DreamEdit3D: Personalization of Multi-View Diffusion Models for 3D Editing

Jinxin Ai, Matthias Nießner, and Ziya Erkoç

Technical University of Munich, Germany

Fig. 1: DreamEdit3D produces multi-view consistent edits guided by natural language, given a source 3D object. We apply personalization to multi-view diffusion models to preserve the identity of the input shapes. We show that multiple diverse edits can be generated from one source by preserving the input.

Abstract. While 2D diffusion models have achieved remarkable success in identity-preserving personalization, extending this capability to 3D assets remains a significant challenge due to the complexities of multi-view consistency and spatial control. Inspired by these 2D advancements, we present a novel personalization method for text-guided 3D editing that enables compositional, object-level control through natural language. Given a 3D input, we render orthogonal views and extract object-level segmentation masks to isolate semantic components. We then learn distinct token embeddings for each component through a tailored two-phase optimization strategy: multi-view textual inversion with attention alignment, followed by full fine-tuning of multi-view diffusion model. During inference, these disentangled tokens seamlessly compose with editing prompts to generate multi-view consistent images, which are subsequently lifted into high-fidelity textured 3D meshes. Extensive evaluations across diverse editing scenarios demonstrate that our method

successfully transfers the flexibility of 2D personalization to 3D, achieving state-of-the-art edit faithfulness and identity preservation compared to existing baselines.

Keywords: 3D editing · Diffusion models · Textual inversion · Multi-view generation

1 Introduction

The rapid evolution of diffusion models has revolutionized the generation of visual content, extending remarkably from 2D images to 3D assets. Recent text-to-3D methodologies [23, 33, 47] have demonstrated impressive capabilities in synthesizing novel 3D objects from natural language descriptions. However, the editing of existing 3D assets remains a formidable challenge. In practical modeling workflows, users rarely want to regenerate an entire asset from scratch; rather, they require precise, compositional control to modify specific semantic parts of an object while strictly preserving the identity and geometry of the unedited regions. Existing text-guided 3D editing methods [6,9,39] often struggle with this, as global text prompts tend to entangle attributes, leading to unintended global modifications that destroy the original asset's core identity.

A compelling solution to identity preservation has recently emerged in the 2D domain. Personalization techniques, such as Textual Inversion and DreamBooth, have achieved remarkable success by learning unique token embeddings for specific subjects, allowing them to be seamlessly synthesized in novel contexts, styles, and forms without losing their defining characteristics [1, 11, 37]. Naturally, extending this paradigm to 3D editing is highly desirable. Yet, directly lifting 2D personalization to 3D is non-trivial. It introduces two major bottlenecks: the necessity of maintaining strict multi-view consistency across generated views, and the difficulty of spatially disentangling complex 3D objects into independently controllable semantic parts within the diffusion latent space.

Inspired by the success of 2D identity preservation, we present a novel, disentangled personalization framework designed specifically for compositional, text-guided 3D editing. Our core insight is that by explicitly isolating semantic components in the multi-view image space, we can learn distinct, disentangled token embeddings that bring the precise control of 2D personalization into the 3D domain.

To achieve this, our framework begins by rendering four orthogonal views of a given 3D input and extracting object-level segmentation masks to isolate its semantic parts. To ensure each learned token accurately represents its corresponding 3D component without bleeding into others, we propose a tailored, two-phase optimization strategy. In the first phase, we perform multi-view textual inversion guided by an attention alignment mechanism, forcing the cross-attention maps of the learned tokens to align precisely with the extracted segmentation masks across all views. In the second phase, we perform full UNet fine-tuning via joint multi-view training, which embeds strong multi-view consistency and structural priors directly into the model.

At inference, this disentangled representation unlocks highly flexible, object-level control. The optimized tokens can be combined with natural language editing prompts to independently modify specific parts of the subject. A multi-view diffusion model then synthesizes consistent, edited images across all views, which are finally lifted back into a high-fidelity, textured 3D mesh using 3D reconstruction techniques [20]. Extensive experiments demonstrate that our approach successfully translates the power of 2D personalization to 3D, allowing for complex, localized edits that maintain the highest fidelity to the original asset's unedited regions (Fig. 1). To summarize, our contributions are:

- We propose a disentangled token learning for multi-view diffusion models that decomposes 3D objects into editable semantic components through personalized token embeddings.
- We demonstrate the effectiveness of our approach on a diverse benchmark of editing scenarios, achieving favorable results compared to state-of-the-art on widely-used metrics such as CLIP and VLM-based evaluation.

2 Related Work

2.1 2D Editing

Early image editing methods operated in the latent spaces of GANs [14], enabling semantic manipulation through latent direction traversal [21], but were limited by inversion fidelity and the expressiveness of the learned space. The rise of text-to-image diffusion models [2,4,10,35,36] shifted editing toward prompt-driven manipulation via vision-language alignment. To improve spatial precision, Prompt-to-Prompt [18] redirects edits through cross-attention map manipulation, while inpainting-based methods [28] allow region-specific modification. There have been significant improvements in personalization-based editings to preserve the identity of the input [1,15,38,40,46,51]. DreamBooth [37] and Textual Inversion [11] bind visual concepts to learned text tokens via per-subject fine-tuning. ControlNet [53] further enables geometry-aware editing through auxiliary structural conditioning. Break-A-Scene [1] also employs token optimization to decompose scene into tokens based on the masks and can achieve identity-preserving editing.

2.2 Text-Guided 3D Editing

The text-guided 3D editing has been explored widely in the research community including gaussian-based, mesh-based ones as well as the ones leveraging 2D diffusion priors [3,6,7,16,30,31,44,48,52,54]. Vox-E [39] performs volumetric editing by distilling diffusion guidance into a voxel grid, enabling localized modifications through a volumetric attention mechanism. However, its reliance on score distillation sampling (SDS) makes optimization prohibitively slow, requiring approximately one hour per edit. MVEdit [6] proposes a multi-view editing pipeline that leverages 2D diffusion models to edit rendered views and reconstructs the

result into 3D. While MVEdit produces strong texture edits, it struggles to preserve object identity when changing the geometry of the object, such as pose modifications and redesign, often producing unrealistic colors.

