

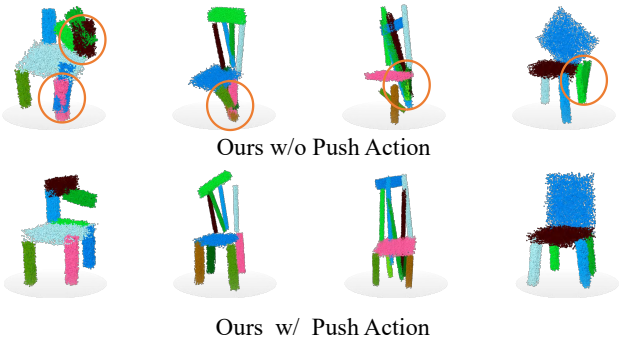

Figure 5: **Ablation Study of Push Action.** Based on our method, we can clearly separate overlapping parts, which helps reduce the overlap problem.

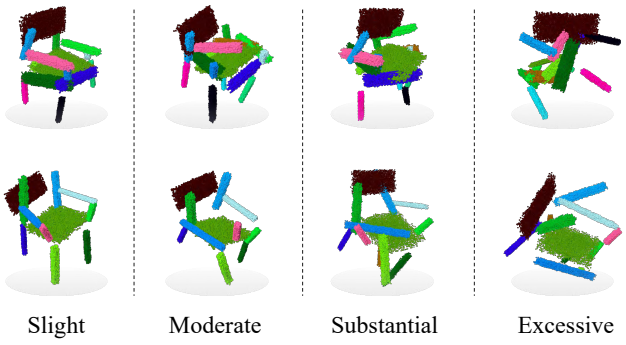

Figure 6: **Different noise levels in our experiments.** Illustrations of input under various noise conditions, including slight, moderate, substantial, and excessive noise.

poses, all baselines demonstrate notably poor performance, as further analysis of their training loss functions reveals why they fail in zero-shot scenarios (details in the Appendix). In the Appendix, we also present an experiment designed to illustrate both the effectiveness of the proposed method and its limitations on challenging samples. To evaluate the proposed method under varying levels of task complexity, we conducted experiments with different numbers of components in the Appendix. As well as more visual results and the details of our experiments.

5.3 Comparisons with supervised scenario

As indicated in Table 2, our work achieves comparable results by the early supervised learning method: Complement. This finding underscores that our method can deliver competitive outcomes even compared to pose-accessible supervised learning. While our work may not yet achieve the performance of existing well-designed supervised learning methods, we hope it offers insights that may contribute to future research in zero-shot learning.

Scenario	Methods	SCD ↓	PA ↑	fPA ↑
Zero-Shot	Ours	45.0	<u>9.0</u>	12.3
Zero-Shot	Simple	31.7	2.49	6.26
Supervised	Complement	<u>24.1</u>	8.78	-
Supervised	DGL	**9.1**	**39.0**	-

Table 2: **Comparisons with methods on Supervised scenario.** PA is a metric used to evaluate the accuracy of each part. Our zero-shot method can surpass the Complement with supervised learning.

5.4 Ablation Study

In this work, we propose an improved approach to address the collision issue that arises when identical parts are placed in the same position. To mitigate this issue, we introduce an explicit pushing-apart operation. Traditional model-based training methods struggle to achieve this directly, as the model generates the predicted poses of parts, and adjustments can only be made by tuning the model parameters, which limits its operational flexibility. However, in this study, we innovatively incorporate an explicit pushing-apart operation into the

