

From One to More: Contextual Part Latents for 3D Generation

Shaocong Dong^{1*}, Lihe Ding^{2*}, Xiao Chen², Yaokun Li², Yuxin Wang¹,
Yucheng Wang¹, Qi Wang¹, Jaehyeok Kim¹, Chenjian Gao², Zhanpeng Huang³,
Zibin Wang³, Tianfan Xue^{2,4†}, Dan Xu^{1†}

¹HKUST ²CUHK ³SenseTime Research ⁴Shanghai AI Laboratory

{sdongae, danxu}@cse.ust.hk, {dl023, tfxue}@ie.cuhk.edu.hk



Figure 1. *CoPart* achieves high-quality part-based 3D generation and supports various applications.

Abstract

To generate 3D objects, early research focused on multi-view-driven approaches relying solely on 2D renderings. Recently, the 3D native latent diffusion paradigm has demonstrated superior performance in 3D generation, because it fully leverages the geometric information provided in ground truth 3D data. Despite its fast development, 3D diffusion still faces three challenges. First, the majority of these methods represent a 3D object by one single latent, regardless of its complexity. This may lead to detail loss when generating 3D objects with multiple complicated parts. Second, most 3D assets are designed parts by parts, yet the current holistic latent representation overlooks the independence of these parts and their interrelationships, limiting the model’s generative ability. Third, current methods rely on global conditions (e.g., text, image, point cloud) to control the generation process, lacking detailed controllability. Therefore, motivated by how 3D designers create a 3D object, we present a new part-based 3D generation framework, *CoPart*, which represents a 3D object with multiple

contextual part latents and simultaneously generates coherent 3D parts. This part-based framework has several advantages, including: i) reduces the encoding burden of intricate objects by decomposing them into simpler parts, ii) facilitates part learning and part relationship modeling, and iii) naturally supports part-level control. Furthermore, to ensure the coherence of part latents and to harness the powerful priors from foundation models, we propose a novel mutual guidance strategy to fine-tune pre-trained diffusion models for joint part latent denoising. Benefiting from the part-based representation, we demonstrate that *CoPart* can support various applications including part-editing, articulated object generation, and mini-scene generation. Moreover, we collect a new large-scale 3D part dataset named *Partverse* from *Objaverse* through automatic mesh segmentation and subsequent human post-annotations. By training on the proposed dataset, *CoPart* achieves promising part-based 3D generation with high controllability. Project page: <https://copart3d.github.io>.

1. Introduction

With the emergence of large-scale 3D datasets [7], many techniques have been proposed to convert the raw 3D data into different representations for generative modeling. Pioneering multi-view driven works [30, 41] render 3D mesh into multi-view images and train multi-view diffusion models or large reconstruction models with only 2D image supervision. These methods can generate consistent multi-view images of 3D objects but are poor at recovering high-quality geometry since the accurate 3D shape supervision is omitted when converting mesh into multi-view images. Another 3D latent diffusion method CLAY [56] converts 3D meshes into latent tokens by a 3D VAE [19, 54] and trains a latent diffusion model. This method implicitly preserves previously overlooked geometric supervision through 3D occupancy auto-encoding, resulting in improved generation quality.

Despite all these advances of 3D latent diffusion, it still faces three challenges, making it still sub-optimal for 3D generation. First, the current methods treat intricate 3D meshes and simple ones equally, using the same number of tokens. However, the constrained representative ability of 3D VAE will inevitably cause information loss for complex data; and the unbalanced data distribution will make simpler geometries dominate the training process. Second, most 3D designers create complex 3D objects part by part, so they can spend more time adding detailed geometries for each part. On the contrary, state-of-the-art 3D generation algorithms neither utilize the part representations nor explicitly model the relationships between parts, limiting their ability to generate detailed and independent parts. For instance, when generating “a person with a hat,” most algorithms will fuse the head and hat together as a single object within a limited resolution of each local region, leading to low quality. However, users need two distinguishable detailed parts, especially for the face. Third, current methods utilize global conditions (e.g., text, image, point clouds) to control the generation process, which lacks detailed local controllability.

Based on these observations, we demonstrate that all these issues can be addressed through part-based 3D object representation and generation, as part-based modeling can i) naturally distribute complexity across individual parts, ii) efficiently learn part-level information from the data, and iii) provide detailed control at the part level. Therefore, we propose a novel part-based 3D generation framework, *CoPart*, which represents a 3D object with multiple **Contextual Part** latents and generates part latents by learning a joint distribution across diverse 3D data.

While 3D part generation has been explored previously, our solution offers a very distinct perspective. Majority of 3D part generation models either i) are restricted by PartNet [32] categories with limited generalizability [10, 21,

33], or ii) adopt a “top-down” strategy [27], segmenting input images into part patches for individual reconstruction. The latter limits the model’s ability to leverage part information during training and depends heavily on segmentation quality. In contrast, *CoPart* adopts a “bottom-up” framework that directly generates coherent parts and leverage the diverse Objaverse [7] dataset, ensuring greater generalizability.

Specifically, *CoPart* encodes each 3D part using a geometric token and an image token extracted from the part image, instead of one single global latent. This approach is beneficial for two reasons. First, each part has the complementary geometry and image tokens. Geometry tokens model the detailed shape, while the image tokens not only provide appearance information but also offer semantic cues for understanding part relationships. Second, decoupling geometry from image latents allows us to leverage the capabilities of pre-trained 3D and 2D autoencoders more effectively. For geometric tokens, since the part geometry is normally simpler than object geometry, they can be more efficiently encoded by a 3D part VAE [19]. For image tokens, since each part can be rendered in much higher resolution, 2D diffusion model can generate more detailed textures.

