VoxHammer: Training-Free Precise and Coherent 3D Editing in Native 3D Space

Lin Li[1,2*] Zehuan Huang[1*†] Haoran Feng[3] Gengxiong Zhuang[1] Rui Chen[1]
Chunchao Guo[4] Lu Sheng[1 ✉]

[1]Beihang University [2]Renmin University of China [3]Tsinghua University [4]Tencent Hunyuan

Project page: https://huanngzh.github.io/VoxHammer-Page/

Figure 1. High-quality 3D assets edited by our method using text prompts. Our method uses a training-free approach to perform percise and coherent 3D local editing, transforming multiple 3D assets in the scene (left) into high-quality results (right). The bottom row shows a detailed comparison of each 3D asset before and after editing, as well as the conditioning texts.

Abstract

3D local editing of specified regions is crucial for the game industry and robot interaction. Recent methods typically edit rendered multi-view images and then reconstruct 3D models, but they face challenges in precisely preserving unedited regions and overall coherence. Inspired by structured 3D generative models, we propose VoxHammer, a novel training-free approach that performs precise and coherent editing in 3D latent space. Given a 3D model, VoxHammer first predicts its inversion trajectory and obtains its inverted latents and key-value tokens at each timestep. Subsequently, in the denoising and editing phase, we replace the denoising features of preserved regions with the corresponding inverted latents and cached key-value tokens. By retaining these contextual features, this approach ensures consistent reconstruction of preserved areas and coherent integration of edited parts. To evaluate the consistency of preserved regions, we constructed Edit3D-Bench, a human-annotated dataset comprising hundreds of samples, each with carefully labeled 3D editing regions. Experiments demonstrate that VoxHammer significantly outperforms existing methods in terms of both 3D consistency of preserved regions and overall quality. Our method holds promise for synthesizing high-quality edited paired data, thereby laying the data foundation for in-context 3D generation.

* Equal contribution † Project lead ✉ Corresponding author

1. Introduction

In recent years, the rapid advancement of generative AI has greatly facilitated the creation of 3D assets [32, 47, 53, 62, 90, 97], providing powerful production tools for industries such as gaming, robotics, and VR. Among these, 3D local editing [2, 6, 41, 71, 72, 106] is a crucial task that enables partial modification of existing or AI-generated 3D assets while keeping other regions unchanged. It presents challenges in maintaining consistency in preserved regions and ensuring overall coherence in the edited model.

Existing 3D editing methods can be broadly categorized into two pipelines. One approach [6, 13, 15, 51, 71, 106] employs Score Distillation Sampling (SDS) [62] to optimize 3D representations so that it aligns with input prompts, but per-scene editing typically takes minutes or even hours. Another approach [1, 5, 20, 26, 42, 63] attempts to edit multi-view images [34, 73] rendered from the 3D model and then reconstruct the 3D model from the modified views. These techniques achieve higher efficiency through a feed-forward process. However, editing in 2D space instead of 3D space usually introduces position bias in the 3D reconstruction stage, making accurate local editing difficult. In addition, inconsistencies among edited multi-view images [2, 4, 20] lead to artifacts in the edited 3D model, compromising quality and coherence [42, 104].

Recently, advanced 3D generative models [43, 47, 48, 86, 90, 96, 97], trained in native 3D space, can generate high-fidelity 3D content from text or image prompts. These models exhibit significant advantages in 3D consistency and quality, inspiring us to edit 3D assets in native 3D space. However, fine-tuning these models for editing is constrained by a critical data bottleneck: large-scale paired datasets for 3D local editing are exceptionally difficult to acquire. Therefore, our research focuses on unleashing the potential of pretrained 3D generative models for precise and coherent 3D editing, eliminating the need for additional training.

We propose *VoxHammer*, a training-free framework for precise and coherent 3D editing. Our method is based on a pretrained structured 3D latent diffusion model [90], and introduce a two-stage process: precise 3D inversion, and denoising and editing based on the inverted latents. Given a 3D model (mesh, NeRF [58], or Gaussian Splat [38]), *VoxHammer* first predicts its inversion trajectory of 3D diffusion process and caches its inverted latents and key-value tokens at each timestep. We demonstrate that the inversion can reconstruct the given model's 3D geometry and texture with high precision. In the subsequent denoising and editing phase, we denoise the edited region, and replace the denoising features of preserved regions with the corresponding inverted latents and cached key-value tokens. By retaining these contextual features, our approach ensures consistent reconstruction of preserved areas and coherent integration of edited parts. This is achieved without training the base model and only through feature replacement at inference, allowing high-quality 3D local editing at minimal cost.

The lack of labeled editing regions in existing datasets makes it challenging to objectively evaluate consistency in preserved areas. To address this, we constructed *Edit3D-Bench*, a human-annotated dataset comprising hundreds of samples with carefully labeled 3D editing regions. Quantitative and qualitative experiments on *Edit3D-Bench* show that *VoxHammer* significantly outperforms existing methods in terms of both editing accuracy and overall quality. Our training-free method also holds promise for synthesizing high-quality edited paired data, thereby laying the data foundation for in-context 3D generation. Our main contributions are summarized as follows:

- We propose a training-free, native 3D local editing framework that leverages a pretrained 3D generative model for highly precise and coherent editing.
- We introduce precise 3D inversion and denoising-based editing using inverted latents, where we replace inverted latents and key-value tokens in the 3D latent space to ensure consistent reconstruction of preserved regions and coherent editing.
- We built *Edit3D-Bench*, a benchmark to thoroughly validate the superiority of *VoxHammer* in both editing accuracy and overall quality. Our method holds promise for synthesizing high-quality edited paired data, thereby laying the data foundation for in-context 3D generation.

2. Related Work

3D generative models. Recent advances in diffusion models [31, 74] and the availability of high-quality 3D datasets [11, 12] have significantly accelerated the development of 3D generative models [8, 9, 14, 25, 27, 28, 32, 35, 43, 45, 50, 52–55, 57, 67, 75, 77, 80–84, 86, 89, 91, 95, 97, 101, 103]. Some methods [33, 34, 54, 55, 64, 75, 77, 81, 83, 91] generate 3D models by first synthesizing multi-view images and then reconstructing 3D from these views. But inconsistent multi-view synthesis may lower the quality of the final 3D model. A series of methods [10, 16, 43, 44, 46, 47, 49, 76, 85–87, 90, 97, 102, 103] train native 3D generative models that comprise of a variational autoencoder [39] and a diffusion transformer (DiT) [61] for denoising in latent space. These approaches unify 3D generation with high fidelity and consistency, laying the foundation for downstream inversion and editing.

