Sherpa3D: Boosting High-Fidelity Text-to-3D Generation via Coarse 3D Prior

Fangfu Liu[1], Diankun Wu[1], Yi Wei[1], Yongming Rao[2], Yueqi Duan[1†]
[1]Tsinghua University, [2]BAAI

Figure 1. **Gallery of Sherpa3D:** Blender rendering for various textured meshes from *Sherpa3D*, which is able to generate high-fidelity, diverse, and multi-view consistent 3D contents with input text prompts. Our method is also compatible with popular graphics engines.

Abstract

Recently, 3D content creation from text prompts has demonstrated remarkable progress by utilizing 2D and 3D diffusion models. While 3D diffusion models ensure great multi-view consistency, their ability to generate high-quality and diverse 3D assets is hindered by the limited 3D data. In contrast, 2D diffusion models find a distillation approach that achieves excellent generalization and rich details without any 3D data. However, 2D lifting methods suffer from inherent view-agnostic ambiguity thereby leading to serious multi-face Janus issues, where text prompts fail to provide sufficient guidance to learn coherent 3D results. Instead of retraining a costly viewpoint-aware model, we study how to fully exploit easily accessible coarse 3D knowledge to enhance the prompts and guide 2D lifting optimization for refinement. In this paper, we propose ***Sherpa3D****, a new text-to-3D framework that achieves high-fidelity, generalizability, and geometric*

†Corresponding author.

consistency simultaneously. Specifically, we design a pair of guiding strategies derived from the coarse 3D prior generated by the 3D diffusion model: a structural guidance for geometric fidelity and a semantic guidance for 3D coherence. Employing the two types of guidance, the 2D diffusion model enriches the 3D content with diversified and high-quality results. Extensive experiments show the superiority of our Sherpa3D over the state-of-the-art text-to-3D methods in terms of quality and 3D consistency. Project page: https://liuff19.github.io/Sherpa3D/.

1. Introduction

3D content generation [36, 39, 58, 81] finds a broad range of applications, including games, movies, virtual/augmented reality and robots. However, the conventional process of creating premium 3D assets is still expensive and challenging as it requires multiple labor-intensive and time-consuming stages [33]. Fortunately, this challenge has prompted the development of recent text-to-3D methods [10, 26, 28, 37, 39, 49, 52, 58, 79]. Only using text prompts to automate 3D generation, these techniques pave a promising way towards streamlining 3D creation.

Powered by the great breakthroughs in diffusion models [54, 62, 63, 86], two research lines of rationalization have recently emerged in text-to-3D: inference-only 3D diffusion methods and optimization-based 2D lifting methods. Specifically, the inference-only methods [20, 29, 54] seek to directly generate 3D-consistent assets by extensively training a new diffusion model on 3D data. However, due to the scarcity of 3D datasets compared to accessible 2D images or text data, these 3D diffusion models suffer from low quality and limited generalizability. Without requiring any 3D data for training, 2D lifting methods [4, 10, 39, 49, 58, 77, 79] can produce high-quality and diversified 3D results by distilling 3D knowledge from pre-trained 2D diffusion models [62, 63, 86], also known as Score Distillation Sampling (SDS). Yet lifting 2D observations into 3D is inherently ambiguous without sufficient 3D guidance from text prompts, leading to notorious multi-view inconsistency (*e.g.*, Janus problems) in 2D lifting methods.

These findings motivate us to think: *is it possible to bridge the two aforementioned streams to achieve generalizability, high-fidelity, and geometric consistency simultaneously?* An intuitive idea is to leverage more 3D data [11, 12] to fine-tune a view-point aware diffusion model, but it requires substantial computational resources and is prone to overfitting due to data bias [42, 69]. In contrast, our key insight is to utilize the easily accessible 3D diffusion model as guidance and study how to fully exploit coarse 3D knowledge to guide 2D lifting optimization for refinement. In particular, when maintaining the quality and generalizability of the original 2D diffusion model, we hope the 2D lifting awareness can be guided by the strong 3D geometric information from the 3D diffusion model. However, it is non-trivial in pursuit of this balance. Relying too heavily on the coarse 3D priors from the 3D diffusion model may degrade the generation quality, whereas little 3D guidance could result in a lack of geometric awareness, leading to multi-view inconsistency.