To achieve text-guided fast 3D editing, PrEditor3D [9] applies DDPM inversion and Prompt-to-Prompt within a sparse multi-view diffusion model, followed by feed-forward 3D reconstruction via GTR [55] trained on large-scale datasets such as Objaverse. While efficient, it offers limited editing capability for keeping the shape consistent. In contrast, our approach learns object-level token embeddings for each object, enabling object-level editing (*e.g.*, reshaping or pose changes) and supports color/texture changes. Furthermore, our method is substantially faster than optimization-based approaches: training requires approximately a few minutes, and once trained, each edit including textured mesh reconstruction takes roughly two minutes at inference time.

2.3 Textual Inversion and Personalization

Textual inversion [11] introduced the idea of learning new token embeddings to represent specific visual concepts within the vocabulary of a pre-trained text-to-image diffusion model. The approach has been investigated further down the line [22,37,43]. DreamBooth [37] extends this idea by fine-tuning the diffusion model itself with a class-specific prior preservation loss, achieving higher fidelity to the target concept. Custom Diffusion [22] further improves efficiency by fine-tuning only the cross-attention layers and supports composing multiple concepts. Break-A-Scene [1] takes a complementary approach by learning *multiple* disentangled token embeddings from a single image, where each token captures a distinct object or region guided by segmentation masks and an attention-based loss that encourages spatial correspondence between tokens and image regions. In the 3D domain, DreamBooth3D [34] adapts DreamBooth for 3D-consistent generation from multi-view images. Break-A-Scene's disentangled token learning operates in the 2D domain. Our work extends this paradigm to 3D editing by operating on multi-view images rendered from a 3D mesh and introducing joint multi-view training on MVDream to ensure 3D-consistent token representations.

2.4 Multi-View Diffusion Models

Multi-view diffusion models generate consistent images from multiple viewpoints, enabling 3D-aware generation [13,24,26,27,42,45,50]. Zero-1-to-3 [24] fine-tunes a diffusion model to synthesize novel views given a single input image and a relative camera transformation. SyncDreamer [26] generates synchronized multi-view images by modeling cross-view attention within the diffusion process. MVDream [42] extends text-to-image diffusion models to generate multi-view consistent images by finetuning on large-scale 3D data, achieving strong 3D coherence while retaining the generative power of the 2D prior. We adopt MVDream [42] as our multi-view diffusion backbone and introduce a joint multi-view training strategy that optimizes disentangled token embeddings across all views simultaneously, ensuring that the learned representations maintain 3D consistency.

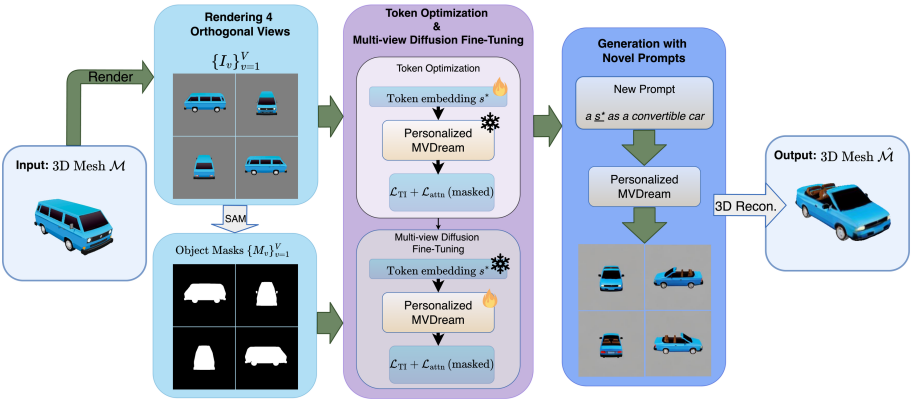

Fig. 2: Method overview. Top: Given a 3D mesh, we render four orthogonal views and obtain object masks via SAM. **Middle:** In Phase 1 (TI), a token embedding s^* is learned for the object through textual inversion on 4 views with a frozen UNet and attention alignment loss. In Phase 2 (DB), the full UNet is fine-tuned jointly across all 4 views with prior preservation. **Bottom:** At inference, tokens are composed with edit prompts, MVDream generates 4 consistent edited views, and GTR reconstructs the final 3D mesh.

3 Method

Given a 3D object represented as a textured mesh, our goal is to enable compositional, object-level editing through natural language. Our method consists of four stages: (1) rendering multi-view images and obtaining object semantic masks from the input mesh, (2a) learning disentangled token embeddings for each object via a two-phase optimization, (2b) jointly fine-tuning across views in Phase 2 to ensure 3D consistency, and (3) composing the learned tokens with editing prompts to generate modified consistent multi-view images that are reconstructed into an edited 3D mesh. Fig. 2 overviews our method.

3.1 Multi-View Rendering

Given an input 3D mesh $\mathcal{M}$, we render a set of N=4 views $\{I_v\}_{v=1}^{N}$ at orthogonal azimuths starting from 90° with a span of 360° and a fixed elevation of 15°, using a differentiable renderer. This four-view configuration matches the camera setup used by MVDream [42], trained to generate four consistent views.

3.2 Token Optimization

Our token optimization extends Break-A-Scene [1] to 3D by operating within MVDream [42] and introducing joint cross-view training for 3D consistency. We learn a token embedding s^* initialized from a semantically related word (*e.g.*, "robot," "dog") and optimize it in two phases.

Textual Inversion. In the first phase, we optimize only the token embedding s^* while keeping the entire UNet frozen. This phase operates on four orthogonal view images to efficiently learn the initial token representation. We optimize s^* by minimizing a masked diffusion denoising objective:

$$\mathcal{L}_{\text{TI}} = \mathbb{E}_{v,\epsilon,t}\left[\|(\epsilon - \epsilon_\theta(z_t^v, t, c(y))) \odot m\|_2^2\right] + \mu\mathcal{L}_{\text{attn}}, \tag{1}$$

where z_t^v is the noisy latent of the object image I_v at diffusion timestep t, ϵ is the sampled noise, ϵ_θ is the frozen UNet, $c(y)$ is the text conditioning with prompt $y =$ "a photo of s^*", and m is the object mask downsampled to the latent resolution. The mask ensures that the loss focuses on the object region.

As proposed by Break-A-Scene [1], the attention alignment loss $\mathcal{L}_{\text{attn}}$ encourages the cross-attention map of the learned token to spatially align with the ground-truth object mask:

$$\mathcal{L}_{\text{attn}} = \|A - \hat{M}\|_2^2, \tag{2}$$

where A is the aggregated cross-attention map for token s^* across UNet layers and $\hat{M}$ is the normalized ground-truth mask. This loss is weighted by μ and is applied only during Phase 1. To prevent catastrophic drift of the pre-trained vocabulary, we restore the embeddings of all non-learnable tokens after each optimization step, ensuring that only the new token embedding is modified.