3D editing. Early 3D editing methods [6, 13, 15, 51, 71, 100, 106] employ Score Distillation Sampling (SDS) [62] to optimize 3D representation to align with the input prompts, but the per-scene editing requires minutes or even hours. Subsequent works [1, 2, 4, 5, 20, 26, 42, 63, 104] attempt to edit multi-view images rendered from the 3D model and reconstruct 3D from the modified views. But the lack of con-

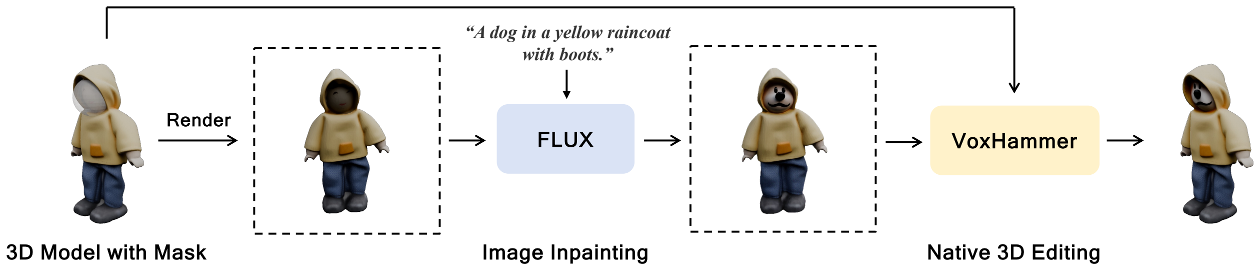

Figure 2. **Pipeline.** Given an input 3D model, a user-specified editing region, and a text prompt, the off-the-shelf models [3, 40] are used to inpaint the rendered view from the 3D model. Subsequently, our *VoxHammer*, a training-free framework based on structured 3D diffusion models [90], performs native 3D editing conditioned on the input 3D and the edited image.

sistency across edited images frequently leads to degraded reconstruction quality. In contrast, our method operates directly in the native 3D space, enabling precise and coherent editing that integrates seamlessly with the overall structure.

Image generation and editing. Diffusion models [21, 31, 40, 66, 68, 70, 74] synthesize images through a gradual denoising process starting from standard noise. To enable image editing with pretrained image diffusion models, several methods [18, 23, 24, 37, 59, 69, 78, 105] employ inversion techniques, which map real images into the diffusion model's denoising trajectory for precise controllable manipulation. Inspired by these approaches, we explore inversion within native 3D generative models [90] and further propose a native 3D editing framework. Our method achieves training-free precise and coherent 3D editing through 3D inversion and novel contextual feature replacement.

3. Methodology

We propose *VoxHammer*, a training-free framework for 3D local editing, aimed at achieving precise and globally coherent modifications on 3D models. As shown in Fig. 2, given an input 3D model (mesh, NeRF [58], or 3DGS [38]), an editing region and a text prompt, our framework first renders a view from the 3D model and utilizes advanced image diffusion models [3, 40] to produce an inpainted image. Subsequently, *VoxHammer* performs native 3D editing conditioned on the input 3D model and the edited image.

The illustration of *VoxHammer* is shown in Fig. 3. We first invert the 3D asset to noise and cache the inverted latents and key-value tokens at each timestep, which are then masked and used to guide the denoising process. We first introduce our base model [90] and inversion design [78] in Sec. 3.1. Then, Sec. 3.2 explores the 3D inversion and validates its consistency in geometry and texture reconstruction. Based on these findings, Sec. 3.3 presents our training-free 3D local editing method through inversion and re-editing.

3.1. Preliminary

Structured 3D diffusion models. *VoxHammer* is based on structured 3D latent diffusion models [88, 90], which are generative models that operate in a sparse voxel-based latent space for high-quality and scalable 3D generation. They represent a 3D asset (mesh, NeRF [58], or 3DGS [38]) as structured latents (SLAT), i.e., a set of local latent vectors $\{(z_i, p_i)\}_{i=1}^{L}$ anchored to active voxels p_i that intersect the object surface, where each $z_i \in \mathbb{R}^C$ encodes fine-scale geometry and appearance. During inference, the model first samples noise in the voxel-based latent space, followed by a two-stage denoising process. In the first stage, referred to as the **structure (ST) stage**, the diffusion model predicts voxel occupancy over a 64^3 grid to obtain sparse structures, where each location corresponds to a surface-intersecting voxel in the coarse space. In the second stage, called the **sparse-latent (SLAT) stage**, the structured latents are denoised to produce fine-grained geometry and texture, thereby enhancing the visual fidelity of the 3D output.

Rectified Flow Inversion. Several methods [37, 59, 69, 78] explore flow inversion for downstream editing tasks. The challenges associated with achieving accurate inversion stem from the accumulation of numerical errors during ODE integration, which lead to noticeable deviations in the reconstructed samples. To address this, RF-Solver [78] introduces a training-free, plug-and-play sampler that improves inversion fidelity by analytically approximating the ODE solution using high-order Taylor expansion. This formulation significantly reduces integration errors and enables more faithful reconstructions. Given a state x_0^{ss}, it integrate the rectified-flow ODE from data to noise using a second-order Taylor-improved Euler scheme:

$$x_{t-\Delta} = x_t + \Delta\, f_\theta(x_t, t) + \frac{1}{2}\Delta^2\, \partial_t f_\theta(x_t, t) \quad (1)$$

$$\partial_t f_\theta(x_t, t) \approx \frac{f_\theta(x_{t-\Delta/2}, t - \Delta/2) - f_\theta(x_t, t)}{\Delta/2} \quad (2)$$

where x_t represents the state of the data at the current time t, and $x_{t-\Delta}$ is the predicted state for the next denoising step, which is a time interval Δ away. $f_\theta(x_t, t)$ denotes the noise-prediction network, and $\partial_t f_\theta(x_t, t)$ is the second equation