Towards this end, we propose **Sherpa3D** in this paper, which greatly boosts high-fidelity and highly diversified text-to-3D generation with geometric consistency. Our method begins by employing a 3D diffusion model to craft a basic 3D guide with limited details. Building upon the coarse 3D prior, we introduce two guiding strategies to inform 2D diffusion model throughout lifting optimization: a structural guide for geometric fidelity and a semantic guide for 3D coherence. Specifically, the structural guide leverages the first-order gradient information of the normals from the 3D prior to supervise the optimization of the structure. These normals are then integrated into the input of a pre-trained 2D diffusion model, refining the geometric details. Concurrently, our semantic guide extracts high-level features from multi-views of the 3D prior. These features guide the 2D lifting optimization to perceive the geometric consistency under the preservation of original generalizability and quality. Furthermore, we design an annealing function, which modulates the influence of the 3D guidance to better preserve the capabilities of 2D and 3D diffusion models. As a result, our Sherpa3D is aware of the geometric consistency with rich details and generalizes well across diverse text prompts. Extensive experiments verify the efficacy of our framework and show that our Sherpa3D outperforms existing methods for high-fidelity and geometric consistency (see qualitative results gallery in Figure 1 and quantitative results in Table 2).

2. Related Work

2.1. Text-to-image Generation

Recently, text-to-image models such as unCLIP [61], Imagen [63], and Stable Diffusion [62] have shown remarkable capability of generating high-quality and creative images given text prompts. Such significant progress is powered by advances in diffusion models [13, 25, 55, 72], which can be pre-trained on billions of image-text pairs [64, 66] and understands general objects with complex semantic concepts (nouns, artistic styles, *etc.*) [62]. Despite the great success of photorealistic and diversified image generation, using language to generate different viewpoints of the same object with 3D coherence remains a challenging problem [80].

2.2. Text-to-3D Generation

Building on promising text-to-image diffusion models, there has been a surge of studies in text-to-3D generation.

However, it is non-trivial due to the scarcity of diverse 3D data [8, 12, 82] compared to 2D. Existing 3D native diffusion models [20, 29, 45, 54, 85, 88] usually work on a limited object category and struggle with generating in-the-wild 3D assets. To achieve generalizable 3D generation, pioneering works DreamFusion [58] and SJC [77] propose to distill the score of image distribution from pre-trained 2D diffusion models [62, 63] and show impressive results. Following works [10, 27, 38, 39, 49, 75, 76, 79, 84, 90] continue to enhance various aspects such as generation fidelity and optimization stability or explore more application scenarios [60, 70, 91]. As it is inherently ambiguous to lift 2D observations into 3D, they may suffer from multi-face issues. Although some methods use prompt engineering [4] or train a costly viewpoint-aware model [42, 69] to alleviate such problems, they fail to generate high-quality results [10] or easily overfit to domain-specific data [12, 69]. In this work, we bridge the gap between 3D and 2D diffusion models through meticulously designed 3D guidance, which leads the 2D lifting process to achieve high-fidelity, diversified, and coherent 3D generation.

2.3. 3D Generative Models

Extensive research has been conducted in the field of 3D generative modeling, exploring diverse 3D representations like 3D voxel grids [15, 22, 46], point clouds [3, 47, 51], and meshes [16, 87]. The majority of these approaches rely on training data presented in the form of 3D assets, which proves challenging to obtain at a large scale. Drawing inspiration from the success of neural volume rendering, recent studies have shifted towards investing in 3D-aware image synthesis [6, 7, 18, 21, 56, 65]. This approach offers the advantage of directly learning 3D generative models from images. However, volume rendering networks typically exhibit slow querying speeds, resulting in a trade-off between extended training times and a lack of multi-view consistency. Recently, benefitted from 2D diffusion models, some works generate multi-view images with single-view input [41–44, 68, 83]. As one of the pioneering works, Zero-1-to-3 [42] uses a synthetic dataset to fine-tune the pretrained diffusion models, aiming to learn controls of the relative camera viewpoint. Beyond Zero-1-to-3, SyncDreamer [43] employs a synchronized multiview diffusion model to capture the joint probability distribution of multiview images. This model facilitates the generation of multiview-consistent images through a unified reverse process. Different from these methods, we focus on text-to-3D synthesis, with the goal of generating multi-view consistent 3D contents with text prompts.