UNet Fine-Tuning. Textual inversion alone may not capture fine-grained geometric and textural details, as the expressiveness is limited to the token embedding space. In the second phase, we unfreeze the full UNet and continue to optimize both the UNet parameters θ and the token embedding jointly. We optimize using the masked denoising objective with prior preservation:

$$\mathcal{L}_{\text{FT}} = \mathbb{E}_{v,\epsilon,t}\left[\|(\epsilon - \epsilon_{\theta'}(z_t^v, t, c(y))) \odot m\|_2^2\right] + \lambda\mathcal{L}_{\text{prior}}, \tag{3}$$

where θ' denotes the updated UNet parameters. The prior preservation loss $\mathcal{L}_{\text{prior}}$ [37] regularizes the fine-tuning to prevent language drift:

$$\mathcal{L}_{\text{prior}} = \mathbb{E}_{\epsilon,t}\left[\|\epsilon - \epsilon_{\theta'}(z_t^{\text{pr}}, t, c(y_{\text{pr}}))\|_2^2\right], \tag{4}$$

where z_t^{pr} are latents from class-prior images generated by the frozen model and y_{pr} is the class prompt. The attention alignment loss is not applied in Phase 2, as the UNet weights are now being modified. Rather than processing each view independently, which can lead to view dependent inconsistencies, we introduce a joint multi-view training strategy that leverages the multi-view generation architecture of MVDream to enforce 3D consistency.

Specifically, MVDream jointly processes multiple views using a shared UNet augmented with cross-view attention layers, enabling information exchange across viewpoints. We exploit this design by constructing training batches that contain all views of an object simultaneously. The corresponding latent representations $\{z_t^v\}_{v=1}^4$, together with their camera embeddings $\{e_v\}_{v=1}^4$, are fed into the UNet

in a single forward pass, allowing cross-view attention to promote coherent and consistent predictions across viewpoints:

$$\mathcal{L}_{\text{MV}} = \mathbb{E}_{\epsilon,t}\left[\sum_{v=1}^{4} \|\epsilon_v - \epsilon_{\theta'}(\{z_t^v\}_{v=1}^4, t, c(y), \{e_v\}_{v=1}^4)_v\|_2^2\right], \tag{5}$$

where $\epsilon_{\theta'}(\cdot)_v$ denotes the predicted noise corresponding to view v from the joint forward pass.

3.3 Compositional Editing and Reconstruction

Prompt Composition. Given an editing instruction for an object, we construct a prompt that uses the learned tokens with the desired modification. For example, to redesign a sofa to be "single seat," we compose the prompt: "a photo of s^* redesigned to single seat," where s^* is the learned token for the object whose identity should be preserved. This allows fine-grained control over appearance, geometry, and pose.

Multi-View Generation. Using the prompt y_{edit} and the fine-tuned UNet $\epsilon_{\theta'}$, we generate four consistent edited views $\{\hat{I}_v\}_{v=1}^4$ via the standard MVDream sampling procedure with classifier-free guidance [19]:

$$\begin{aligned}\tilde{\epsilon}_v = \epsilon_{\theta'}(\{z_t^v\}, t, \varnothing, \{e_v\}) \\ + w \cdot (\epsilon_{\theta'}(\{z_t^v\}, t, c(y_{\text{edit}}), \{e_v\}) - \epsilon_{\theta'}(\{z_t^v\}, t, \varnothing, \{e_v\}))\,,\end{aligned} \tag{6}$$

where w is the guidance scale and $\varnothing$ denotes the null text condition.

3D Reconstruction. The four generated views are reconstructed into a textured 3D mesh using GTR [55], a transformer-based large reconstruction model. GTR encodes the multi-view images into a triplane representation [5] via a transformer generator, extracts geometry through differentiable marching cubes [41], and applies a lightweight per-instance texture refinement procedure.

4 Experiments

4.1 Experimental Setup

Benchmark. We construct a benchmark of 25 diverse editing cases spanning different object categories and edit types. Each case consists of a source 3D mesh paired with a text editing prompt. The benchmark covers a range of editing scenarios including attribute transfer ("dog as a cat," "dog as a pig"), style transfer ("koala in lego style"), pose modifications ("cheetah lying on the floor," "robot sitting"), object addition ("basket with apples," "cake in a plate"), appearance editing ("person smile with teeth," "person wearing sunglasses"), and shape redesign ("sofa redesigned to single seat").

Baselines. We evaluate our method against several state-of-the-art text-guided 3D editing approaches. These include MVEdit [6], which reconstructs 3D results from 2D multi-view diffusion edits; Vox-E [39], which distills diffusion guidance into a voxel grid for volumetric editing; and PrEditor3D [9], a training-free framework leveraging 3D priors. We further compare our performance against recent advancements in the field, including Steer3D [29], Edit-TRELLIS [25, 49], GaussCtrl [48], GaussianEditor [8], and Instruct-NeRF2NeRF [17].

Evaluation Metrics. Following the evaluation protocol of PrEditor3D [9], we evaluate all methods using the following metrics, computed over 70 rendered views per object:

- **CLIP Directional Similarity** (CLIP_{dir}): Measures whether the change from the original to the edited rendering aligns with the change from the source prompt to the edit prompt in CLIP embedding space [12]. A positive value indicates that the edit moves in the correct semantic direction.
- **CLIP Directional Cosine** ($\text{CLIP}_{\text{dir-cos}}$): The cosine similarity variant of the directional metric, providing a normalized measure of edit alignment.
- **CLIP Directional Avg** ($\text{CLIP}_{\text{dir-avg}}$): First averages the per-view image direction across all rendered views, then computes the dot product with the text direction. This reduces noise from individual views.
- **CLIP Directional Avg-Cosine** ($\text{CLIP}_{\text{dir-avg-cos}}$): The cosine similarity variant of $\text{CLIP}_{\text{dir-avg}}$, providing a normalized measure of the averaged edit direction alignment.
- **GPT-4V**: VLM-based metric to evaluate the quality of the generated shapes from the renderings. We measure quality in terms of: Prompt Alignment, 3D Plausibility, Identity Preservation, Visual Quality, 3D Consistency, Completeness, and Overall Quality.