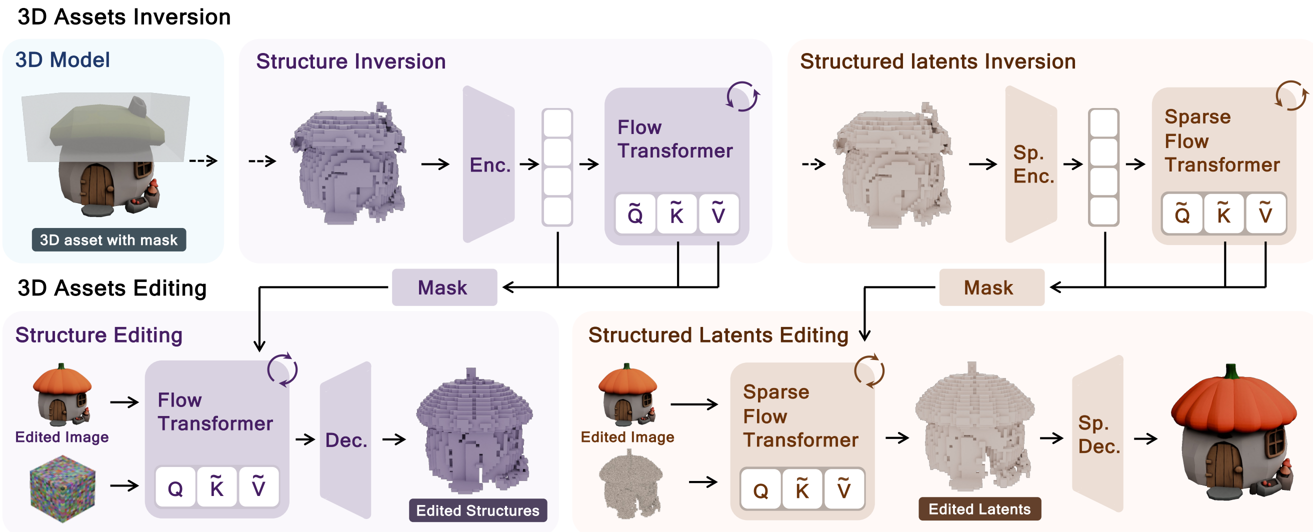

Figure 3. **Architecture of VoxHammer.** Our framework adopts TRELLIS [90] as the base model, which predicts sparse structures at the first structure (ST) stage and denoise fine-grained structured latents at the second sparse-latent (SLAT) stage. *VoxHammer* performs inversion prediction in both the ST and SLAT stages, which map the textured 3D asset to its terminal noise, with latents and key/value tensors cached at each timestep. Subsequently, *VoxHammer* denoises from the inverted noise, and replace the features of the preserved regions with the corresponding cached latents and key-value tokens, thereby achieving precise and coherent editing in native 3D space.

approximates using a finite difference scheme. The Taylor-improved update yields a local truncation error $\mathcal{O}(\Delta_t^3)$ and a global error $\mathcal{O}(\Delta_t^2)$, which is crucial for high-fidelity reconstructions of fine structures. Inspired by it, we explore inversion within native 3D generative models [90] and further propose our precise and coherent 3D editing framework through our novel contextual feature replacement.

3.2. 3D Inversion

We introduce an inversion prediction strategy within the structured 3D generation pipeline [90] to map the textured 3D asset to its terminal noise. The inversion proceeds in both the structure (ST) stage and the sparse-latent (SLAT) stage, with latents and key/value (K/V) tensors cached at each time step for later reuse during editing.

The inherent invertibility of the underlying Flux model enables this inversion strategy. In the forward pass, the pipeline generates assets by following a predefined discrete time schedule, denoted as $0 = s_0 < s_1 < \cdots < s_T = 1$. To perform the inversion, we reverse the execution of this schedule by traversing the trajectory from timestep s_T back to s_0. This reversed traversal deterministically traces the generation path backward, allowing us to map a final 3D asset to its corresponding source noise.

During the ST stage, the K, V tensors from all attention layers are stored into a dictionary $\mathcal{KV}^{\text{st}}$ indexed by latent time, block order, positional encoding, layer ID, and attention type. In the SLAT stage, we first extract the preserved set Ω_{keep} from the decoded output of the ST stage by removing the edit voxels, normalize the features, and run the same Taylor-improved inversion scheme. During this process, K/V tensors are cached into $\mathcal{KV}^{\text{slat}}$. Throughout both stages, we apply classifier-free guidance (CFG) [30] only in a late-time interval $t \in [0.5, 1.0]$:

$$f_{\text{cfg}} = (1+\omega)\, f_\theta(\text{cond}) - \omega\, f_\theta(\text{neg}) \tag{3}$$

and revert to $f_\theta(\text{cond})$ otherwise, thereby stabilizing early inversion steps and improving fidelity. In practice, we keep ω fixed and activate guidance only in the late interval, which preserves the invertibility of early steps while providing sufficient semantic sharpness for the features in Ω_{keep}. During the next editing stage (Sec. 3.3), the denoising features of the preserved regions are directly overwritten by the inverted source features, ensuring geometric and textural fidelity in unedited regions.

3.3. 3D Editing

Based on the inversion strategy, we further introduce a training-free local editing strategy, where the model denoises from the inverted noise, and performs latent replacement and key-value replacement on the unedited regions. Both operations are guided by 3D edit masks, to achieve precise preservation in unedited regions.

Latent replacement. In the structure (ST) stage, latent replacement is performed using a binary edit mask $M^{\text{ss}} \in \{0,1\}^{H\times W\times D}$. At each denoising step t, the latent is blended with the inverted source latent $\hat{z}_t^{\text{ss}}$:

$$z_t^{\text{ss}} \leftarrow M^{\text{ss}} \odot z_t^{\text{ss}} + (1 - M^{\text{ss}}) \odot \hat{z}_t^{\text{ss}} \tag{4}$$

Table 1. **Quantitative comparison on our Edit3D-Bench**. We compute Chamfer Distance (CD.), masked PSNR, SSIM, LPIPS of unedited region to evaluate 3D consistency, FID, FVD to evaluate overall 3D quality, and DINO-I and CLIP-T to assess condition alignment.

Method	Unedited Region Preservation				Overall 3D Quality		Condition Alignment	
	CD. ↓	PSNR (M) ↑	SSIM (M) ↑	LPIPS (M) ↓	FID ↓	FVD ↓	DINO-I ↑	CLIP-T ↑
Vox-E [71]	/	13.84	0.827	0.316	87.41	3000.3	0.721	0.274
MVEdit [5]	0.017	26.12	0.945	0.070	58.53	946.5	0.911	0.281
Tailor3D [63]	0.043	20.94	0.861	0.148	110.52	3812.1	0.704	0.258
Instant3DiT [2]	0.016	27.70	0.957	0.067	45.93	450.1	0.903	0.260
TRELLIS [90]	0.047	23.64	0.919	0.131	38.19	757.2	0.911	0.283
Ours (full)	**0.012**	**41.68**	**0.994**	**0.027**	**23.05**	**187.8**	**0.947**	**0.287**
w/o Attn KV	0.015	35.71	0.986	0.042	27.68	361.8	0.938	0.285
w/ Noise Re-init	0.014	36.31	0.989	0.038	25.71	259.6	0.945	0.287

To mitigate visible seams at mask boundaries, M^{ss} can be replaced with a soft mask $\widetilde{M}^{\text{ss}} \in [0,1]^{H\times W\times D}$ obtained by dilation and Gaussian falloff. In the sparse-latent (SLAT) stage, features at the unedited coordinate set Ω_{keep} are replaced with the inverted source latent at each denoising step:

$$\forall\, \mathbf{u} \in \Omega_{\text{keep}} : \quad z_t^{\text{slat}}[\mathbf{u}] \leftarrow \hat{z}_t^{\text{slat}}[\mathbf{u}] \tag{5}$$

Similar to the ST stage, coordinates near the boundaries can be weighted to ensure smooth transitions, effectively applying a soft-mask-like effect.