To learn the distribution of both geometry and image tokens of each part, we finetune the diffusion models for 3D geometries [23] and 2D images [4], leveraging their pre-trained priors for better generation quality. To ensure both consistency between different parts and between geometry tokens and image tokens, we introduce a mutual guidance diffusion model, inspired by [8]. The mutual guidance facilitates information exchange between different parts as well as between each part’s geometric and image tokens, achieving both part consistency and geometry-image consistency. Furthermore, to eliminate the ambiguity of part order and effectively control the part generation using the input 3D bounding boxes, we propose a novel strategy to encode bounding box conditions to guide part generation. In this way, with the input of bounding boxes and text descriptions, we can generate high-quality 3D objects by decoding and assembling part latents, as shown in the first row of Fig. 1.

Collecting high-quality 3D part data for training is also non-trivial. One option is the PartNet dataset [32], but it only contains 24 categories of objects and has poor textures, restricting model generalizability. Another option is Objaverse [7], which offers more diversity but suffers from inconsistent part labels as different 3D designers prefer different ways to partition an object, often leading to over- or under-segmentation. To address this, we first employ a mesh segmentation pipeline to automatically decompose objects into reasonable parts. Then we manually conduct simple post-processing, including filtering low-quality data and grouping the over-segmented parts. Additionally, we

utilize a multi-modal vision-language model [24] to generate text prompts for each part. In this way, we obtain a high-quality 3D part dataset with 91k parts for 12k objects.

With the proposed *CoPart* model trained on our 3D part dataset, it unlocks many various new applications, as shown in the bottom of Fig. 1. First, we can run structure diffusion [28] to obtain bounding boxes and articulation information while using *CoPart* to generate parts, achieving novel articulated objects generation. Second, we can generate a mini-scene by considering each object in a scene as a part. Thirdly, we can achieve part-based 3D editing by resampling selected part latents. Experimental results also show that our method can generate high-quality 3D objects with more accurate parts compared with the previous works, and also support various applications as discussed above.

2. Related Work

2.1. 3D Generation

In contrast to earlier category-specific 3D generation methods [2, 6, 9, 13, 17, 34, 42, 46, 51], contemporary 3D generative models are capable of producing diverse 3D objects conditioned on text or images. DreamFusion [36] and its subsequent works [26, 37, 45] introduce the Score Distillation Sampling (SDS) loss to adapt 2D diffusion models for 3D generation. Multi-view diffusion approaches [29, 30, 41] fine-tune 2D diffusion models to generate multi-view consistent images. Meanwhile, LRM [15] trains a large-scale reconstruction model to predict 3D radiance fields from a single image. More recently, CLAY [56] and its follow-ups [23, 53] directly train 3D-native diffusion models, achieving significantly improved performance.

2.2. Part Generation

StructureNet [31] uses a graph network to understand the relationship between parts while Grass [22] adopting recursive autoencoders for shape structures. DSG-Net [52] proposes disentangled structure and geometry for 3d generation. Other methods [11, 12] employ 3D Gaussian mixture to represent parts. SPAGHETTI [14] and Neural Template [16] train an auto-encoder to map 3D objects into a part-aware latent space, enabling part-aware editing. SALAD [21] replaces the auto-encoder in SPAGHETTI with a diffusion model, achieving superior performance. DiffFacto [33] learns a controllable part-based point cloud diffusion model. PartNeRF [44] and NeuForm [25] also introduce part-based neural representations. While effective, these methods are limited by their reliance on category-specific part data, which restricts their generalizability. Part123 [27] leverages the powerful SAM [20] model to segment multi-view images and perform part-aware reconstruction. Concurrent to our work, PartGen [5], also adopts a "top-down" strategy by first segmenting parts from multi-view images and then performing part completion and re-

construction. However, this approach not only heavily depends on segmentation quality but also struggles with the limited information provided by small segmented patches, which constrains the quality of part reconstruction. In contrast, we propose a "bottom-up" strategy that directly learns the part distribution from diverse data and jointly generates coherent parts.

3. Synchronized Part Latent Diffusion

An effective part-based 3D generative model should be able to generate consistent parts with both high-quality geometry and appearance. However, this is non-trivial for the following three reasons. First, consistency between different parts is hard to ensure. Second, it is not easy to efficiently leverage limited part data to achieve high-quality part-based 3D generation. Third, simultaneously generating parts introduces ambiguity in part ordering.

In this paper, we provide a synchronized part latent diffusion framework to address the above challenges as shown in Fig. 2. In Sec. 3.1, we first introduce our method to represent 3D objects using part latents. Next, in Sec. 3.2.1, we propose an effective framework to synchronize part latent diffusion through mutual guidance, and fine-tune the part latent diffusion model from pre-trained foundation models for efficient part data utilization. Finally, in Sec. 3.2.3, we discuss our approach to inject conditions to resolve the part order ambiguity and enhance controllability.

3.1. Part Representation Encoding

To model the distribution of 3D parts, we decompose a 3D mesh \mathcal{M} into part-based 3D representations that preserve both geometric and appearance information from the ground truth data. Our approach utilizes hybrid part latents to represent 3D texture parts through the combination of geometric tokens (encoded by a 3D part VAE) and image tokens (encoded by an image VAE), as detailed below.

Part Geometric Token Encoding. Given a 3D part geometry \mathcal{M}_p segmented from \mathcal{M} , we sample points $\mathbf{P} \in \mathbb{R}^{S \times 3}$ and their corresponding normals $\mathbf{Q} \in \mathbb{R}^{S \times 3}$ on the part mesh's surface, where $S = 4096$ is the number of sampled points. Then a 3D part VAE encoder \mathcal{E}_{3D} is used to extract 3D part geometric tokens $\mathbf{L}_{3D} = \mathcal{E}_{3D}(\mathbf{P}, \mathbf{Q}) \in \mathbb{R}^{T \times D}$, where T and D represent the token length and dimensions respectively. To enhance the part-level representation learning, we fine-tune the part VAE from a pre-trained holistic 3D VAE [23] using our part data. Additionally, we modify the VAE decoder \mathcal{D}_{3D} from [23] to predict Flexicube [40] parameters, enabling differentiable rendering. More implementation details can be found in the supplementary material. These designs allow us to incorporate normal and depth rendering losses to supervise the VAE fine-tuning.