4.2 Comparison with Baselines

Quantitative Results. Left part of Table 1 reports CLIP-based results averaged across all 25 benchmark cases. Our method achieves the highest CLIP directional similarity (3.16), $\text{CLIP}_{\text{dir-cos}}$ (11.84), and $\text{CLIP}_{\text{dir-avg-cos}}$ (15.98), indicating that our edits most faithfully follow the intended semantic direction. Vox-E is the second best on directional metrics but substantially lower, suggesting that while its outputs match the target text, the edits do not consistently move in the correct semantic direction. MVEdit and PrEditor3D achieve lower scores across all directional metrics, as they apply relatively conservative edits.

Right part of Table 1 reports GPT-4V evaluation results. Our method achieves the highest scores across all seven dimensions, with particularly strong margins on Prompt Alignment (8.60 vs. 5.36 for the next best) and Completeness (8.46 vs. 7.48). MVEdit achieves the second-highest Visual Quality (6.56) and 3D Consistency (7.16), reflecting its conservative editing strategy that preserves appearance but limits edit expressiveness. Vox-E scores the lowest across most metrics, particularly on Visual Quality (3.92) and 3D Consistency (5.00), due

Table 1: Quantitative comparison averaged over all evaluation cases. Best results are in **bold**. Metrics are scaled by ×100. GPT-4V evaluation averaged over all evaluation cases. Each method is scored out of 10, for the given metric. Prompt Alignment (Pr. Algn.), 3D Plausibility (3D Pl.), Identity Preservation (Ident.), Visual Quality (Vis. Q), 3D Consistency (3D Con.), Completeness (Compl.), and Overall.

	CLIP				GPT-4V						
Method	Dir↑	Dir-cos↑	Dir-avg↑	Dir-avg-cos↑	Pr. Algn.↑	3D Pl.↑	Ident.↑	Vis. Q.↑	3D Con.↑	Compl.↑	Overall↑
MVEdit [6]	1.77	6.57	1.77	8.42	5.36	6.84	5.60	6.56	7.16	7.48	6.32
Vox-E [39]	2.41	6.91	2.41	8.30	5.32	4.56	5.12	3.92	5.00	5.54	4.40
PrEditor3D [9]	1.26	5.05	1.26	7.26	4.84	5.68	5.96	5.29	6.04	6.83	5.36
Steer3D [29]	0.55	2.37	0.55	2.86	3.80	4.64	4.12	5.08	4.20	5.12	4.32
Edit-TRELLIS [25, 49]	**3.32**	11.17	**3.32**	14.45	7.08	6.76	6.04	6.00	6.24	6.96	6.64
GaussCtrl [48]	0.27	1.27	0.27	1.87	2.68	3.44	2.76	4.68	3.00	3.80	3.24
GaussianEditor [8]	−0.06	−0.27	−0.06	−0.33	2.44	2.92	1.84	3.64	2.24	3.08	2.52
Instruct-NeRF2NeRF [17]	2.05	6.44	2.05	7.56	4.36	4.40	4.48	3.92	4.52	4.72	3.92
Ours	3.16	**11.84**	3.16	**15.98**	**8.60**	**7.28**	**7.67**	**6.62**	**7.54**	**8.46**	**7.40**

Table 2: User study results (30 participants). Each cell shows the percentage of times our method was preferred over the baseline. Higher is better.

vs Baseline	Prompt Algn.↑	Vis. Qual.↑	Shape Pres.↑
MVEdit [6]	80.6%	75.5%	77.0%
PrEditor3D [9]	88.8%	82.4%	80.7%
Vox-E [39]	89.9%	94.5%	90.3%

Table 3: Compute cost on Consumer GPU RTX 3090, 4-view 256 × 256. Inference: 22.20 s model loading + 2.43 s warm-up. T-: training.

	Training				Inference
	T-Load	T-TI	T-DB	T-Total	
Time	22.95 s	66.56 s	126.47 s	3.60 min	24.63 s
VRAM	5.97 GB	12.86 GB	12.86 GB	12.86 GB	8.23 GB

to the artifacts introduced by its voxel-based optimization. PrEditor3D achieves the highest Identity Preservation (5.96) among baselines, but its overall quality remains lower than ours. Our personalization approach enables more faithful, higher-quality edits while preserving object identity.

User Study. We conduct a user study to complement our automatic metrics. Our benchmark covers 15 objects across 25 editing cases, each paired with 3 baselines, yielding 75 pairwise comparisons in total. For each comparison, participants are asked three questions: prompt alignment, visual quality, and shape preservation. Each participant is randomly assigned 20 out of the 75 comparisons. We collected responses from 30 participants, resulting in 600 pairwise judgments. As shown in Table 2, our method is consistently preferred over all baselines across all three dimensions. The margin is largest against Vox-E (89.9–94.5%), whose voxel-based optimization frequently introduces visual artifacts. Against MVEdit, the preference is smallest on visual quality (75.5%), consistent with its conservative editing strategy that preserves appearance at the cost of limited edit expressiveness. Against PrEditor3D, our method is strongly preferred on prompt alignment (88.8%), reflecting PrEditor3D's difficulty in achieving the intended semantic changes.

Qualitative Results. Figure 3 presents qualitative comparisons across representative benchmark cases. In the "van convertible" example, only our method suc-

Table 4: Ablation study on the *Robot Sitting* case. Best results per group are in **bold**. ↑: higher is better. CLIP metrics are scaled by ×100.