Key-value replacement. Beyond latent replacement, feature-level consistency is enforced by our proposed key-value replacement in the attention mechanism. In the ST stage, self–attention use binary masks W^{self} to indicate edited tokens. During editing, K/V tensors in unedited regions are replaced by their cached counterparts:

$$K \leftarrow W \odot K_{\text{new}} + (1 - W) \odot K_{\text{cache}} \tag{6}$$
$$V \leftarrow W \odot V_{\text{new}} + (1 - W) \odot V_{\text{cache}} \tag{7}$$

Optional attention masks can be supplied to attention calculation to block mixing between edited and preserved tokens, which is especially helpful when the edited region is small but semantically strong. Similar to the solution used in inversion, the model also adopts strategies of rescaled time scheduling and late-time CFG [30] in the denoise phase. All the above modifications are implemented by dynamically adjusting forward functions at inference time, without retraining or weight updates.

4. Experiments

4.1. Setup

Implementation details. Our method is built on TRELLIS [90] and executed on a single NVIDIA A100 GPU. We set the sampling steps to 25 for both the inversion and denoising phases, and set the classifier-free guidance (CFG) scales of both stages to 5.0 to balance reconstruction fidelity and edit creativity.

Baselines. We adopt five competitive 3D editing methods as baselines. Vox-E [71] performs per-scene optimization on voxel representation with the guidance of image diffusion models. MVEdit [5], Tailor3D [63] and Instant3dit [2] achieves customized 3D asset editing through multi-view editing. TRELLIS [90] provides a native 3D editing method based on RePaint [56], which guides the denoising process by sampling from the known regions.

Evaluation dataset. We conduct an evaluation on *Edit3D-Bench*, a new benchmark we curated for systematic 3D editing assessment. It consists of 100 high-quality 3D models, with 50 carefully selected from Google Scanned Objects (GSO) [19] and 50 from PartObjaverse-Tiny [92]. For each model, we provide 3 distinct editing prompts that cover a wide range of modifications. Each prompt is accompanied by a complete set of annotated 3D assets, including the 2D renderings of the original object, a 2D mask of the edit region, a 2D edited image generated by FLUX.1 Fill [3], illustrating the intended target edit, and a 3D mask specifying the precise editing region in 3D space. This well-structured dataset serves as a rigorous benchmark for assessing the accuracy, robustness, and fidelity of 3D editing methods.

Evaluation metrics. We use three aspects of metrics to comprehensively evaluate performance. First, for unedited region preservation, we assess the fidelity of preserved regions by computing Chamfer Distance (CD) [22] for geometry consistency, as well as masked PSNR, SSIM [79], and LPIPS [99] on rendered multi-view images for texture. Second, for overall 3D quality, we evaluate holistic performance by computing FID [29] on rendered images and conducting a user study to capture human perceptual preferences. Finally, for the alignment with the input prompts, we evaluate the alignment of the edited 3D assets with the edited image using DINO-I [60], and its alignment with the text prompt using CLIP-T [65].

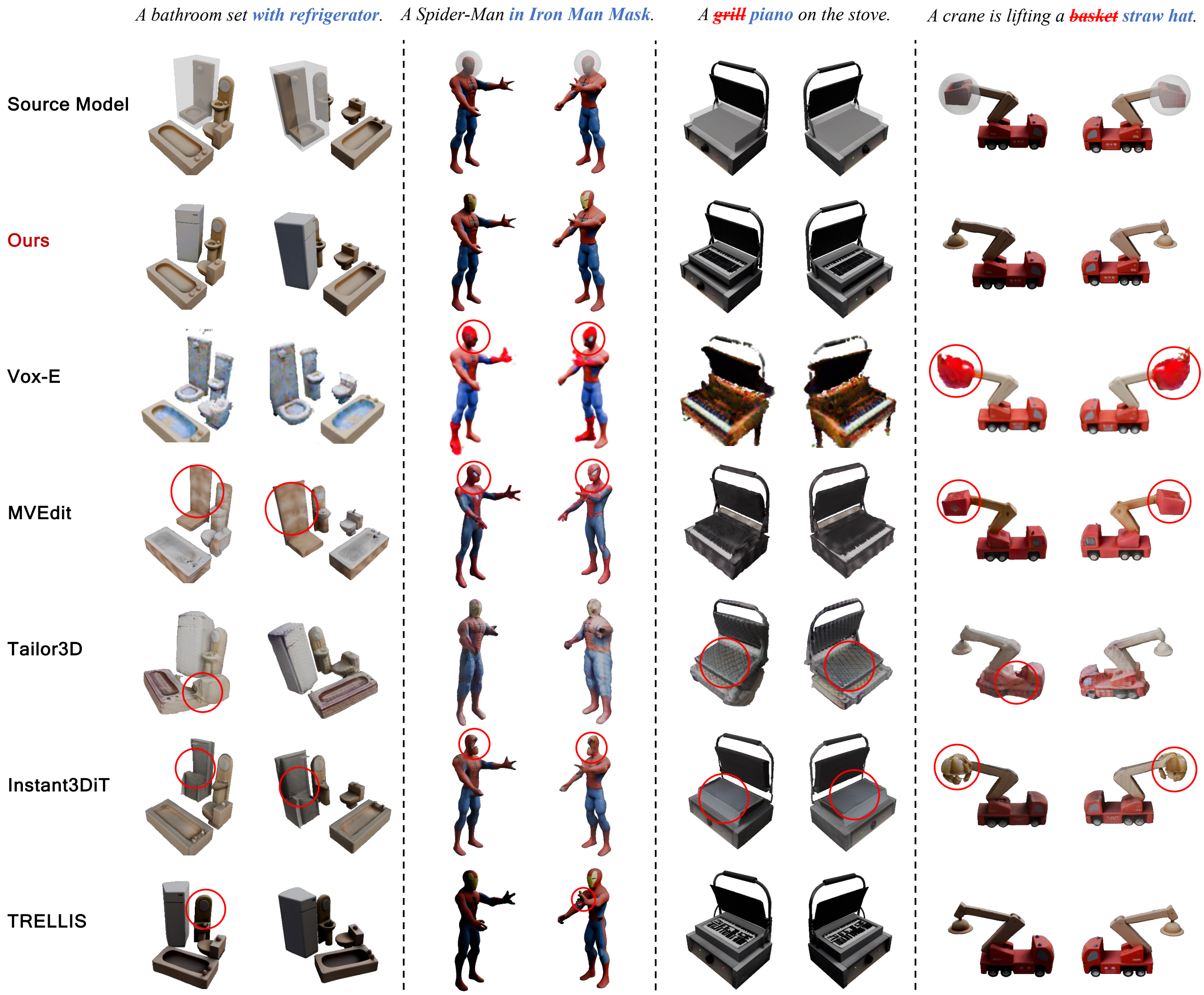

Figure 4. **Qualitative comparisons on Edit3D-Bench.** Our method achieves best performance on precision of editing and overall quality.