3. Method

Given a text prompt, our goal is to generate 3D assets with high quality, generalizability, and multi-view consistency. Our framework can be divided into three stages: (1) build coarse 3D prior from the 3D diffusion model (Sec. 3.2); (2) formulate two guiding strategies (*e.g.*, structural and semantic guidance) for 2D lifting process (Sec. 3.3); (3) incorporate both 3D guidance and SDS loss with an annealing technique in optimization and generate the final 3D object (Sec. 3.4). In this way, we can leverage the full power of state-of-the-art 3D and 2D diffusion models to obtain 3D coherence as 3D models, retaining intricate details and creative freedom as 2D models. Our pipeline is depicted in Figure 2. Before introducing our Sherpa3D in detail, we first review the theory of Score Distillation Sampling (SDS).

3.1. Preliminaries

Score Distillation Sampling (SDS). As one of the most representative 2D lifting methods, Dreamfusion [58] first presents the concept of Score Distillation Sampling (SDS), which is an algorithm to optimize a 3D representation such that the image rendered from any view maintains a high likelihood as evaluated by the 2D diffusion model given text prompts. SDS consists of two key components: (1) a 3D representation with parameter θ, which can produce an image x at desired camera $\mathbf{c}$ through a parametric function $\mathbf{x} = g(\theta; \mathbf{c})$; (2) a pre-trained text-to-image 2D diffusion model ϕ with a score function $\epsilon_\phi(\mathbf{x}_t; y, t)$ that predicts the sample noise ϵ given noisy image $\mathbf{x}_t$, noise level t and text embedding y. The score function guides the direction of the gradient for updating θ to reside rendered images in high-density areas conditioned on text y. The gradient is calculated by SDS as:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, \mathbf{x}) = \mathbb{E}_{t,\epsilon}\left[w(t)\left(\epsilon_\phi\left(\mathbf{x}_t; y, t\right) - \epsilon\right)\frac{\partial \mathbf{x}}{\partial \theta}\right], \quad (1)$$

where $w(t)$ is a weighting function. In practice, the denoising score function ϵ_ϕ is often replaced with another function $\tilde{\epsilon}_\phi$ that uses classifier-free guidance [24] that controls the strength of the text condition (see Supplementary).

3.2. Sculpting a Coarse 3D Prior

To facilitate text-to-3D generation, most existing methods [27, 49, 58] rely on implicit 3D representations such as Neural Radiance Fields (NeRF) [50] and its variants [5, 53]. However, it is difficult for NeRF-based modeling to extract the high-quality surface with material and texture [78]. To address this, we adopt the hybrid scene representation of DMTet [67], including a deformable tetrahedral grid that encodes a signed distance function (SDF) and a differentiable marching tetrahedra (MT) layer that extracts explicit surface mesh. Equipped with the hybrid representation, we sculpt a coarse 3D prior from 3D diffusion model G_{3D} (*e.g.*, Shap-E [29]) by the following procedure. Given text prompts y, we first use the 3D diffusion model G_{3D} to generate 3D results M_0 and employ multi-layer perceptions