	CLIP				GPT-4V						
Configuration	Dir↑	Dir-cos↑	Dir-avg↑	Dir-avg-cos↑	Pr. Align.↑	3D Pl.↑	Ident.↑	Vis. Q.↑	3D Con.↑	Compl.↑	Overall↑
Multi-view joint training											
Front view only	−0.39	−2.76	−0.39	−3.92	3	4	8	5	4	6	5
Back view only	−0.49	−2.88	−0.49	−4.09	2	3	6	3	4	5	3
Side view only	−0.39	−1.39	−0.39	−1.81	2	3	5	3	4	4	3
4 orthogonal views (Ours)	**0.51**	**3.15**	**0.51**	**4.79**	**8**	**7**	**9**	**6**	**8**	**9**	**7**
Masked losses											
w/o cross attention loss	0.11	0.71	0.11	0.73	3	5	6	4	5	6	5
w/o masked diffusion loss	−0.27	−2.43	−0.27	−3.38	2	3	7	4	5	6	4
w/o both mask loss	−0.41	−2.15	−0.41	−2.72	3	5	7	4	6	7	5
Full (Ours)	**0.51**	**3.15**	**0.51**	**4.79**	**8**	**7**	**9**	**6**	**7**	**8**	**7**
Two-phase optimization											
w/o DB (TI only, Phase 1)	**4.80**	**15.77**	**4.80**	**17.77**	2	3	1	3	4	5	3
w/o TI (DB only, Phase 2)	0.40	2.53	0.40	3.80	**9**	**7**	8	6	7	**8**	7
TI + DB (Ours)	0.51	3.15	0.51	4.79	**9**	**7**	**9**	**7**	**8**	8	**8**

cessfully opens the roof, whereas all three baselines leave the van unchanged. For "two eagles together," MVEdit and Vox-E fail to generate a second eagle, while PrEditor3D produces two instances but exhibits noticeable color drift. In "cake in a plate," MVEdit and Vox-E omit the plate entirely, and although PrEditor3D introduces one, it appears blurry. In the "sofa single seat" case, only our method faithfully follows the prompt, correctly transforming the sofa into a single-seat design. For "person wearing sunglasses," MVEdit and Vox-E generate unrealistic outputs, while PrEditor3D does produce sunglasses but with disconnected temples at the eye corners, resulting in an unnatural appearance. Across all scenarios, our method demonstrates superior adherence to the editing prompts while consistently preserving object identity.

To further highlight the versatility of our approach, we showcase a diverse set of editing tasks. These include attribute transfer ("dog as a cat," "dog as a pig"), where species are altered while maintaining the original pose; object addition ("basket with apples"), where new elements are coherently integrated; pose modification ("robot sitting," "lady sitting," "lady riding a horse"), where body configurations are adjusted; appearance editing ("dog smiling," "shoes in red"), targeting fine-grained attributes; style transfer ("koala in lego style"), altering material appearance; and structural transformations ("boat to houseboat," "boat with sails," "lady with child"), involving substantial geometric changes. In all cases, the learned token embedding effectively preserves the core identity of the input object while enabling precise and flexible modifications.

4.3 Ablation Studies

We conduct a comprehensive ablation study to assess the contribution of each component. All experiments are performed on the *Robot Sitting* case ("a photo of a robot" → "a photo of a robot sitting"). Table 4 presents the quantitative results, while Figure 6 provides corresponding visual comparisons.

Fig. 3: Qualitative comparison for benchmark.

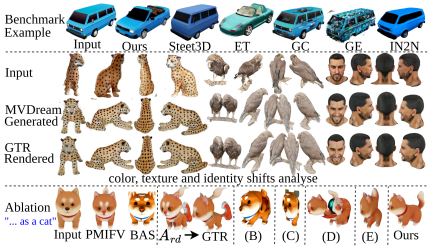

Fig. 4: Visualizations for benchmark, color drift, and backbone replacement.

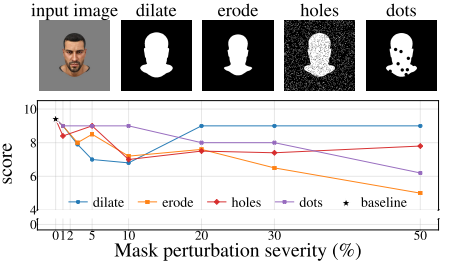

Fig. 5: Robustness analysis.

Ablation (a): Multi-view training. As shown in Fig. 6, when multi-view training is removed and only a single view is used, the edits become inconsistent across views due to the Janus problem. Specifically, when only the front view is used for fine-tuning, the white dots (eyes) from the front appear on all four generated views. Conversely, when only the back view is used, the generated views lose the white dots entirely. When only a single side view is used, every generated view exhibits arms, replicating the side-view features across all viewpoints. This demonstrates that single-view fine-tuning, regardless of which view is chosen, introduces the Janus problem by propagating view-specific features to all viewpoints. We also tested jointly fine-tuning with 2 and 3 views: as more views are jointly used, the Janus problem diminishes, yielding increasingly consistent results across viewpoints.

Ablation (b): Masked losses. Fig. 6 shows that each masking component addresses a distinct failure mode. Without the cross-attention loss, the token embedding absorbs background statistics, causing the reconstructed robot to shift toward the gray background color. Without the masked diffusion loss, the denoising objective is no longer restricted to the foreground region, leading to identity degradation manifested as spurious artifacts (*e.g.*, a white dot on the robot's abdomen). When both losses are disabled, the two failure modes compound: the output suffers from both background color bleeding and structural artifacts, demonstrating the complementary roles of these losses.

Ablation (c): Two-phase optimization. Fig. 6 presents the two optimization phases serve complementary roles. Without textual inversion (DB only), the token embedding is not optimized to represent the input object, so the model lacks a learned concept of the robot. As a result, the generated shape deviates from the original structure , the robot's legs become human-like, as the model defaults to its generic prior for "sitting." Without DreamBooth fine-tuning (TI only), the multi-view diffusion model is not adapted to the learned token, and the strong semantic prior of "sitting" dominates: the model generates a generic sitting human, entirely discarding the robot's identity. Notably, as shown in Table 4, the TI-only configuration achieves substantially higher CLIP directional