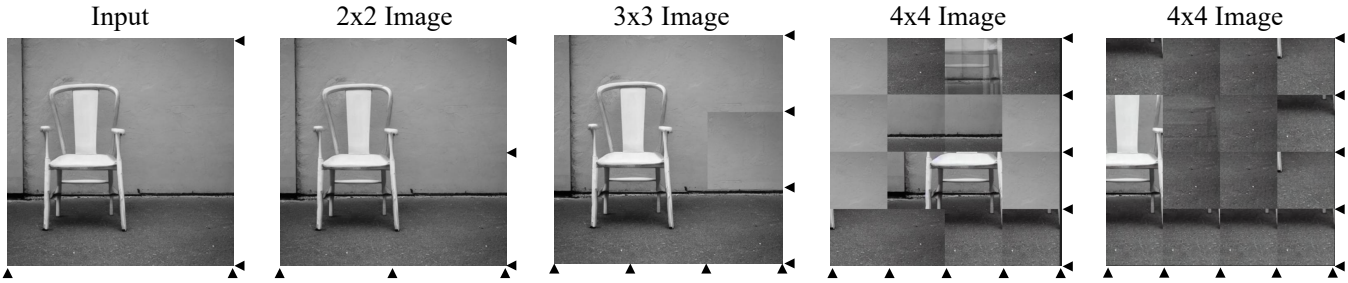

Figure 7: **Visual results of 2D Image Reassembly.** This figure showcases the effectiveness of the 2D diffusion model in reassembling fragmented images without simplifying the problem, achieving near-perfect reconstruction for 3×3 image puzzles.

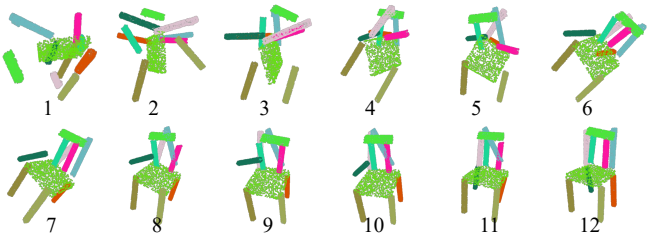

Figure 8: **The process of step-by-step assembly with 2D Stable Diffusion.** By simplifying the experiments' complexity, the 3D part assembly can also be achieved using 2D stable diffusion.

original method. This operation effectively separates parts, thereby reducing collision issues. The experimental results, as shown in Fig. 5, demonstrate the effectiveness of the proposed method.

5.5 Tranfer our work to 2D Diffusion Model

As analyzed in Section 4, our method is not limited to specific types of diffusion models. A pretrained 2D image diffusion model, in theory, can also evaluate the quality of assembly results by differentiably rendering 3D components into 2D space and feeding them into the model. However, in practice, this approach is challenging due to the difficulty of propagating 2D signals back to 3D point clouds and pose parameters. To test this potential, we conducted a simplified experiment where all parts shared a common rotational perturbation. This perturbation was optimized using Eq. 9. We utilized stable diffusion 2.1 [Rombach *et al.*, 2022] with the prompt "a picture of colorful chair" to generate a realistic chair shape. As shown in Fig. 8, we demonstrate the visualization of the assembly process. These images are processed through microrendering, serving as inputs for the diffusion model to obtain SDS loss. This experiment demonstrates that 2D diffusion models have inherent assembly potential, though sophisticated methods are required to fully utilize it. More details in the Appendix.

5.6 Transfer our work to 2D Image Reassembly

In addition to exploring the potential of 2D diffusion models for 3D part assembly tasks, we also examined their capability in 2D image reassembly. Our model can almost perfectly reconstruct 3×3 image fragments without oversimplifying the challenge. 2D Image Reassembly involves reassembling cropped segments of a 2D image [Scarpellini *et al.*, 2024]. In this experiment, no special architecture was designed; instead, we implemented a classifier utilizing two CNN layers. An MLP was implemented to predict the correct position of each sub-image within the whole picture. Training followed the method outlined in Eq. 9, using stable diffusion 2.1 [Rombach *et al.*, 2022] with the prompt "a picture of chair". As illustrated in Fig. 7, our approach successfully handles 2D image reassembly challenges up to a complexity of 4×4. More details in the Appendix.

6 Conclusion

In this work, we introduced a novel zero-shot assembly method that leverages the inherent assembly capabilities of general-purpose diffusion models to generate continuous rigid transformations for object assembly without prior training on specific shapes or configurations. It uncovers the implicit assembly abilities of general models, enabling assembly tasks even with previously unseen data. Our work is the first zero-shot part assembly framework, which aims to harness existing extensive work on diffusion models to achieve assembly at virtually no additional cost, which represents a meaningful and valuable contribution.

7 Limitation and Future Work

The bottom row of Fig. 2 exemplifies a failure case, revealing challenges in accurately placing overlapping parts. Especially when they are far from their GT positions, using the push operation cannot accurately place overlapping parts. Adjusting random inputs can mitigate this issue, but it is not robust enough. In the future, we plan to investigate an interpretable approach to reposition misplaced parts. This requires us to better explore the underutilized assembly knowledge inherent in general models and to be able to identify which positions have vacancies, thus allowing for the effective transfer of parts. Our goal is to improve the model's performance in assembly tasks without extensive additional costs or extensive supervised learning.