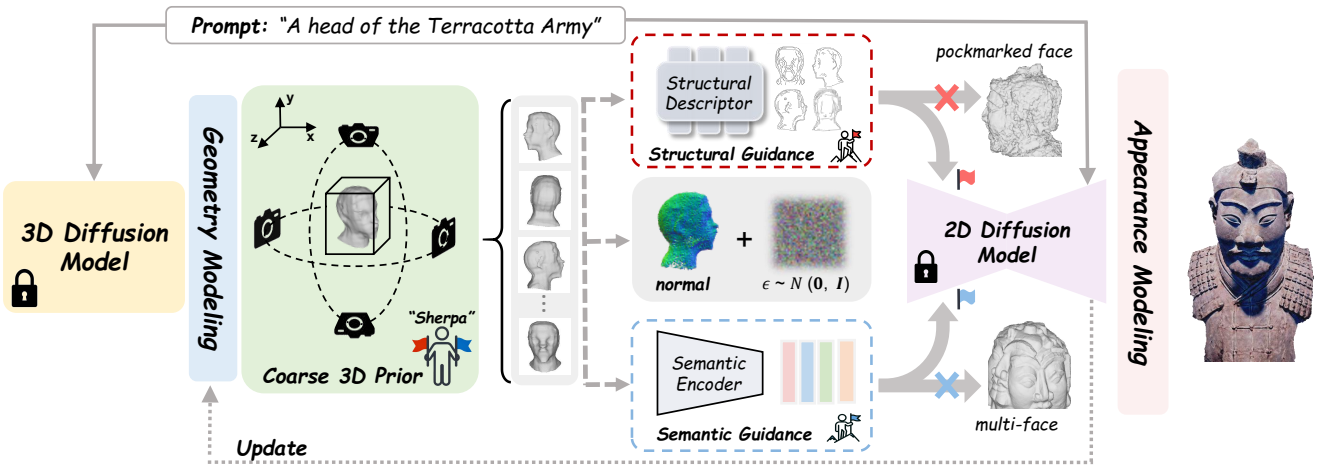


Figure 2. **Pipeline of our Sherpa3D.** Given a text as input, we first prompt 3D diffusion to build a coarse 3D prior M encoded in the geometry model (*e.g.*, DMTet). Next, we render the normal map of the extracted mesh in DMTet and derive two guiding strategies from M. (a) Structural Guidance: we utilize the structural descriptor to compute salient geometric features for preserving geometry fidelity (*e.g.*, without a pockmarked face problem). (b) Semantic Guidance: we leverage a semantic encoder (*e.g.*, CLIP) to extract high-level information for keeping 3D consistency (*e.g.*, without multi-face issues). Employing the two guidance in 2D lifting process, we use the normal map as shape encoding of the 2D diffusion model and unleash its power to generate high-quality and diversified results with 3D coherence. Then we achieve the final 3D results via photorealistic rendering through appearance modeling. (*"Everest's summit eludes many without* ***Sherpa****."*)

(MLPs) to query SDF values for each vertex along a regular grid. Next we sample a point set $\mathcal{P} = \{\boldsymbol{p}_i \in \mathbb{R}^3\}$ from M_0 with their SDF values $\{SDF(\boldsymbol{p}_i)\}$. For each $\boldsymbol{p}_i$, the DMTet network $\mathcal{F}$ can predict SDF value $s(\boldsymbol{p}_i)$, and a position offset $\Delta\boldsymbol{p}_i$ by:

$$(s(\boldsymbol{p}_i), \Delta\boldsymbol{p}_i) = \mathcal{F}(\boldsymbol{p}_i; \theta), \tag{2}$$

where θ is the parameters of network $\mathcal{F}$. Then, we incorporate 3D priors into the DMTet network $\mathcal{F}_\theta$ with the point set derived from 3D diffusion model by minimizing:

$$\mathcal{L}_{SDF} = \sum_{\boldsymbol{p}_i \in \mathcal{P}} |s(\boldsymbol{p}_i) - SDF(\boldsymbol{p}_i)|^2 + \lambda_{def} \sum_{\boldsymbol{p}_i \in \mathcal{P}} ||\Delta\boldsymbol{p}_i||_2, \tag{3}$$

where λ_{def} is the hyperparameter controlling L_2 regularization strengths on offsets to avoid artifacts. Finally, we apply the MT layer to extract mesh representation M. Now, we have leveraged the knowledge from the 3D diffusion model to construct a coarse 3D prior, which is encoded implicitly in DMTet $\mathcal{F}_\theta$ and represented explicitly by mesh M. Next, we will discuss how to utilize the coarse 3D prior M as a guide during the subsequent 2D diffusion lifting optimization to refine a high-quality result with 3D coherence.

3.3. 3D Guidance for 2D Lifting Optimization

What knowledge can serve as guidance? The purpose of introducing a 3D prior as guidance is to address the prevalent issue of viewpoint inconsistency both in geometry and appearance. Through empirical studies, we have identified geometric inconsistency as the main cause of 3D incoherence, leading to multi-face Janus problem [37, 69]. In contrast, appearance inconsistency emerges in much more extreme scenarios with lesser significance. Therefore, we disentangle the geometry from the 3D model and fully leverage coarse prior M to guide 2D lifting geometry optimization with view-point awareness. Our analysis of the coarse 3D prior indicates that it contains the essential geometric structures and captures the basic categorical attributes, keeping semantic rationality across different views. Building upon these observations, a natural insight is to preserve such inherent 3D knowledge as guidance and continuously benefit the 2D lifting process. For example, given text prompts "a head of the Terracotta Army," we hope the knowledge in the guidance can prevent issues such as a pockmarked face or the unrealistic scenario of having a face on the back (*e.g.*, Janus problem). To this end, we have designed two guiding strategies derived from M: structural guidance for geometric fidelity and semantic guidance for 3D coherence.