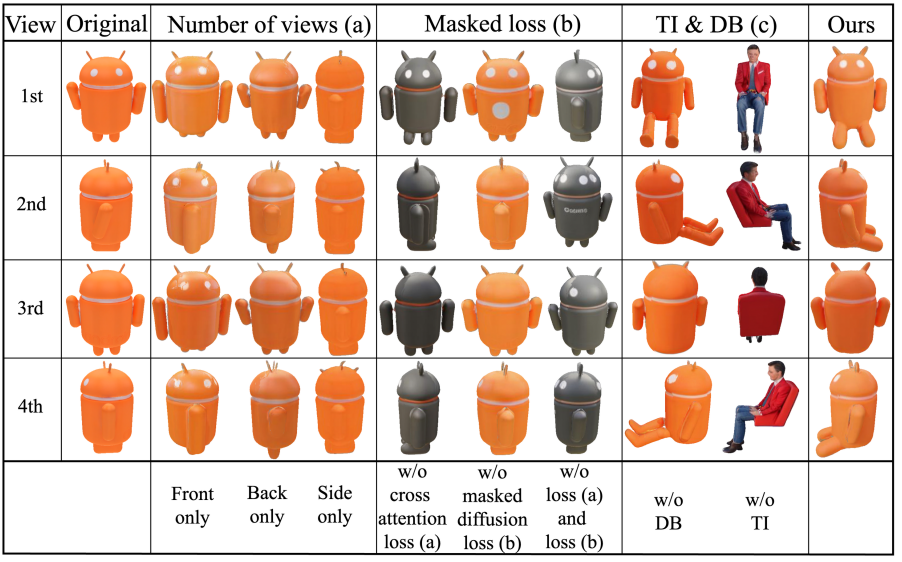

Fig. 6: Qualitative ablation study on the *Robot Sitting* case ("...robot" → "...robot sitting"). **(a)** Training with only a single view (front, back, or side) reduces 3D consistency. **(b)** Removing mask-based losses degrades edit localization. **(c)** Without TI, identity is partially lost; TI-only (no DreamBooth) fails to preserve the object.

scores (CLIP_{dir}: 4.80 vs. 0.51 for ours). However, this is misleading: without DreamBooth, the model produces a human in a canonical sitting pose, which strongly aligns with the "sitting" keyword in CLIP space despite completely failing to preserve the input object. The GPT-4V evaluation reveals this discrepancy TI only scores the lowest on Identity Preservation (1/10) and Prompt Alignment (2/10), confirming that high CLIP directional scores do not necessarily reflect faithful editing when object identity is lost. Our combined two-phase pipeline achieves the best balance: token optimisation stage anchors the token to the input object's appearance, while multiview diffusion model fine-tuning stage adapts the generative model to faithfully edit the learned concept without sacrificing identity. These results demonstrate that all components are essential for achieving our final performance.

Ablation (d): Backbone Replacement. We feed our Personalized MVDream Inferred Front View (PMIFV) to $\text{MVAdapter}_{\text{sdxl}}$ and $\text{MVAdapter}_{\text{realdream}}$ for six-view generation, then GTR, but yield inconsistent, blurry views that hurt reconstruction. For BAS+SEVA and BAS + Era3D, Break-A-Scene is fine-tuned with the mask, sampled three times to generate a single front view image, and the best result (by CLIP score) is fed to SEVA or Era3D, showing that Break-A-Scene itself is the bottleneck for personalized generation. Finally, PMIFV+SEVA and PMIFV+Era3D reconstruct directly from PMIFV, but since this falls outside their training distribution, the resulting views are 3D-inconsistent. Table 5

Table 5: Ablation study replacing different backbones. Best results per group are in **bold**. ↑: higher is better. CLIP metrics are scaled by ×100.

	CLIP				GPT-4V						
Configuration	Dir↑	Dir-cos↑	Dir-avg↑	Dir-avg-cos↑	Pr. Algn.↑	3D Pl.↑	Ident.↑	Vis. Q.↑	3D Con.↑	Compl.↑	Overall↑
PMIFV+MVAdapter$_{sdxl}$	1.93	6.14	1.93	7.83	5.32	3.60	4.20	3.28	4.12	4.80	3.88
PMIFV+MVAdapter$_{realdream}$	2.12	6.83	2.12	8.81	5.64	3.84	4.52	3.64	4.32	4.96	4.04
BAS+SEVA	2.41	6.64	2.41	8.91	5.52	4.04	4.00	4.40	4.04	4.88	4.36
BAS+Era3D	2.58	7.57	2.58	10.32	5.40	3.96	4.04	4.16	4.32	4.74	4.20
PMIFV+SEVA	2.42	7.72	2.42	10.61	6.64	4.72	5.36	4.36	4.80	5.64	4.84
PMIFV+Era3D	2.15	6.81	2.15	9.19	5.92	4.32	4.92	4.20	4.88	5.48	4.56
Ours	**3.16**	**11.84**	**3.16**	**15.98**	**8.60**	**7.28**	**7.67**	**6.62**	**7.54**	**8.46**	**7.40**

shows a quantitative comparison. And the lower part of Figure 4 shows visual comparisons, where A$_{rd}$: PMIFV+MV-Adapter$_{\text{realdream}}$; B: BAS+SEVA; C: BAS+Era3D; D: PMIFV+SEVA; E: PMIFV+Era3D.

Robustness analysis. As shown in Figure 5, we evaluate robustness to mask perturbations on the "smile with teeth" editing case, running 10 inferences per perturbation level and scoring each with GPT-4V across six dimensions, namely edit success, identity preservation, mask leak-freeness, structural integrity, view consistency, and robustness, each scored $0-10$. Small dilations are absorbed into the learned shape with only a moderate drop in score, and beyond a 10% perturbation, the initial token ("character") recovers the ground-truth boundary. Other perturbation types degrade gracefully with increasing severity, indicating that our method is robust to imperfect masks. Table 3 shows the runtime cost.

Limitations. Our approach has several limitations. **(i) Limited resolution.** MVDream operates at a fixed resolution of 256×256, which limits the quality of the generated multi-view images and consequently affects the fidelity of the reconstructed mesh. **(ii) Complex scenes.** Our method struggles with highly complex scenes; editing within large-scale environments with many interacting objects remains challenging, though we believe this can be addressed through hierarchical decomposition in future work. **(iii) Fixed view configuration.** Our method relies on four orthogonal views at a single elevation, the default configuration of MVDream. Since GTR benefits from a larger number of input views at varying elevations, this constraint limits reconstruction quality and could be alleviated by adopting a multi-view backbone that supports more flexible camera configurations. **(iv) Multi-concept part editing.** Break-A-Scene's disentanglement works best with up to four concepts and requires spatial separation between them; decomposing an object into "head, upper body, arms, legs, and feet" violates both conditions. We also observed mask leakage: even with two spatially adjacent parts (e.g., head and body), semantic information leaks across boundaries despite the use of clean masks. Furthermore, four-view occlusion causes per-region masks for fine-grained components (e.g., a dog's left ear) to be incomplete due to self-occlusion, leaving some regions unsupervised. **(v) Color, texture, and identity shifts.** As shown in the middle part of Figure 4, fine-tuning cannot reach an exact local minimum without overfitting

and forgetting the MVDream prior, although with grained semantic masks (e.g., head, body). Combined with stochastic inference (seeds, prompt) causes mild drift from the source identity. GTR provide small reconstruction errors in cross-view regions (e.g., cat spine).