Acknowledgments

This work was supported by the Zhejiang Provincial Key Research and Development Project (2023C01043), Engineering Research Center of Integration and Application of Digital Learning Technology, Ministry of Education, and

the Academy of Social Governance, Zhejiang University. Ruiyuan Zhang and Qi Wang contributed equally to this work. Please ask Prof. Wu (chao.wu@zju.edu.cn) for correspondence.

References

[Baranchuk *et al.*, 2021] Dmitry Baranchuk, Ivan Rubachev, Andrey Voynov, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. *arXiv preprint arXiv:2112.03126*, 2021.

[Cheng *et al.*, 2023] Junfeng Cheng, Mingdong Wu, Ruiyuan Zhang, Guanqi Zhan, Chao Wu, and Hao Dong. Score-pa: Score-based 3d part assembly. *arXiv preprint arXiv:2309.04220*, 2023.

[Chervinskii *et al.*, 2023] Fedor Chervinskii, Sergei Zobov, Aleksandr Rybnikov, Danil Petrov, and Komal Vendidandi. Auto-assembly: a framework for automated robotic assembly directly from cad. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11294–11300. IEEE, 2023.

[Gao *et al.*, 2024] Xiang Gao, Wei Hu, Renjie Liao, et al. Generative 3d part assembly via part-whole-hierarchy message passing. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20850–20859. IEEE, 2024.

[Ghasemipour *et al.*, 2022] Seyed Kamyar Seyed Ghasemipour, Satoshi Kataoka, Byron David, Daniel Freeman, Shixiang Shane Gu, and Igor Mordatch. Blocks assemble! learning to assemble with large-scale structured reinforcement learning. In *International Conference on Machine Learning*, pages 7435–7469. PMLR, 2022.

[Guo *et al.*, 2025] Wenxiang Guo, Yu Zhang, Changhao Pan, Rongjie Huang, Li Tang, Ruiqi Li, Zhiqing Hong, Yongqi Wang, and Zhou Zhao. Techsinger: Technique controllable multilingual singing voice synthesis via flow matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 23978–23986, 2025.

[Ho *et al.*, 2020] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[Hu *et al.*, 2024] Zhipeng Hu, Minda Zhao, Chaoyi Zhao, Xinyue Liang, Lincheng Li, Zeng Zhao, Changjie Fan, Xiaowei Zhou, and Xin Yu. Efficientdreamer: High-fidelity and robust 3d creation via orthogonal-view diffusion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4949–4958, 2024.

[Jaiswal *et al.*, 2016] Prakhar Jaiswal, Jinmiao Huang, and Rahul Rai. Assembly-based conceptual 3d modeling with unlabeled components using probabilistic factor graph. *Computer-Aided Design*, 74:45–54, 2016.

[Karazija *et al.*, 2023] Laurynas Karazija, Iro Laina, Andrea Vedaldi, and Christian Rupprecht. Diffusion models for zero-shot open-vocabulary segmentation. *arXiv preprint arXiv:2306.09316*, 2023.

[Li *et al.*, 2023] Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2206–2217, 2023.

[Lin *et al.*, 2023] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023.

[Liu *et al.*, 2023a] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023.

[Liu *et al.*, 2023b] Jiaxiang Liu, Tianxiang Hu, Yan Zhang, Xiaotang Gai, Yang Feng, and Zuozhu Liu. A chatgpt aided explainable framework for zero-shot medical image diagnosis. *arXiv preprint arXiv:2307.01981*, 2023.

[Liu *et al.*, 2024] Jiaxiang Liu, Tianxiang Hu, Huimin Xiong, Jiawei Du, Yang Feng, Jian Wu, Joey Zhou, and Zuozhu Liu. Vpl: Visual proxy learning framework for zero-shot medical image diagnosis. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9978–9992, 2024.

[Liu *et al.*, 2025] Jiaxiang Liu, Tianxiang Hu, Jiawei Du, Ruiyuan Zhang, Joey Tianyi Zhou, and Zuozhu Liu. Kpl: Training-free medical knowledge mining of vision-language models. *arXiv preprint arXiv:2501.11231*, 2025.

[Luo and Hu, 2021] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2837–2845, 2021.

[Mo *et al.*, 2019] Kaichun Mo, Shilin Zhu, Angel X Chang, Li Yi, Subarna Tripathi, Leonidas J Guibas, and Hao Su. Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 909–918, 2019.