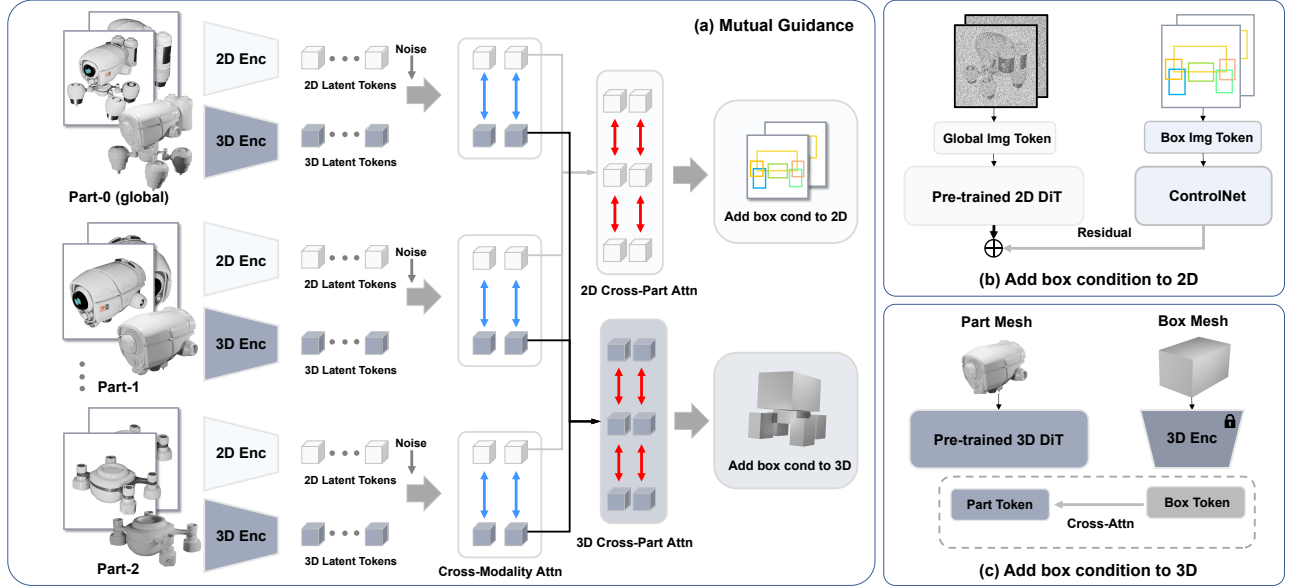


Figure 2. The framework of *CoPart* operates as follows: Gaussian noise is added to part image and geometric tokens extracted from the VAE, which are then fed into 3D and 2D denoisers. Mutual guidance (a) is introduced to facilitate information exchange between the 3D and 2D modalities (via Cross-Modality Attention) as well as between different parts (via Cross-Part Attention). Additionally, (b) the 3D bounding boxes are treated as cube meshes, and the extracted box tokens are injected into the 3D denoiser through cross-attention. Simultaneously, the boxes are rendered into 2D images and injected into the 2D denoiser via ControlNet.

Part Image Token Encoding. To encode part appearance, we render the part mesh \mathcal{M}_p into multi-view part-centric images $\{\mathcal{O}_k\}_{k=1}^v$. Using a pre-trained image VAE \mathcal{E}_{2D} [4], we obtain part image tokens: $\mathbf{L}_{2D} = \{\mathbf{F}_k | \mathbf{F}_k = \mathcal{E}_{2D}(\mathcal{O}_k) \in \mathbb{R}^{T \times D}\}_{k=1}^v$, where v denotes the number of views, and T and D represent the token length and dimensions for each view respectively.

3.2. Synchronized Diffusion

As introduced in Sec. 3.1, a 3D object can be represented by N part latents, comprising geometric tokens and image tokens: $\{\mathbf{L}_{3D}^p, \mathbf{L}_{2D}^p\}_{p=1}^N$. To enable effective part generation, we leverage the powerful priors from pre-trained geometric and image diffusion models, and further fine-tune them with our part data. Specifically, we fine-tune a pre-trained 3D latent diffusion [23] to generate geometric part latents \mathbf{L}_{3D}^p , while fine-tuning pre-trained image diffusion models [4] to generate image part latents \mathbf{L}_{2D}^p .

To ensure consistent 3D object generation across all diffusion processes, we apply two types of synchronization. First is the inter-part synchronization, which ensures the part consistency. It synchronizes between parts \mathbf{L}_b^i and \mathbf{L}_b^j , where $b \in \{3D, 2D\}$ denotes the modality and $i \neq j$ indicates different parts. Second is the intra-part synchronization between geometric and appearance representations \mathbf{L}_{3D}^i and \mathbf{L}_{2D}^i within the same i -th part, ensuring cross-modality (geometry and image modality) consistency. Details of both synchronization through guidance are described below.

3.2.1. Mutual Guidance

The inter-part synchronization design is inspired by BiDiff [8], which synchronizes a 3D diffusion model and a 2D diffusion model through bidirectional guidance. We further extend this approach by adding mutual guidance between different part latents as well as between 3D and 2D modalities. The original bidirectional guidance, which relies on 2D-to-3D lifting and 3D-to-2D rendering, proves memory-intensive and inefficient for exchanging part-level information. Instead, we adopt a more effective implicit mutual guidance strategy that uses attention to exchange information between different parts and modalities (Fig. 2) (a), as we use Transformer-based diffusion models [35]. Specifically, given noisy part latents $\mathbf{L}_{3D}^{p,t}$ and $\mathbf{L}_{2D}^{p,t} = \{\mathbf{F}_k^{p,t}\}_{k=1}^v, p = 1, \dots, N$ at diffusion timestep t , we define the intermediate features from the 3D and 2D diffusion as:

$$\begin{aligned} \mathcal{G}^p &= \text{DiT}(\mathbf{L}_{3D}^{p,t}, y, t), p = 1, \dots, N \\ \mathcal{I}^p &= \text{DiT}(\mathbf{L}_{2D}^{p,t}, y, t), p = 1, \dots, N, \end{aligned} \quad (1)$$

where $\mathcal{G}^p \in \mathbb{R}^{T' \times D'}$ denotes the intermediate 3D features of p -th part, $\mathcal{I}^p = \{\mathcal{F}_k^p \in \mathbb{R}^{T' \times D'}\}_{k=1}^v$ is the intermediate 2D features of p -th part containing v views, and y represents additional conditions such as bounding boxes.