5 Conclusion

We presented DreamEdit3D, a disentangled personalization framework for text-guided 3D editing that learns object-specific token embeddings within a multi-view diffusion model. Our two-phase optimization textual inversion followed by joint multi-view UNet fine-tuning enables flexible, language-driven edits while preserving object identity. Quantitative results on 25 diverse editing cases show that our method achieves the best CLIP directional similarity and GPT-4V scores compared to recent methods such as MVEdit, Vox-E, PrEditor3D, Steer3D, GaussCtrl, GaussianEditor Instruct-NeRF2NeRF and Edit-TRELLIS, indicating stronger semantic alignment. A user study with 30 participants further confirms consistent preference for our method in prompt alignment, visual quality, and shape preservation. Ablation studies highlight the contribution of each component: joint multi-view training improves 3D consistency, masked losses reduce background artifacts, and the two-phase design balances edit fidelity and identity preservation.

Future Work. Several promising directions remain. **(i) Improved reconstruction.** Incorporating 3D-aware attention into the token learning phase, along with stronger reconstruction backbones, could improve final mesh quality. **(ii) Higher-resolution, more diverse views.** Fine-tuning MVDream for super-resolution output, or integrating novel-view synthesis models such as Zero-1-to-3 [24], would increase the resolution and diversity of generated views. **(iii) Multi-concept editing.** Following PartCraft [32] and Break-A-Scene [1], recombining disentangled tokens across or within objects could enable creative composition, while attribute-level control would allow finer colour and texture specification. **(iv) Scene-level editing.** A zoom-in strategy focusing on local regions could extend our framework to large-scale scenes. **(v) Alternative backbones.** Extend MVAdapter with token optimisation and single-image fine-tuning to generate more, higher-resolution, consistent views; SEVA and Era3D could be adapted for token-based conditioning.

Acknowledgements

This work was supported by the ERC Consolidator Grant Gen3D (101171131) awarded to Matthias Nießner. We gratefully thank Ziya Erkoç for his supervision, guidance, and insightful feedback throughout this project, and Prof. Dr. Matthias Nießner for his support and for providing an excellent research environment at the Visual Computing Lab. We thank Prof. Dr. Angela Dai for the video voice-over, and the Leibniz-Rechenzentrum (LRZ) for providing the computational resources used in this work.

References

1. Avrahami, O., Aberman, K., Fried, O., Cohen-Or, D., Lischinski, D.: Break-a-scene: Extracting multiple concepts from a single image. In: SIGGRAPH Asia 2023 Conference Papers. SA '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3610548.3618154, https://doi.org/10.1145/3610548.3618154
2. Baldridge, J., Bauer, J., Bhutani, M., Brichtova, N., Bunner, A., Castrejon, L., Chan, K., Chen, Y., Dieleman, S., Du, Y., et al.: Imagen 3. arXiv preprint arXiv:2408.07009 (2024)
3. Barda, A., Gadelha, M., Kim, V.G., Aigerman, N., Bermano, A.H., Groueix, T.: Instant3dit: Multiview inpainting for fast editing of 3d objects (2024), https://arxiv.org/abs/2412.00518
4. Betker, J., Goh, G., Jing, L., Brooks, T., Wang, J., Li, L., Ouyang, L., Zhuang, J., Lee, J., Guo, Y., Manassra, W., Dhariwal, P., Chu, C., Jiao, Y., Ramesh, A.: Improving image generation with better captions (2023), https://cdn.openai.com/papers/dall-e-3.pdf, technical report, OpenAI
5. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L.J., Tremblay, J., Khamis, S., et al.: Efficient geometry-aware 3d generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16123–16133 (2022)
6. Chen, H., Shi, R., Liu, Y., Shen, B., Gu, J., Wetzstein, G., Su, H., Guibas, L.: Generic 3d diffusion adapter using controlled multi-view editing (2024)
7. Chen, M., Shapovalov, R., Laina, I., Monnier, T., Wang, J., Novotny, D., Vedaldi, A.: Partgen: Part-level 3d generation and reconstruction with multi-view diffusion models (2024), https://arxiv.org/abs/2412.18608
8. Chen, Y., Chen, Z., Zhang, C., Wang, F., Yang, X., Wang, Y., Cai, Z., Yang, L., Liu, H., Lin, G.: Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21476–21485 (2024)
9. Erkoç, Z., Gümeli, C., Wang, C., Nießner, M., Dai, A., Wonka, P., Lee, H.Y., Zhuang, P.: Preditor3d: Fast and precise 3d shape editing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 640–649 (2025)
10. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., Podell, D., Dockhorn, T., English, Z., Lacey, K., Goodwin, A., Marek, Y., Rombach, R.: Scaling rectified flow transformers for high-resolution image synthesis (2024), https://arxiv.org/abs/2403.03206
11. Gal, R., Alaluf, Y., Atzmon, Y., Patashnik, O., Bermano, A.H., Chechik, G., Cohen-Or, D.: An image is worth one word: Personalizing text-to-image generation using textual inversion. arXiv preprint arXiv:2208.01618 (2022)
12. Gal, R., Patashnik, O., Maron, H., Bermano, A.H., Chechik, G., Cohen-Or, D.: StyleGAN-NADA: CLIP-guided domain adaptation of image generators. ACM Transactions on Graphics (TOG) **41**(4), 1–13 (2022)
13. Gao, R., Holynski, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: Cat3d: Create anything in 3d with multi-view diffusion models (2024), https://arxiv.org/abs/2405.10314
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)