[Narayan *et al.*, 2022] 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.

[Peebles and Xie, 2023] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[Poole *et al.*, 2022] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[Rombach *et al.*, 2022] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent

diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022.

[Saharia *et al.*, 2022] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

[Scarpellini *et al.*, 2024] Gianluca Scarpellini, Stefano Fiorini, et al. Diffassemble: A unified graph-diffusion model for 2d and 3d reassembly. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28098–28108, 2024.

[Sellán *et al.*, 2022] Silvia Sellán, Yun-Chun Chen, Ziyi Wu, Animesh Garg, and Alec Jacobson. Breaking bad: A dataset for geometric fracture and reassembly. *Advances in Neural Information Processing Systems*, 35:38885–38898, 2022.

[Song and Ermon, 2019] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

[Sung *et al.*, 2017] Minhyuk Sung, Hao Su, Vladimir G Kim, Siddhartha Chaudhuri, and Leonidas Guibas. Complementme: Weakly-supervised component suggestions for 3d modeling. *ACM Transactions on Graphics (TOG)*, 36(6):1–12, 2017.

[Vaswani, 2017] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[Wang *et al.*, 2024a] Xiang Wang, Shiwei Zhang, Hangjie Yuan, Zhiwu Qing, Biao Gong, Yingya Zhang, Yujun Shen, Changxin Gao, and Nong Sang. A recipe for scaling up text-to-video generation with text-free videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6572–6582, 2024.

[Wang *et al.*, 2024b] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems*, 36, 2024.

[Xu *et al.*, 2024] Qun-Ce Xu, Hao-Xiang Chen, Jiacheng Hua, Xiaohua Zhan, Yong-Liang Yang, and Tai-Jiang Mu. Fragmentdiff: A diffusion model for fractured object assembly. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–12, 2024.

[Yoon *et al.*, 2003] Youngrock Yoon, Guilherme N DeSouza, and Avinash C Kak. Real-time tracking and pose estimation for industrial objects using geometric features. In *2003 IEEE International conference on robotics and automation (cat. no. 03CH37422)*, volume 3, pages 3473–3478. IEEE, 2003.

[Zhan *et al.*, 2020] Guanqi Zhan, Qingnan Fan, and others Mo. Generative 3d part assembly via dynamic graph learning. *Advances in Neural Information Processing Systems*, 33:6315–6326, 2020.

[Zhang *et al.*, 2022] Rufeng Zhang, Tao Kong, Weihao Wang, Xuan Han, and Mingyu You. 3d part assembly generation with instance encoded transformer. *IEEE Robotics and Automation Letters*, 7(4):9051–9058, 2022.

[Zhang *et al.*, 2024a] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2024.

[Zhang *et al.*, 2024b] Ruiyuan Zhang, Jiaxiang Liu, Zexi Li, Hao Dong, Jie Fu, and Chao Wu. Scalable geometric fracture assembly via co-creation space among assemblers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 7269–7277, 2024.

[Zhang *et al.*, 2024c] Yu Zhang, Rongjie Huang, Ruiqi Li, JinZheng He, Yan Xia, Feiyang Chen, Xinyu Duan, Baoxing Huai, and Zhou Zhao. Stylesinger: Style transfer for out-of-domain singing voice synthesis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19597–19605, 2024.

[Zhang *et al.*, 2024d] Yu Zhang, Ziyue Jiang, Ruiqi Li, Changhao Pan, Jinzheng He, Rongjie Huang, Chuxin Wang, and Zhou Zhao. Tcsinger: Zero-shot singing voice synthesis with style transfer and multi-level style control. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1960–1975, 2024.

[Zhang *et al.*, 2024e] Yu Zhang, Changhao Pan, Wenxiang Guo, Ruiqi Li, Zhiyuan Zhu, Jialei Wang, Wenhao Xu, Jingyu Lu, Zhiqing Hong, Chuxin Wang, et al. Gtsinger: A global multi-technique singing corpus with realistic music scores for all singing tasks. *arXiv preprint arXiv:2409.13832*, 2024.

[Zhang *et al.*, 2025] Ruiyuan Zhang, Yuyao Chen, Jiaxiang Liu, Dianbing Xi, Yuchi Huo, Jie Liu, and Chao Wu. Sgw-based multi-task learning in vision tasks. In *Asian Conference on Computer Vision*, pages 124–141. Springer, 2025.