Structural guidance. Given the current DMTet net $\mathcal{F}$ with parameters θ that encodes the coarse 3D prior M, we apply a differentiable render f_n (*e.g.*, nvidiffrast [34]) to generate a set of normal maps $\mathcal{N} = \{\boldsymbol{n}_i | \boldsymbol{n}_i = f_n(\mathcal{F}_\theta, \mathbf{c}_i), i = 1, ...n\}$, where $\mathbf{c}_i$ is the camera position randomly sampled in spherical coordinates. To extract the salient geometric structure features, we first use a Gaussian filter with a kernel standard deviation σ

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}$$

to reduce the noise impact and obtain $\{\sigma(\boldsymbol{n}_i)\}$. As gradients are simple but effective tools for revealing the geometric contours and salient structures [14, 30], we then compute the structural descriptor sets $\{G_\sigma(\boldsymbol{n}_i)\}$ by

$$G_\sigma(\boldsymbol{n}_i) = \sqrt{(\frac{\partial\sigma(\boldsymbol{n}_i)}{\partial x})^2 + (\frac{\partial\sigma(\boldsymbol{n}_i)}{\partial y})^2}, \tag{5}$$

where x and y are the coordinate directions of the normal map $\boldsymbol{n}_i$. Throughout the 2D lifting process of updating $\mathcal{F}_\theta$ with newly rendered normal maps $\tilde{\mathcal{N}} = \{\tilde{\boldsymbol{n}}_i\}$, it should follow the structural guidance as:

$$\min_\theta \mathcal{L}_{\text{struc}} := \sum_{i=1}^{n} ||G_\sigma(\boldsymbol{n}_i) - G_\sigma(\tilde{\boldsymbol{n}}_i)||_2^2, \tag{6}$$

which enables the 2D lifting process to preserve geometric fidelity and a well-aligned structure with the coarse 3D prior when generating rich details.

Semantic guidance. While structural guidance maintains low-level geometric perception from coarse 3D prior, semantic guidance extracts high-level features for 3D coherence. We first apply the pre-trained CLIP [59] model as semantic encoder ψ to the normal set $\mathcal{N}$ and obtain semantic feature maps $\mathcal{N}_c = \{\psi(\boldsymbol{n}_i)\}$, proven to effectively capture semantic attributes like facial expressions or view categories [17]. Following the notation as above, we then define the semantic guidance with cosine similarity:

$$\min_\theta \mathcal{L}_{\text{sem}} := \sum_{i=1}^{n} \frac{\psi(\boldsymbol{n}_i)\cdot\psi(\tilde{\boldsymbol{n}}_i)}{\|\psi(\boldsymbol{n}_i)\|\|\psi(\tilde{\boldsymbol{n}}_i)\|}. \tag{7}$$

Employing this guidance, we ensure that different views retain inherent high-level information throughout the 2D lifting optimization process. Experiments show that it can effectively mitigate multi-face problems, keeping 3D content semantically plausible from all viewing angles.

3.4. Optimization

In this subsection, we incorporate both structural and semantic guidance derived from coarse 3D prior to 2D lifting optimization so that it can produce vivid and diversified objects with multi-view consistency. For the disentangled geometry modeling, we use the randomly sampled normal map $\boldsymbol{n}$ as the input, bridging the gap between 3D and 2D diffusion. To update the geometry model DMTet network $\mathcal{F}_\theta$, we choose to use the publicly available *Stable Diffusion* [62] as pre-trained 2D diffusion model ϕ and compute the gradient of the SDS loss similar in Eq. 14:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\theta, \boldsymbol{n}) = \mathbb{E}_{t,\epsilon}\left[w(t)\left(\epsilon_\phi\left(\boldsymbol{z}_t^{\boldsymbol{n}}; y, t\right) - \epsilon\right)\frac{\partial \boldsymbol{z}_t^{\boldsymbol{n}}}{\partial \boldsymbol{n}}\frac{\partial \boldsymbol{n}}{\partial \theta}\right], \tag{8}$$

where $\partial \boldsymbol{z}_t^{\boldsymbol{n}}/\partial \boldsymbol{n}$ calculates the gradient of the encoder in the latent diffusion model (LDM) [62]. Additionally, we introduce a step annealing technique to balance the influence of the 3D guidance during 2D lifting optimization:

$$\gamma(\lambda) = \lambda e^{-\beta \max(0, n_{\text{cur}} - m)}, \tag{9}$$

where n_{cur} is the current epoch and $\{\beta, m, \lambda\}$ are the hyperparameters that control how γ decreased. Therefore, the total loss $\mathcal{L}_{\text{geo}}$ to lift 2D geometry optimization with 3D guidance is a weighted sum of three loss terms:

$$\mathcal{L}_{\text{geo}}(\theta, \boldsymbol{n}) = \mathcal{L}_{\text{SDS}} + \gamma(\lambda_{\text{struc}})\mathcal{L}_{\text{struc}} + \gamma(\lambda_{\text{sem}})\mathcal{L}_{\text{sem}}, \tag{10}$$

which not only enables the 3D content generation without multi-view inconsistency issues but also preserves the generalization and quality in 2D diffusion model ϕ. As our pipeline can be integrated into any appearance model [9, 10, 35], we adopt a similar approach as Fantasia3D [10] to better align our text and 3D object. Denote $\mathcal{T}$ with parameters η as our appearance model, we have the rendered image $\mathbf{x} = \mathcal{T}_\eta(\mathcal{F}_\theta, \boldsymbol{c}_i)$. To update η, we again apply the SDS loss for the final complete generated 3D object with detailed texture and coherent geometry:

$$\nabla_\eta \mathcal{L}_{\text{app}}(\eta, \mathbf{x}) = \mathbb{E}_{t,\epsilon}\left[w(t)\left(\epsilon_\phi\left(\boldsymbol{z}_t^{\mathbf{x}}; y, t\right) - \epsilon\right)\frac{\partial \boldsymbol{z}_t^{\mathbf{x}}}{\partial \mathbf{x}}\frac{\partial \mathbf{x}}{\partial \eta}\right], \tag{11}$$

which shares similar notations defined in Eq. 8. Finally, through the tailored 3D structural and semantic guidance that bridges the 2D and 3D diffusion models, our Sherpa3D can mitigate the multi-face problem and achieve high-fidelity and diversified results.

3.5. Implementation Details

We apply the multilayer perceptron (MLP) comprising of three hidden layers to approximate $\mathcal{F}_\theta$ and $\mathcal{T}_\eta$. Adam optimizer [32] is used to update $\mathcal{F}_\theta$ and $\mathcal{T}_\eta$ with an initial learning rates of $1e-3$ decaying into $5e-4$. For 3D representations, we use textured mesh with a DMTet resolution of 128 to achieve a balance between quality and generation speed. We sample random camera poses at a fixed radius of 2.5, y-axis FOV of $45°$, with the azimuth in $[-180°, 180°]$ and elevation in $[-30°, 30°]$. We load Shap-E from [48] for 3D diffusion model and choose *stabilityai/stable-diffsuion-2-1-base* [62] for 2D diffusion model. For weighting factors, we follow the same strategy as [27] to tune $w(t)$. λ_{struc} is set to 10 and λ_{sem} is 30 to balance the magnitude of SDS loss. Notably, our method only needs a single NVIDIA RTX3090 (24GB) GPU within 25 minutes. More details of optimization, architecture design, and hyperparameter settings can be found in the supplementary.

4. Experiments

In this section, we conduct comprehensive experiments to evaluate our text-to-3D framework Sherpa3D and show