15. Guo, Z., Wu, Y., Chen, Z., Chen, L., Zhang, P., He, Q.: Pulid: Pure and lightning id customization via contrastive alignment (2024), https://arxiv.org/abs/2404.16022
16. Haque, A., Tancik, M., Efros, A.A., Holynski, A., Kanazawa, A.: Instruct-nerf2nerf: Editing 3d scenes with instructions (2023), https://arxiv.org/abs/2303.12789
17. Haque, A., Tancik, M., Efros, A.A., Holynski, A., Kanazawa, A.: Instruct-nerf2nerf: Editing 3d scenes with instructions. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 19740–19750 (2023)
18. Hertz, A., Mokady, R., Tenenbaum, J., Aberman, K., Pritch, Y., Cohen-Or, D.: Prompt-to-prompt image editing with cross attention control. arXiv preprint arXiv:2208.01626 (2022)
19. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
20. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: Lrm: Large reconstruction model for single image to 3d. arXiv preprint arXiv:2311.04400 (2023)
21. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4401–4410 (2019)
22. Kumari, N., Zhang, B., Zhang, R., Shechtman, E., Zhu, J.Y.: Multi-concept customization of text-to-image diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
23. Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 300–309 (2023)
24. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9298–9309 (2023)
25. Liu, S., Han, Y., Xing, P., Yin, F., Wang, R., Cheng, W., Liao, J., Wang, Y., Fu, H., Han, C., et al.: Step1x-edit: A practical framework for general image editing. arXiv preprint arXiv:2504.17761 (2025)
26. Liu, Y., Lin, C., Zeng, Z., Long, X., Liu, L., Komura, T., Wang, W.: Syncdreamer: Generating multiview-consistent images from a single-view image. arXiv preprint arXiv:2309.03453 (2023)
27. Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., Wang, W.: Wonder3d: Single image to 3d using cross-domain diffusion (2023), https://arxiv.org/abs/2310.15008
28. Lugmayr, A., Danelljan, M., Romero, A., Yu, F., Timofte, R., Van Gool, L.: RePaint: Inpainting using denoising diffusion probabilistic models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11461–11471 (2022)
29. Ma, Z., Chen, H., Yue, Y., Gkioxari, G.: Feedforward 3d editing via text-steerable image-to-3d. arXiv preprint arXiv:2512.13678 (2025)
30. Michel, O., Bar-On, R., Liu, R., Benaim, S., Hanocka, R.: Text2mesh: Text-driven neural stylization for meshes (2021), https://arxiv.org/abs/2112.03221
31. Mikaeili, A., Perel, O., Safaee, M., Cohen-Or, D., Mahdavi-Amiri, A.: Sked: Sketch-guided text-based 3d editing (2023), https://arxiv.org/abs/2303.10735
32. Ng, K.W., Zhu, X., Song, Y.Z., Xiang, T.: Partcraft: Crafting creative objects by parts (2024), https://arxiv.org/abs/2407.04604

33. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
34. Raj, A., Kaza, S., Poole, B., Niemeyer, M., Ruiz, N., Mildenhall, B., Zada, S., Aberman, K., Rubinstein, M., Barron, J., et al.: Dreambooth3d: Subject-driven text-to-3d generation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 2349–2359 (2023)
35. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with CLIP latents. In: arXiv preprint arXiv:2204.06125 (2022)
36. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10684–10695 (2022)
37. Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K.: Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 22500–22510 (2023)
38. Ruiz, N., Li, Y., Jampani, V., Wei, W., Hou, T., Pritch, Y., Wadhwa, N., Rubinstein, M., Aberman, K.: Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models (2024), https://arxiv.org/abs/2307.06949
39. Sella, E., Fiebelman, G., Hedman, P., Averbuch-Elor, H.: Vox-e: Text-guided voxel editing of 3d objects. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 430–440 (2023)
40. Shah, V., Ruiz, N., Cole, F., Lu, E., Lazebnik, S., Li, Y., Jampani, V.: Ziplora: Any subject in any style by effectively merging loras (2026), https://arxiv.org/abs/2311.13600
41. Shen, T., Gao, J., Yin, K., Liu, M.Y., Fidler, S.: Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. Advances in Neural Information Processing Systems **34**, 6087–6101 (2021)
42. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. arXiv preprint arXiv:2308.16512 (2023)
43. Voynov, A., Chu, Q., Cohen-Or, D., Aberman, K.: p+: Extended textual conditioning in text-to-image generation. arXiv preprint arXiv:2303.09522 (2023)
44. Wang, J., Fang, J., Zhang, X., Xie, L., Tian, Q.: Gaussianeditor: Editing 3d gaussians delicately with text instructions (2024), https://arxiv.org/abs/2311.16037
45. Wang, P., Shi, Y.: Imagedream: Image-prompt multi-view diffusion for 3d generation (2023), https://arxiv.org/abs/2312.02201
46. Wang, Q., Bai, X., Wang, H., Qin, Z., Chen, A., Li, H., Tang, X., Hu, Y.: Instantid: Zero-shot identity-preserving generation in seconds (2024), https://arxiv.org/abs/2401.07519
47. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. Advances in neural information processing systems **36**, 8406–8441 (2023)
48. Wu, J., Bian, J.W., Li, X., Wang, G., Reid, I., Torr, P., Prisacariu, V.A.: Gaussctrl: Multi-view consistent text-driven 3d gaussian splatting editing. In: European conference on computer vision. pp. 55–71. Springer (2024)
49. Xiang, J., Lv, Z., Xu, S., Deng, Y., Wang, R., Zhang, B., Chen, D., Tong, X., Yang, J.: Structured 3d latents for scalable and versatile 3d generation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21469–21480 (2025)

50. Yang, Y., Long, X.X., Dou, Z., Lin, C., Liu, Y., Yan, Q., Ma, Y., Wang, H., Wu, Z., Yin, W.: Wonder3d++: Cross-domain diffusion for high-fidelity 3d generation from a single image. IEEE Transactions on Pattern Analysis and Machine Intelligence **48**(2), 1674–1688 (Feb 2026). https://doi.org/10.1109/tpami.2025.3618675, http://dx.doi.org/10.1109/TPAMI.2025.3618675
51. Ye, H., Zhang, J., Liu, S., Han, X., Yang, W.: Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. arXiv preprint arXiv:2308.06721 (2023)
52. Ye, J., Xie, S., Zhao, R., Wang, Z., Yan, H., Zu, W., Ma, L., Zhu, J.: Nano3d: A training-free approach for efficient 3d editing without masks. arXiv preprint arXiv:2510.15019 (2025)
53. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3836–3847 (2023)
54. Zheng, Y., Huang, M., Chen, N., Mao, Z.: Pro3d-editor : A progressive-views perspective for consistent and precise 3d editing (2025), https://arxiv.org/abs/2506.00512
55. Zhuang, P., Han, S., Wang, C., Siarohin, A., Zou, J., Vasilkovsky, M., Shakhrai, V., Korolev, S., Tulyakov, S., Lee, H.Y.: Gtr: Improving large 3d reconstruction models through geometry and texture refinement. arXiv preprint arXiv:2406.05649 (2024)