4.2. Main Results

Quantitative comparison. As shown in Tab. 1, our method significantly outperforms all baselines across nearly all metrics. For unedited region preservation, our approach achieves the best scores in Chamfer Distance, PSNR, SSIM, and LPIPS, demonstrating its superior ability to maintain the original geometry and texture. This is because our method operates directly in the native 3D space and leverages latents and key-value replacement to explicitly enforce consistency. In contrast, multi-view based methods like MVEdit [5] and Instant3DiT [2], which rely on lifting multi-view edits to 3D space, usually introduce multi-view inconsistency and spatial bias, and struggle with maintaining 3D consistency. TRELLIS [90] adapts Repaint [56] to achieve native 3D editing, but lacks inversion and key-value replacement to introduce the context of the reserved region, thus showing poor performance in 3D consistency. For overall 3D quality and condition alignment, our method further demonstrates superiority by achieving the lowest FID and the highest DINO-I and CLIP-T scores. These results collectively indicate that our editing operations yield coherent and accurate outcomes in high-fidelity 3D models.

Qualitative comparison. The qualitative results in Fig. 4 highlight the superiority of our method. Our approach consistently generates edits that are both locally accurate and geometrically coherent, while preserving unedited regions with high fidelity. In contrast, the baselines exhibit various artifacts. Some methods suffer from poor reconstruction quality, leading to blurry outputs and distortions in preserved regions, as observed in Vox-E [71] and Tai-

Figure 5. **Ablation studies.** Results demonstrate the effectiveness of key-value replacement in attention mechanism and latent replacement.

Table 2. **User preference study.** Our method achieves consistently higher preference than Instant3DiT and TRELLIS in text alignment and overall 3D quality.

Method	Text Alignment ↑	Overall 3D Quality ↑
Instant3DiT [2]	4.7%	1.6%
TRELLIS [90]	25.0%	17.2%
Ours	**70.3%**	**81.2%**

Table 3. **Analysis of two-stage inversion.** We report Chamfer Distance (CD.) and PSNR, SSIM, LPIPS of rendered views to analyze the reconstruction consistency.

Inversion Stage	CD. ↓	PSNR ↑	SSIM ↑	LPIPS ↓
ST stage	0.0094	37.68	0.936	0.067
ST + SLAT stage	**0.0055**	**39.70**	**0.987**	**0.012**

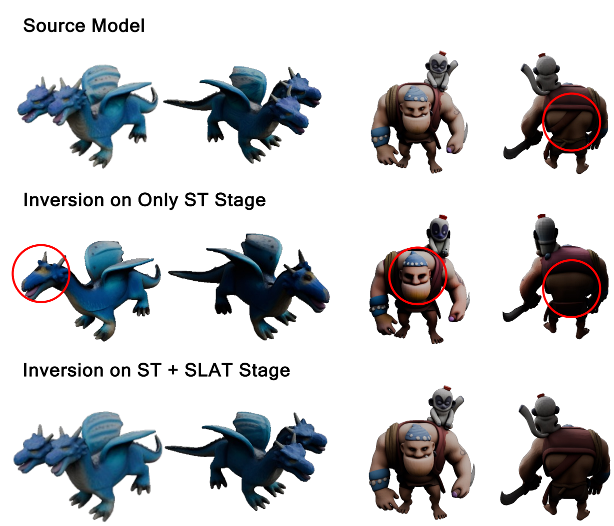

Figure 6. **The impact of inversion stages on reconstruction.** ST stage inversion lacks detailed consistency, while inversion on both stages achieves fine-grained geometry and texture reconstruction.

lor3D [63]. Others are overly conservative, retaining most of the original content with only minimal edits, as is the case for MVEdit [5]. Instant3DiT [2], by contrast, often fails to generate results that align with text prompts, resulting in misplaced or misrepresented modifications. In particular, the native TRELLIS [90] editing method, lacking inversion and KV-cache mechanisms, fails to effectively constrain the original 3D structure, often resulting in positional shifts and inconsistencies in the preserved regions. Our method effectively avoids these issues, demonstrating the robustness of our native 3D, inversion-based editing framework.

User study. To evaluate the perceptual quality of our editing results, we conducted a user study with 30 participants. For each editing task, participants were presented with the input 3D models and the edit prompts, paired with the edited 3D assets by *VoxHammer* alongside two strong baseline methods TRELLIS [90] and Instant3DiT [2]. They were then asked to select the result that best aligned with the text prompt and exhibited the highest overall quality. The study results demonstrate a clear human preference for our method, confirming its superior performance and robustness across multiple evaluation aspects.

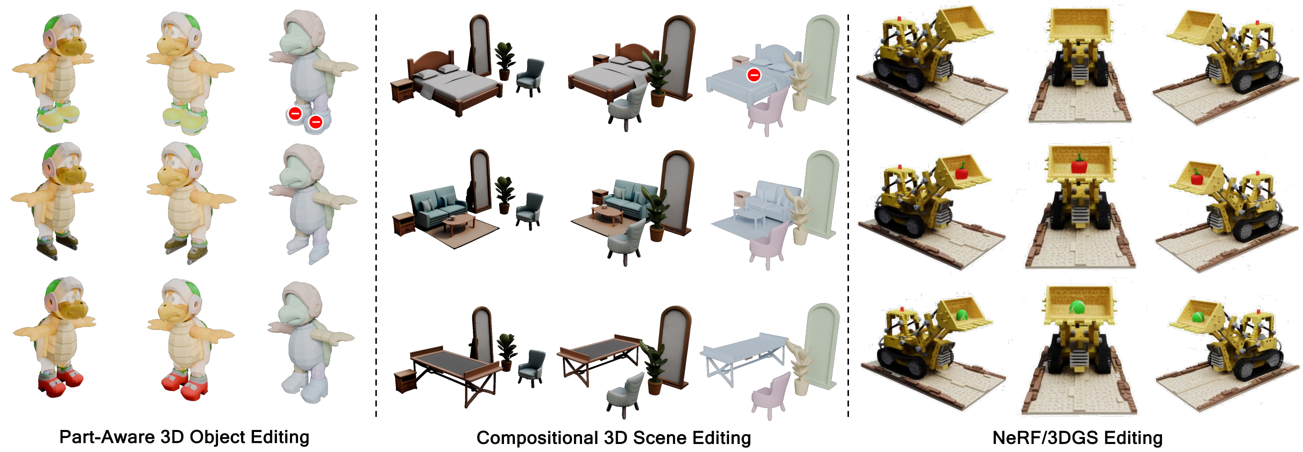

Figure 7. **More applications.** *VoxHammer* easily generalizes to part-aware 3D object, scene, and NeRF [58] or 3DGS [38] editing. We show the input models in the top row and the edited results in the bottom two rows.