To ensure the inter-part consistency, we extend selected self-attention blocks in each modality to attend tokens from all parts:

$$\begin{aligned} \mathcal{G}^{p'} &= \text{Attention}(\mathcal{G}^p, \{\mathcal{G}^i\}_{i=1}^N) \\ \mathcal{F}_k^{p'} &= \text{Attention}(\mathcal{F}_k^p, \{\mathcal{F}_k^i\}_{i=1}^N), \end{aligned} \quad (2)$$

where Attention(query, key/value) is a standard attention block. This mechanism enables each part to be guided by other parts, facilitating inter-part mutual guidance and synchronization.

Similarly, to ensure cross-modality consistency, we add new attention blocks to exchange information between 3D and 2D features:

$$\begin{aligned}\mathcal{G}^{p'} &= \mathcal{G}^p + \text{LN}(\text{Attention}(\mathcal{G}^p, \{\mathcal{F}_k^p\}_{k=1}^v)) \\ \mathcal{F}_k^{p'} &= \mathcal{F}_k^p + \text{LN}(\text{Attention}(\mathcal{F}_k^p, \mathcal{G}^p)),\end{aligned}\quad (3)$$

where LN() is a linear layer initialized by zeros for training stability. Furthermore, to guarantee multi-view consistency for 2D branch, we also extend some self-attention blocks in 2D diffusion to attend tokens from other views of the same part:

$$\mathcal{F}_k^{p'} = \text{Attention}(\mathcal{F}_k^p, \{\mathcal{F}_k^p\}_{k=1}^v). \quad (4)$$

3.2.2. Global Guidance

To further enhance inter-part consistency, we include a global branch for both 3D and 2D latent diffusions to jointly denoise holistic latents. This global branch functions as an additional ‘‘global part’’ that interacts with other part branches as mentioned in Sec. 3.2.1. In particular, the global branch shares parameters with the part branch, distinguished only by concatenating learnable part-global embeddings to the text embeddings. This architecture enables the network to differentiate between part and global branches, and it further regularizes the fine-tuning process by maintaining global supervision, thereby preventing significant deviation from the original pre-trained weights.

3.2.3. Part Guidance Encoding

One challenge of this design is the part order ambiguity. While we pre-define the part order during training, the network can alter this order during inference, creating a discrepancy between training and inference phases. To alleviate the part order ambiguity and enhance controllability, we introduce two conditions to both the 3D and 2D diffusion denoising processing.

First, we incorporate part-level text prompts to guide the network to distinguish different parts, thereby boosting local semantic controllability. Second, we introduce 3D bounding box conditions for each part as an additional constraint. A naive approach is to use MLPs (multi-layer perceptrons) to encode the coordinates of 3D bounding boxes and use the concatenation of coordinate embeddings with timestep embeddings as conditions. However, learning the correlation between embedded coordinates and actual 3D parts location is not easy.

To solve this challenge in 3D bounding box encoding, we propose a novel strategy to encode 3D bounding boxes by treating each box as a mesh with six surfaces, and then extracting box geometric latents through the pre-trained 3D

mesh VAE encoder, as illustrated in Fig. 2 (c). The encoding can be written as

$$\mathbf{L}_{box}^p = \mathcal{E}_{3D}(\mathbf{P}_{box}^p, \mathbf{Q}_{box}^p), p = 1, \dots, N, \quad (5)$$

where \mathbf{P}_{box}^p and \mathbf{Q}_{box}^p present the sampled points and normals on the p -th bounding box surfaces. In this way, we can encode the 3D bounding box information to the same latent space of the original geometric latents \mathcal{G}^p , and simply added to the geometry latent through an additional cross-attention block:

$$\mathcal{G}^{p'} = \mathcal{G}^p + \text{LN}(\text{Attention}(\mathcal{G}^p, \mathbf{L}_{box}^p)), \quad (6)$$

where LN() denotes a linear layer initialized to zeros.

For image tokens, we implement a dual-pathway approach to incorporate bounding boxes. First, we implicitly query bounding box information from 3D geometric tokens into image tokens through the cross-modal attention mechanism described in Eq. 2. Second, we encode 3D bounding boxes into the image latent space to guide the denoising of the global image branch. Specifically, we render 3D boxes into 2D to generate multi-view bounding box wireframe images. These wireframe images are then encoded into latent tokens by an image VAE and integrated into the 2D denoiser through a lightweight ControlNet [55], as illustrated in Fig. 2 (b). It is noteworthy that these bounding box 2D features are exclusively added to the global branch of the 2D diffusion model.