4.3. Ablation Study

We conduct a series of ablation studies to analyze the reconstruction performance of two-stage inversion, and the contributions of the key components of our editing strategies, particularly the feature replacement strategy and the inversion-based initialization.

Analysis of two-stage inversion. We conducted analysis of doing inversion (1) *only in ST stage*, and (2) *in both ST and SLAT stage*, which is our full setting. As shown in Tab. 3 and Fig. 6, we found that the initial ST stage inversion provides a reasonable reconstruction of coarse geometry, but lacks detailed geometry and appearance consistency. After incorporating the second SLAT stage, which handles high-resolution geometry and fine-grained texture details, the reconstruction quality improves significantly. It demonstrates that our two-stage inversion process faithfully restores the source 3D model with high fidelity, offering a robust basis for the subsequent editing stage.

Assessment of editing strategies. We compare our full setting against four variants: (1) *w/o Attn KV*, where we disable key-value replacement for attention mechanism; and (2) *w/ Noise Re-init*, where we initiate the denoising process from randomly sampled Gaussian noise instead of the inverted noise of the source 3D asset. The quantitative results are presented in Tab. 1, and a qualitative comparison is shown in Fig. 5. Missing key-value replacement leads to a significant degradation in preservation quality, as the edit concept leaks into unedited regions. Re-initializing noise leads to a loss of positional information, causing unexpected alterations in preserved areas. In contrast, our full setting effectively maintains the consistency of the preserved regions and achieves overall coherent editing, which validates that our inversion and key-value replacement is essential for achieving high-fidelity local 3D editing.

4.4. More Applications

Part-aware object editing. *VoxHammer* enables flexible editing of part-aware generated 3D assets [7, 17, 49, 50, 76, 92, 93, 98, 102], where the pre-segmented structure offers 3D masks for us. We report visualization results in Fig. 7.

Compositional 3D scene editing. *VoxHammer* further extends to compositional 3D scene editing [36, 94]. As shown in Fig. 7, it supports fine-grained local modifications while preserving the integrity of the surrounding scene.

NeRF or 3DGS editing. Benefiting from the versatility of the base model, *VoxHammer* also generalizes NeRF [58] or 3DGS [38] editing, as visualized in Fig. 7.

5. Conclusion

We presented *VoxHammer*, a training-free framework for precise and coherent 3D local editing. By leveraging accurate 3D inversion and feature replacement in the latent space of a pretrained structured 3D diffusion model, *VoxHammer* preserves unedited regions with high fidelity while seamlessly integrating edits. To enable objective evaluation, we introduced *Edit3D-Bench*, a human-annotated benchmark for 3D local editing. Comprehensive experiments demonstrate that our method outperforms prior approaches in both consistency and quality. Beyond editing, *VoxHammer* also enables the synthesis of paired data, laying a foundation for future in-context 3D generation.

Acknowledgment

This work was supported by National Natural Science Foundation of China (62132001), the Fundamental Research Funds for the Central Universities, and Xiaomi Young Talents Program/Xiaomi Foundation.

References

[1] Roi Bar-On, Dana Cohen-Bar, and Daniel Cohen-Or. Editp23: 3d editing via propagation of image prompts to multi-view, 2025. 2

[2] Amir Barda, Matheus Gadelha, Vladimir G. Kim, Noam Aigerman, Amit H. Bermano, and Thibault Groueix. Instant3dit: Multiview inpainting for fast editing of 3d objects, 2024. 2, 5, 6, 7

[3] Black Forest Labs. FLUX.1 Tools: Introducing Fill, Depth, Canny, and Redux. https://bfl.ai/blog/24-11-21-tools, 2024. Accessed: 2025-08-15. 3, 5

[4] Chenjie Cao, Chaohui Yu, Fan Wang, Xiangyang Xue, and Yanwei Fu. Mvinpainter: Learning multi-view consistent inpainting to bridge 2d and 3d editing, 2024. 2

[5] Hansheng Chen, Ruoxi Shi, Yulin Liu, Bokui Shen, Jiayuan Gu, Gordon Wetzstein, Hao Su, and Leonidas Guibas. Generic 3d diffusion adapter using controlled multi-view editing, 2024. 2, 5, 6, 7

[6] Minghao Chen, Junyu Xie, Iro Laina, and Andrea Vedaldi. Shap-editor: Instruction-guided latent 3d editing in seconds, 2023. 2

[7] Minghao Chen, Jianyuan Wang, Roman Shapovalov, Tom Monnier, Hyunyoung Jung, Dilin Wang, Rakesh Ranjan, Iro Laina, and Andrea Vedaldi. Autopartgen: Autogressive 3d part generation and discovery. *arXiv preprint arXiv:2507.13346*, 2025. 8

[8] Sijin Chen, Xin Chen, Anqi Pang, Xianfang Zeng, Wei Cheng, Yijun Fu, Fukun Yin, Yanru Wang, Zhibin Wang, Chi Zhang, Jingyi Yu, Gang Yu, Bin Fu, and Tao Chen. Meshxl: Neural coordinate field for generative 3d foundation models, 2024. 2

[9] Yiwen Chen, Tong He, Di Huang, Weicai Ye, Sijin Chen, Jiaxiang Tang, Xin Chen, Zhongang Cai, Lei Yang, Gang Yu, Guosheng Lin, and Chi Zhang. Meshanything: Artist-created mesh generation with autoregressive transformers, 2024. 2

[10] Yiwen Chen, Zhihao Li, Yikai Wang, Hu Zhang, Qin Li, Chi Zhang, and Guosheng Lin. Ultra3d: Efficient and high-fidelity 3d generation with part attention, 2025. 2

[11] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2

[12] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[13] Nam Anh Dinh, Itai Lang, Hyunwoo Kim, Oded Stein, and Rana Hanocka. Geometry in style: 3d stylization via surface normal deformation, 2025. 2

[14] Junting Dong, Qi Fang, Zehuan Huang, Xudong Xu, Jingbo Wang, Sida Peng, and Bo Dai. Tela: Text to layer-wise 3d clothed human generation. In *European Conference on Computer Vision*, pages 19–36. Springer, 2025. 2

[15] Shaocong Dong, Lihe Ding, Zhanpeng Huang, Zibin Wang, Tianfan Xue, and Dan Xu. Interactive3d: Create what you want by interactive 3d generation, 2024. 2

[16] Shaocong Dong, Lihe Ding, Xiao Chen, Yaokun Li, Yuxin Wang, Yucheng Wang, Qi Wang, Jaehyeok Kim, Chenjian Gao, Zhanpeng Huang, Zibin Wang, Tianfan Xue, and Dan Xu. From one to more: Contextual part latents for 3d generation, 2025. 2

[17] Shaocong Dong, Lihe Ding, Xiao Chen, Yaokun Li, Yuxin Wang, Yucheng Wang, Qi Wang, Jaehyeok Kim, Chenjian Gao, Zhanpeng Huang, et al. From one to more: Contextual part latents for 3d generation. *arXiv preprint arXiv:2507.08772*, 2025. 8

[18] Wenkai Dong, Song Xue, Xiaoyue Duan, and Shumin Han. Prompt tuning inversion for text-driven image editing using diffusion models, 2023. 3

[19] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items, 2022. 5

[20] Ziya Erkoç, Can Gümeli, Chaoyang Wang, Matthias Nießner, Angela Dai, Peter Wonka, Hsin-Ying Lee, and Peiye Zhuang. Preditor3d: Fast and precise 3d shape editing, 2024. 2

[21] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024. 3

[22] Haoqiang Fan, Hao Su, and Leonidas Guibas. A point set generation network for 3d object reconstruction from a single image, 2016. 5

[23] Haoran Feng, Zehuan Huang, Lin Li, Hairong Lv, and Lu Sheng. Personalize anything for free with diffusion transformer, 2025. 3

[24] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022. 3

[25] Daoyi Gao, Yawar Siddiqui, Lei Li, and Angela Dai. Meshart: Generating articulated meshes with structure-guided transformers, 2025. 2

[26] Will Gao, Dilin Wang, Yuchen Fan, Aljaz Bozic, Tuur Stuyck, Zhengqin Li, Zhao Dong, Rakesh Ranjan, and Nikolaos Sarafianos. 3d mesh editing using masked lrms, 2024. 2

[27] Zekun Hao, David W. Romero, Tsung-Yi Lin, and Ming-Yu Liu. Meshtron: High-fidelity, artist-like 3d mesh generation at scale, 2024. 2

[28] Zexin He, Tengfei Wang, Xin Huang, Xingang Pan, and Ziwei Liu. Neural lightrig: Unlocking accurate object normal and material estimation with multi-light diffusion, 2024. 2

[29] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. 5

[30] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4, 5

[31] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 3

[32] 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. 2

[33] Xiufeng Huang, Ka Chun Cheung, Runmin Cong, Simon See, and Renjie Wan. Stereo-gs: Multi-view stereo vision model for generalizable 3d gaussian splatting reconstruction, 2025. 2

[34] Zehuan Huang, Yuan-Chen Guo, Haoran Wang, Ran Yi, Lizhuang Ma, Yan-Pei Cao, and Lu Sheng. Mv-adapter: Multi-view consistent image generation made easy. *arXiv preprint arXiv:2412.03632*, 2024. 2

[35] Zehuan Huang, Hao Wen, Junting Dong, Yaohui Wang, Yangguang Li, Xinyuan Chen, Yan-Pei Cao, Ding Liang, Yu Qiao, Bo Dai, et al. Epidiff: Enhancing multi-view synthesis via localized epipolar-constrained diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9784–9794, 2024. 2

[36] Zehuan Huang, Yuan-Chen Guo, Xingqiao An, Yunhan Yang, Yangguang Li, Zi-Xin Zou, Ding Liang, Xihui Liu, Yan-Pei Cao, and Lu Sheng. Midi: Multi-instance diffusion for single image to 3d scene generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23646–23657, 2025. 8

[37] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12469–12478, 2024. 3

[38] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 3, 8

[39] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2

[40] Black Forest Labs. Flux. https://github.com/black-forest-labs/flux, 2024. 3

[41] Haoxuan Li, Ziya Erkoc, Lei Li, Daniele Sirigatti, Vladyslav Rozov, Angela Dai, and Matthias Nießner. Meshpad: Interactive sketch-conditioned artist-reminiscent mesh generation and editing, 2025. 2

[42] Peng Li, Suizhi Ma, Jialiang Chen, Yuan Liu, Congyi Zhang, Wei Xue, Wenhan Luo, Alla Sheffer, Wenping Wang, and Yike Guo. Cmd: Controllable multiview diffusion for 3d editing and progressive generation, 2025. 2

[43] 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

[44] Weiyu Li, Jiarui Liu, Hongyu Yan, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman3d: High-fidelity mesh generation with 3d native generation and interactive geometry refiner, 2025. 2

[45] Weiyu Li, Xuanyang Zhang, Zheng Sun, Di Qi, Hao Li, Wei Cheng, Weiwei Cai, Shihao Wu, Jiarui Liu, Zihao Wang, Xiao Chen, Feipeng Tian, Jianxiong Pan, Zeming Li, Gang Yu, Xiangyu Zhang, Daxin Jiang, and Ping Tan. Step1x-3d: Towards high-fidelity and controllable generation of textured 3d assets, 2025. 2

[46] Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, and Yan-Pei Cao. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models, 2025. 2

[47] Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *arXiv preprint arXiv:2502.06608*, 2025. 2

[48] Zhihao Li, Yufei Wang, Heliang Zheng, Yihao Luo, and Bihan Wen. Sparc3d: Sparse representation and construction for high-resolution 3d shapes modeling. *arXiv preprint arXiv:2505.14521*, 2025. 2

[49] Yuchen Lin, Chenguo Lin, Panwang Pan, Honglei Yan, Yiqiang Feng, Yadong Mu, and Katerina Fragkiadaki. Partcrafter: Structured 3d mesh generation via compositional latent diffusion transformers, 2025. 2, 8

[50] 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, 8

[51] Fangfu Liu, Hanyang Wang, Weiliang Chen, Haowen Sun, and Yueqi Duan. Make-your-3d: Fast and consistent subject-driven 3d content generation, 2024. 2

[52] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10072–10083, 2024. 2

[53] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[54] 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

[55] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9970–9980, 2024. 2

[56] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint:

Inpainting using denoising diffusion probabilistic models, 2022. 5, 6

[57] Quan Meng, Lei Li, Matthias Nießner, and Angela Dai. Lt3sd: Latent trees for 3d scene diffusion. *arXiv preprint arXiv:2409.08215*, 2024. 2

[58] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2, 3, 8

[59] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023. 3

[60] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2024. 5

[61] 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. 2

[62] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion, 2022. 2

[63] Zhangyang Qi, Yunhan Yang, Mengchen Zhang, Long Xing, Xiaoyang Wu, Tong Wu, Dahua Lin, Xihui Liu, Jiaqi Wang, and Hengshuang Zhao. Tailor3d: Customized 3d assets editing and generation with dual-side images, 2024. 2, 5, 7

[64] Yansong Qu, Shaohui Dai, Xinyang Li, Yuze Wang, You Shen, Liujuan Cao, and Rongrong Ji. Deocc-1-to-3: 3d de-occlusion from a single image via self-supervised multi-view diffusion, 2025. 2

[65] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 5

[66] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022. 3

[67] Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulø, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3dg: Latent 3d gaussian diffusion. *arXiv preprint arXiv:2410.13530*, 2024. 2