3.3. Refinement

After fine-tuning on the part dataset, the synchronized part latent diffusion is capable of understanding 3D parts and jointly generating consistent part geometric tokens and image tokens. These tokens can be decoded into part meshes and multi-view part-centric images by VAE decoders \mathcal{D}_{3D} and \mathcal{D}_{2D} :

$$\mathcal{M}^p = \mathcal{D}_{3D}(\mathbf{I}_{3D}^{p,0}), \mathcal{O}^p = \mathcal{D}_{2D}(\mathbf{I}_{2D}^{p,0}). \quad (7)$$

To further improve quality, we leverage the 3D foundation model [48] as an additional enhancer, utilizing both the part images and geometry generated by our model. While the original pipeline of [48] takes a single image and a generated voxel as input, we found it incapable of understanding diverse 3D parts. Therefore, we modify this approach by replacing the voxels with our detailed part voxels extracted from the generated part meshes, thereby providing essential part geometry prior as follows:

$$\mathcal{M}^{p'} = \mathcal{R}(\mathcal{O}^p, \mathbf{V}^p), \mathbf{V}^p = \text{Voxelize}(\mathcal{T}(\mathcal{M}^p)), \quad (8)$$

where \mathcal{R} represents the stage II enhancer, and \mathcal{T} denotes a transformation to normalize \mathcal{M}^p to $[-1, 1]$. We can further enhance each part efficiently and assemble the parts by using the inverse transformation \mathcal{T}^{-1} :

$$\mathcal{M}' = \{\mathcal{T}^{-1}(\mathcal{M}^{p'})\}_{p=1}^N. \quad (9)$$



Figure 3. Comparison with state-of-the-art 3D generators. *CoPart* can generate detailed and independent 3D parts.

3.4. Optimization Loss

We supervise both the 3D and image branches of *CoPart* by the denoising loss in diffusion models:

$$Loss_{3D} = \frac{1}{N} \sum_{p=1}^N (\mathbb{E}_{\mathbf{L}_{3D}^{p,t}, \epsilon_{3d}^p, t} \|\epsilon_{3d}^p - \mathcal{N}_{3d}(\mathbf{L}_{3D}^{p,t}, \mathbf{L}_{2D}^{p,t}, t)\|_2^2), \quad (10)$$

$$Loss_{2D} = \frac{1}{N} \sum_{p=1}^N (\mathbb{E}_{\mathbf{L}_{2D}^{p,t}, \epsilon_{2d}^p, t} \|\epsilon_{2d}^p - \mathcal{N}_{2d}(\mathbf{L}_{3D}^{p,t}, \mathbf{L}_{2D}^{p,t}, t)\|_2^2),$$

where \mathcal{N} represents the denoiser for 3D and 2D and ϵ denotes Gaussian noise.

4. Applications

One major advantage of our part-based representation of *CoPart* is that it enables us to directly achieve various applications without further training. These applications include part-based editing (Sec. 4.1), articulated object generation (Sec. 4.2), mini-scene generation and long part sequence sampling (Sec. 4.3).

4.1. Part-based Editing

To enable selective part modification while keeping other parts unchanged, we design an inference-time resampling strategy. Specifically, given a mesh parts sequence $\{\mathcal{M}^{p'}\}_{p=1}^N$ either from *CoPart* sampling or segmented from an existing mesh, we denote the parts need to be edited as $\{\mathcal{M}^{p'}\}_{p \in \mathcal{C}}$, where \mathcal{C} is the selected index. To maintain the remaining parts unchanged during sampling while allowing them to provide information for new parts, we first encode the remaining part mesh back into contextual part latents:

$$\begin{aligned} \mathbf{L}_{3D}^{p,0'} &= \mathcal{E}_{3D}(\text{Sample}(\mathcal{M}^{p'})) \\ \mathbf{L}_{2D}^{p,0'} &= \mathcal{E}_{2D}(\text{Render}(\mathcal{M}^{p'})). \end{aligned} \quad (11)$$

Then during each timestep of the new editing sampling process, we directly replace the noisy part latents $\{\mathbf{L}_{3D}^{p,t}, \mathbf{L}_{2D}^{p,t}\}_{p \notin \mathcal{C}}$ by adding noise to $\{\mathbf{L}_{3D}^{p,0'}, \mathbf{L}_{2D}^{p,0'}\}_{p \notin \mathcal{C}}$:

$$\begin{aligned} \widehat{\mathbf{L}}_{3D}^{p,t} &= \sqrt{\bar{\alpha}_t} \mathbf{L}_{3D}^{p,0'} + \sqrt{1 - \bar{\alpha}_t} \epsilon_{3d}, p \notin \mathcal{C} \\ \widehat{\mathbf{L}}_{2D}^{p,t} &= \sqrt{\bar{\alpha}_t} \mathbf{L}_{2D}^{p,0'} + \sqrt{1 - \bar{\alpha}_t} \epsilon_{2d}, p \notin \mathcal{C}, \end{aligned} \quad (12)$$

where $\bar{\alpha}_t$ is noise schedule and ϵ is random Gaussian noise for 3D or 2D. Thus, we can sample additional part latents from pure Gaussian noise while incorporating information from the fixed one by:

$$\mathbf{L}_{3D}^{p,t-1}, \mathbf{L}_{2D}^{p,t-1} = \mathcal{N}(\{\widehat{\mathbf{L}}_{3D}^{p,t}, \widehat{\mathbf{L}}_{2D}^{p,t}\}_{p \notin \mathcal{C}}, \{\mathbf{L}_{3D}^{p,t}, \mathbf{L}_{2D}^{p,t}\}_{p \in \mathcal{C}}), \quad (13)$$

where \mathcal{N} is the diffusion denoiser. In this way, we can modify the text prompts to edit the parts as shown in Fig. 5 (a).

4.2. Articulated Object Generation

The generation of articulated objects involves two key components: i) articulation generation, which includes the bounding boxes indicating the position of each part, and ii) part generation. To achieve this, we leverage an off-the-shelf method [28] to generate the bounding boxes along with their articulation relationships. Subsequently, we utilize *CoPart* to populate each bounding box with coherent parts based on text prompts. This approach enables the creation of novel articulated objects, such as an avocado swivel chair (Fig. 1), which cannot be realized using previous holistic generation methods. Additionally, the part-based generation approach provides the flexibility to manually define articulation information for each part. Please refer to the supplementary video for the visualization of generated articulated objects.