[68] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022. 3

[69] Litu Rout, Yujia Chen, Nataniel Ruiz, Constantine Caramanis, Sanjay Shakkottai, and Wen-Sheng Chu. Semantic image inversion and editing using rectified stochastic differential equations. *arXiv preprint arXiv:2410.10792*, 2024. 3

[70] 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. 3

[71] Etai Sella, Gal Fiebelman, Peter Hedman, and Hadar Averbuch-Elor. Vox-e: Text-guided voxel editing of 3d objects, 2023. 2, 5, 6

[72] Etai Sella, Noam Atia, Ron Mokady, and Hadar Averbuch-Elor. Blended point cloud diffusion for localized text-guided shape editing. *arXiv preprint arXiv:2507.15399*, 2025. 2

[73] 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

[74] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2, 3

[75] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 2

[76] Jiaxiang Tang, Ruijie Lu, Zhaoshuo Li, Zekun Hao, Xuan Li, Fangyin Wei, Shuran Song, Gang Zeng, Ming-Yu Liu, and Tsung-Yi Lin. Efficient part-level 3d object generation via dual volume packing, 2025. 2, 8

[77] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pages 439–457. Springer, 2025. 2

[78] Jiangshan Wang, Junfu Pu, Zhongang Qi, Jiayi Guo, Yue Ma, Nisha Huang, Yuxin Chen, Xiu Li, and Ying Shan. Taming rectified flow for inversion and editing. *arXiv preprint arXiv:2411.04746*, 2024. 3

[79] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. 5

[80] Zhengyi Wang, Jonathan Lorraine, Yikai Wang, Hang Su, Jun Zhu, Sanja Fidler, and Xiaohui Zeng. Llama-mesh: Unifying 3d mesh generation with language models, 2024. 2

[81] Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. Crm: Single image to 3d textured mesh with convolutional reconstruction model. *arXiv preprint arXiv:2403.05034*, 2024. 2

[82] Si-Tong Wei, Rui-Huan Wang, Chuan-Zhi Zhou, Baoquan Chen, and Peng-Shuai Wang. Octgpt: Octree-based multi-scale autoregressive models for 3d shape generation, 2025.

[83] Hao Wen, Zehuan Huang, Yaohui Wang, Xinyuan Chen, Yu Qiao, and Lu Sheng. Ouroboros3d: Image-to-3d generation via 3d-aware recursive diffusion. *arXiv preprint arXiv:2406.03184*, 2024. 2

[84] 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. *arXiv preprint arXiv:2405.20343*, 2024. 2

[85] Ruiqi Wu, Xinjie Wang, Liu Liu, Chunle Guo, Jiaxiong Qiu, Chongyi Li, Lichao Huang, Zhizhong Su, and Ming-Ming Cheng. Dipo: Dual-state images controlled articulated object generation powered by diverse data, 2025. 2

[86] Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Jingxi Xu, Philip Torr, Xun Cao, and Yao Yao. Direct3d: Scalable image-to-3d generation via 3d latent diffusion transformer. *arXiv preprint arXiv:2405.14832*, 2024. 2

[87] Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Yikang Yang, Yajie Bao, Jiachen Qian, Siyu Zhu, Xun Cao, Philip Torr, and Yao Yao. Direct3d-s2: Gigascale 3d generation made easy with spatial sparse attention, 2025. 2

[88] Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3r: Amodal 3d reconstruction from occluded 2d images, 2025. 3

[89] Zhennan Wu, Yang Li, Han Yan, Taizhang Shang, Weixuan Sun, Senbo Wang, Ruikai Cui, Weizhe Liu, Hiroyuki Sato, Hongdong Li, et al. Blockfusion: Expandable 3d scene generation using latent tri-plane extrapolation. *ACM Transactions on Graphics (TOG)*, 43(4):1–17, 2024. 2

[90] 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, 2025. 2, 3, 4, 5, 6, 7

[91] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. 2

[92] Yunhan Yang, Yukun Huang, Yuan-Chen Guo, Liangjun Lu, Xiaoyang Wu, Edmund Y. Lam, Yan-Pei Cao, and Xihui Liu. Sampart3d: Segment any part in 3d objects, 2024. 5, 8

[93] Yunhan Yang, Yufan Zhou, Yuan-Chen Guo, Zi-Xin Zou, Yukun Huang, Ying-Tian Liu, Hao Xu, Ding Liang, Yan-Pei Cao, and Xihui Liu. Omnipart: Part-aware 3d generation with semantic decoupling and structural cohesion. *arXiv preprint arXiv:2507.06165*, 2025. 8

[94] Kaixin Yao, Longwen Zhang, Xinhao Yan, Yan Zeng, Qixuan Zhang, Lan Xu, Wei Yang, Jiayuan Gu, and Jingyi Yu. Cast: Component-aligned 3d scene reconstruction from an rgb image. *ACM Transactions on Graphics (TOG)*, 44(4): 1–19, 2025. 8

[95] Junliang Ye, Zhengyi Wang, Ruowen Zhao, Shenghao Xie, and Jun Zhu. Shapellm-omni: A native multimodal llm for 3d generation and understanding, 2025. 2

[96] 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

[97] 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

[98] Longwen Zhang, Qixuan Zhang, Haoran Jiang, Yinuo Bai, Wei Yang, Lan Xu, and Jingyi Yu. Bang: Dividing 3d assets via generative exploded dynamics. *ACM Transactions on Graphics*, 44(4):1–21, 2025. 8

[99] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018. 5

[100] Yunzhi Zhang, Zizhang Li, Matt Zhou, Shangzhe Wu, and Jiajun Wu. The scene language: Representing scenes with programs, words, and embeddings, 2025. 2

[101] Ruowen Zhao, Junliang Ye, Zhengyi Wang, Guangce Liu, Yiwen Chen, Yikai Wang, and Jun Zhu. Deepmesh: Auto-regressive artist-mesh creation with reinforcement learning, 2025. 2

[102] Wang Zhao, Yan-Pei Cao, Jiale Xu, Yuejiang Dong, and Ying Shan. Assembler: Scalable 3d part assembly via anchor point diffusion, 2025. 2, 8

[103] Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, Bin Fu, Tao Chen, Gang Yu, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[104] Yang Zheng, Mengqi Huang, Nan Chen, and Zhendong Mao. Pro3d-editor : A progressive-views perspective for consistent and precise 3d editing, 2025. 2

[105] Tianrui Zhu, Shiyi Zhang, Jiawei Shao, and Yansong Tang. Kv-edit: Training-free image editing for precise background preservation, 2025. 3

[106] Jingyu Zhuang, Di Kang, Yan-Pei Cao, Guanbin Li, Liang Lin, and Ying Shan. Tip-editor: An accurate 3d editor following both text-prompts and image-prompts, 2024. 2