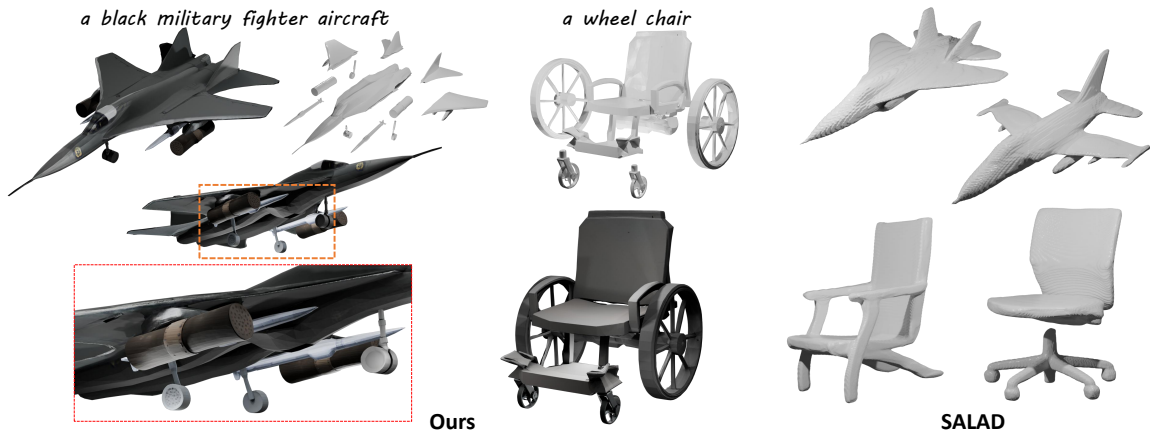


Figure 4. Comparison with part-based generator SALAD.

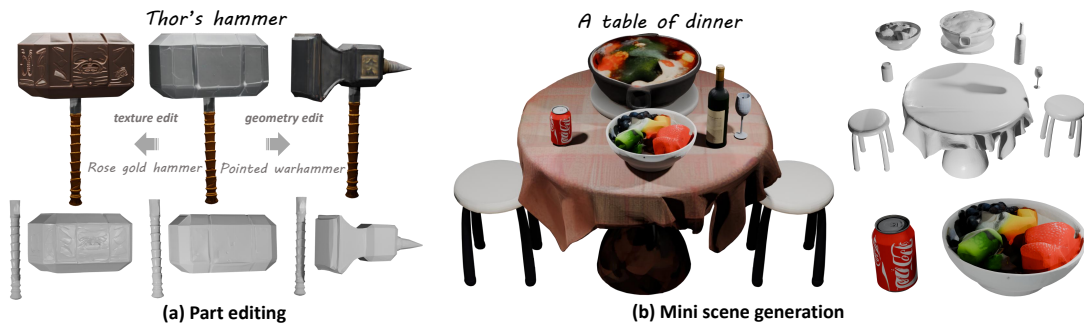


Figure 5. Qualitative results of part editing and mini scene generation.

4.3. Mini-scene and Long Sequence Generation

Our approach extends naturally to mini-scene generation, as scenes can be represented as layouts of bounding boxes, aligning with our box-guided part generation where each part corresponds to an object. Our training data includes mini-scenes, enabling direct scene generation through specified boxes and text (Fig. 5 (b)). For complex objects requiring long part sequences, GPU memory constraints during training limit the maximum number of parts $N=8$. We address it by adopting the strategy from Sec. 4.1: fixing some sampled part latents while replacing others with new box conditions sampled from Gaussian noise. This enables the generation of longer sequences without memory issues. More details can be found in the supplementary materials.

5. PartVerse Dataset

To enhance the generalizability of *CoPart*, we introduce *PartVerse*, a new diverse 3D part dataset comprising 91k high-quality parts from 12k objects with detailed text descriptions. Unlike previous part datasets such as PartNet [32] which only contains 24 categories of daily objects, our *PartVerse*, curated and annotated from Objaverse [7], exhibits enhanced diversity in 175 categories and realistic textures, significantly improving the model’s ability to generate high-quality 3D parts. We provide an data collection and annotation pipeline, more details can be found in the supplementary material.

Automatic Mesh Segmentation. 3D artists follow a spe-

cific modeling pipeline when creating 3D objects, typically creating them part by part. We can restore this information from the raw 3D data. However, the original modeling steps might not always match semantic part boundaries - for instance, some artists might create textured surfaces as separate elements. A pre-labeling algorithm based on SAM-2 [39] and Samesh [43] was constructed using 3D model creation priors combined with semantic segmentation, which balances the mesh faces connectivity and visual semantic during segmentation. This algorithm can allow us to preliminarily obtain semantic parts. We specifically adjusted parameters to favor over-segmentation rather than under-segmentation, since it’s easier for annotators to merge extra segments than to split insufficient ones.

Human post-annotation. After initial segmentation, human reviewers first remove low-quality data including overly complex or unsuitable objects for splitting. Using a Blender-based [3] annotation platform, they then refine the segmentation by merging over-segmented or splitting under-segmented parts according to guidelines: ensuring clear part semantics and maximizing symmetry in part distribution. In this way, complete textured objects are finally split into individual part objects with textures preserved.

6. Experimental Evaluation

This section presents our experimental results. For better visualization, please refer to our supplementary video.

Implementation details. We initialize our 2D and 3D de-

Table 1. Quantitative comparison with SOTA methods. CLIP (N-T) and CLIP (I-T) [38, 56] gauge the geometry alignment of normal maps with the input text and similarity of render images with the input text, respectively. In addition, ULIP-T [49] was also experimented and user-prefer study were conducted. † Our method takes 12s when the number of parts is one.

Method	Whole-aware			Part-aware			Time
	CLIP (N-T)	CLIP (I-T)	ULIP-T	CLIP (N-T)	CLIP (I-T)	ULIP-T	
Shap-E [18]	0.1546	0.1607	0.1054	0.0875	0.1043	0.0795	3s
Unique3D [47]	0.1865	0.2037	0.1528	0.1062	0.1380	0.1032	16s
CraftMan [23]	0.1887	0.1966	0.1476	0.1026	0.1271	0.0997	8s
Rodin [1]	<u>0.2042</u>	0.2416	0.1785	<u>0.1425</u>	<u>0.1571</u>	<u>0.1244</u>	-
Trellis [48]	0.2071	0.2360	<u>0.1751</u>	0.1274	0.1455	0.1119	10s
Ours	0.2010	<u>0.2387</u>	0.1743	0.1607	0.1768	0.1355	65s [†]

noisers using Pixart- α [4] and CraftMan [23], and fine-tune them on PartVerse dataset with 4 NVIDIA A100 GPUs. We set the batch size to 1 and limit the maximum part number to 8. We also perform random selecting or padding during training.

Comparison with state-of-the-art 3D generators. We compare *CoPart* with leading holistic 3D generators in Fig. 3. More results can be found in the supplementary materials. It is evident that *CoPart* outperforms state-of-the-art methods, particularly in the quality of small parts, owing to its part-based representation.

Quantitative comparison. As shown in Tab. 1, we conduct quantitative comparisons following [56]. For the Image-to-3D model, we

Table 2. User study (%preference).

Method	Whole-aware	Part-aware
Rodin [1]	33.3%	25.5%
PartGen [5]	11.8%	13.7%
Ours	54.9%	60.8%

first generate corresponding images from given texts by Flux.1 [50]. Different from [56], half of the test cases we use are part-aware, such as “a rifle stock”. This is reasonable, since a truly general 3D generation model should be able to handle all types. In addition, ULIP [49] was used in evaluation. As shown in Tab. 2, we also conducted a user study with 51 diverse participants from different professions by collecting their preferences for generating textured mesh. These results highlights the advantages of our part-based generation approach in producing decomposable and high-quality 3D assets.

Comparison with part-based generation methods. We compare *CoPart* with the accessible state-of-the-art part generator SALAD [21]. Fig. 4 shows that *CoPart* can generate diverse objects with detailed parts while SALAD is constrained to generate shapes in PartNet distribution. More comparisons can be found in the supplementary materials.

Ablation study of global guidance. We ablate the effect of global guidance in Fig. 6. The results demonstrate that global guidance significantly enhances part coherence, especially in appearance.

Ablation study of refinement. We ablate the effect of refinement (Sec. 3.3), as shown in Fig. 7. The results show that providing both part images (Fig. 7 (a)) and geometries

(Fig. 7 (b)) is essential for the enhancer to accurately comprehend part shapes. By integrating both modalities, the enhancer optimizes the parts and overall performance.



Figure 6. Ablation of global guidance.

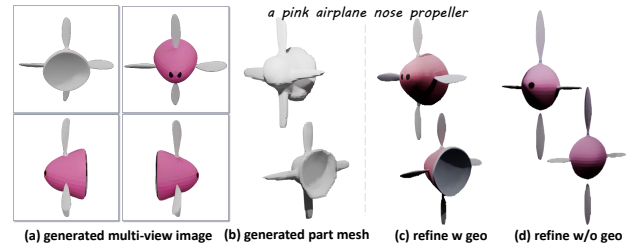


Figure 7. Ablation of refinement module.

7. Discussion of Part Assembly

As depicted in Sec. 3.3, the part latents are decoded in absolute positions and then normalized to perform refinement. After an inverse-normalization, the refined part meshes can be relocated to the 3D bonding boxes for assembling. Our precise strategy for injecting bounding box information generally ensures effective combinations of part meshes. However, assembly errors, such as “mesh clipping”, can occur when the user provides incorrect bounding boxes. We will explore techniques to optimize user-provided bounding boxes in future work.

8. Conclusion

We present *CoPart* for high-quality and diverse 3D part generation. Specifically, we utilize mutual guidance to ensure coherent part latent denoising and introduce 3D box conditions to eliminate part ambiguity. Furthermore, a larger scale 3D part-aware dataset is firstly collected from Objaverse, which can be widely used for various tasks. Our method outperforms SoTA results. We also discuss the limitations of our method in the supplementary material.

Acknowledgments

The work is supported in part by the Early Career Scheme of the Research Grants Council (RGC) of the Hong Kong SAR under grant No. 26202321, ITF PRP/046/24FX, SAIL Research Project, HKUST-Zeekr Collaborative Research Fund, and CUHK-CUHKSZ-GZ 1+1+1 Joint Collaboration Fund No. 4760964. We also gratefully acknowledge the support of SenseTime.

References

- [1] Rodin. <https://hyper3d.ai/>. 8
- [2] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *ICML*, 2018. 3
- [3] Blender Foundation. Blender, 2023. 7
- [4] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart- α : Fast training of diffusion transformer for photorealistic text-to-image synthesis, 2023. 2, 4, 8
- [5] Minghao Chen, Roman Shapovalov, Iro Laina, Tom Monnier, Jianyuan Wang, David Novotny, and Andrea Vedaldi. Partgen: Part-level 3d generation and reconstruction with multi-view diffusion models. *arXiv preprint arXiv:2412.18608*, 2024. 3, 8
- [6] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *CVPR*, 2019. 3
- [7] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *CVPR*, 2023. 2, 7
- [8] Lihe Ding, Shaocong Dong, Zhanpeng Huang, Zibin Wang, Yiyuan Zhang, Kaixiong Gong, Dan Xu, and Tianfan Xue. Text-to-3d generation with bidirectional diffusion using both 2d and 3d priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5115–5124, 2024. 2, 4
- [9] 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. *ACM Transactions on Graphics (ToG)*, 38(1):1–15, 2019. 3
- [10] 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. *ACM Transactions on Graphics (TOG)*, 38(6):1–15, 2019. 2
- [11] 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/CVF international conference on computer vision*, pages 7154–7164, 2019. 3
- [12] 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
- [13] Philipp Henzler, Niloy J. Mitra, and Tobias Ritschel. Escaping plato’s cave: 3d shape from adversarial rendering. In *ICCV*, 2019. 3
- [14] Amir Hertz, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or. Spaghetti: Editing implicit shapes through part aware generation. *ACM Transactions on Graphics (TOG)*, 41(4):1–20, 2022. 3
- [15] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023. 3
- [16] Ka-Hei Hui, Ruihui Li, Jingyu Hu, and Chi-Wing Fu. Neural template: Topology-aware reconstruction and disentangled generation of 3d meshes. In *CVPR*, 2022. 3
- [17] Moritz Ibing, Gregor Kobsik, and Leif Kobbelt. Oc-tree transformer: Autoregressive 3d shape generation on hierarchically structured sequences. *arXiv preprint arXiv:2111.12480*, 2021. 3
- [18] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 8
- [19] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2
- [20] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 3
- [21] Juil Koo, Seungwoo Yoo, Minh Hieu Nguyen, and Minhyuk Sung. Salad: Part-level latent diffusion for 3d shape generation and manipulation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14441–14451, 2023. 2, 3, 8
- [22] Jun Li, Kai Xu, Siddhartha Chaudhuri, Ersin Yumer, Hao Zhang, and Leonidas Guibas. Grass: Generative recursive autoencoders for shape structures. *ACM Transactions on Graphics (TOG)*, 36(4):1–14, 2017. 3
- [23] Weiyu Li, Jiarui Liu, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman: High-fidelity mesh generation with 3d native generation and interactive geometry refiner. *arXiv preprint arXiv:2405.14979*, 2024. 2, 3, 4, 8
- [24] Yanwei Li, Yuechen Zhang, Chengyao Wang, Zhisheng Zhong, Yixin Chen, Ruihang Chu, Shaoteng Liu, and Jiaya Jia. Mini-gemini: Mining the potential of multi-modality vision language models. *arXiv preprint arXiv:2403.18814*, 2024. 3
- [25] Connor Lin, Niloy Mitra, Gordon Wetzstein, Leonidas J Guibas, and Paul Guerrero. Neuform: Adaptive overfitting for neural shape editing. *NeurIPS*, 2022. 3
- [26] 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 *CVPR*, 2023. 3
- [27] Anran Liu, Cheng Lin, Yuan Liu, Xiaoxiao Long, Zhiyang Dou, Hao-Xiang Guo, Ping Luo, and Wenping Wang. Part123: part-aware 3d reconstruction from a single-view

- image. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–12, 2024. 2, 3
- [28] Jiayi Liu, Hou In Ivan Tam, Ali Mahdavi-Amiri, and Manolis Savva. Cage: Controllable articulation generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17880–17889, 2024. 3, 6
- [29] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*, 2023. 3
- [30] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2, 3
- [31] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas J Guibas. StructureNet: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575*, 2019. 3
- [32] 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. 2, 7
- [33] George Kiyohiro Nakayama, Mikaela Angelina Uy, Jiahui Huang, Shi-Min Hu, Ke Li, and Leonidas Guibas. Diffacto: Controllable part-based 3d point cloud generation with cross diffusion. In *CVPR*, 2023. 2, 3
- [34] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, , and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 3
- [35] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022. 4
- [36] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2022. 3
- [37] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023. 3
- [38] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 8
- [39] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 7
- [40] Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4), 2023. 3, 12
- [41] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2, 3
- [42] Edward Smith and David Meger. Deep unsupervised learning using nonequilibrium thermodynamics. In *CoRL*, 2017. 3
- [43] George Tang, William Zhao, Logan Ford, David Benhaim, and Paul Zhang. Segment any mesh: Zero-shot mesh part segmentation via lifting segment anything 2 to 3d. *arXiv preprint arXiv:2408.13679*, 2024. 7
- [44] Konstantinos Tertikas, Despoina Paschalidou, Boxiao Pan, Jeong Joon Park, Mikaela Angelina Uy, Ioannis Emiris, Yanis Avrithis, and Leonidas Guibas. Generating part-aware editable 3d shapes without 3d supervision. In *CVPR*, 2023. 3
- [45] 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. *NeurIPS*, 2024. 3
- [46] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. In *NeurIPS*, 2016. 3
- [47] Kailu Wu, Fangfu Liu, Zhihan Cai, Runjie Yan, Hanyang Wang, Yating Hu, Yueqi Duan, and Kaisheng Ma. Unique3d: High-quality and efficient 3d mesh generation from a single image. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 8
- [48] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024. 5, 8
- [49] Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *CVPR*, 2023. 8
- [50] Chenglin Yang, Celong Liu, Xueqing Deng, Dongwon Kim, Xing Mei, Xiaohui Shen, and Liang-Chieh Chen. 1.58-bit flux. *arXiv preprint arXiv:2412.18653*, 2024. 8
- [51] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *ICCV*, 2019. 3
- [52] Jie Yang, Kaichun Mo, Yu-Kun Lai, Leonidas J Guibas, and Lin Gao. Dsg-net: Learning disentangled structure and geometry for 3d shape generation. *ACM Transactions on Graphics (TOG)*, 42(1):1–17, 2022. 3
- [53] Xianghui Yang, Huiwen Shi, Bowen Zhang, Fan Yang, Jiacheng Wang, Hongxu Zhao, Xinhai Liu, Xinzhou Wang, Qingxiang Lin, Jiaao Yu, Lifu Wang, Zhuo Chen, Sicong Liu, Yuhong Liu, Yong Yang, Di Wang, Jie Jiang, and Chunchao Guo. Tencent hunyuan3d-1.0: A unified framework for text-to-3d and image-to-3d generation, 2024. 3

- [54] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–16, 2023. [2](#), [12](#)
- [55] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. [5](#)
- [56] Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024. [2](#), [3](#), [8](#)