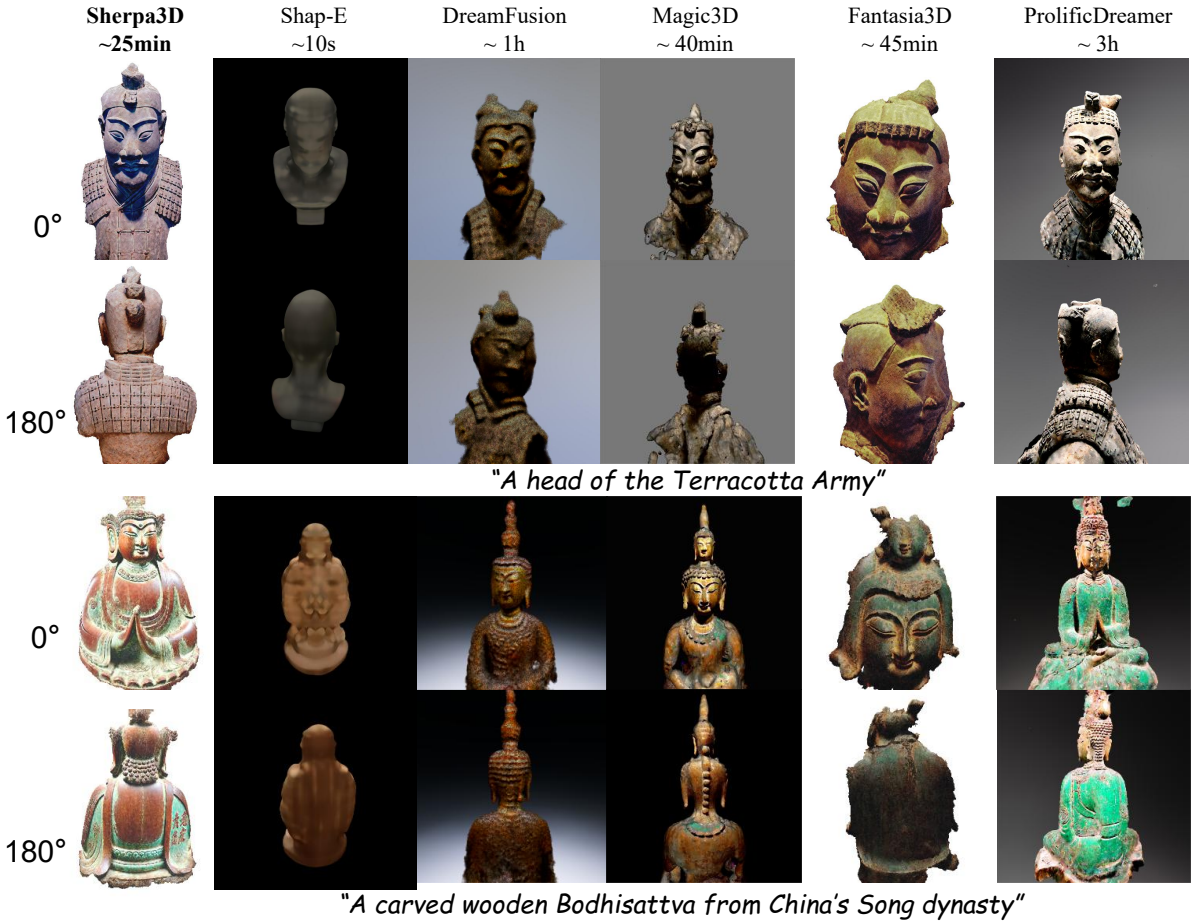


Figure 3. **Qualitative comparisons** with baseline methods across different views (0° and 180°). We can observe that baseline methods suffer from severe multi-face issues while our Sherpa3D can achieve better quality and 3D coherence.

comparison results against other text-to-3D baseline methods. We first present qualitative results compared with five SOTA baselines from different viewpoints. Then we report the quantitative results with a user study. Finally, we carry out ablation studies to further verify the efficacy of our framework design. Please refer to the supplementary for more comparisons, visualizations, and detailed analysis.

4.1. Experiment Setup

Baselines. We extensively compare our method Sherpa3D against five baselines: Shap-E [29], DreamFusion [58], Magic3D [39], ProlificDreamer [79], and Fantasis3D [10]. Due to various reasons, we can't obtain the original implementation of some baselines. For DreamFusion, Magic3D, and ProlificDreamer, we utilize their implementations in the Threestudio library [19] for comparison. For Shap-E and Fantasia3D, we follow their official implementation. We consider these implementations to be the most reliable and comprehensive open-source option available in the field. To ensure a fair comparison, we use the Stable Diffusion [62] model as 2D diffusion prior by default.

Metrics. We will show our results with notable comparisons to other baselines through visualization. As there is no Ground-Truth 3D content corresponding to the text prompt, reference-based metrics like Chamfer Distance are difficult to apply to zero-shot text-to-3D generation. Following [28, 58], we evaluate the CLIP R-Precision [57], which can measure how well the rendered images of generated 3D content align with the input text. We use 100 prompts from the Common Objects in Context (COCO) dataset [40] as DreamFusion [58]. we also conduct a user study to further demonstrate the multi-view consistency and overall generation quality of our method,

4.2. Qualitative Comparisons

We first demonstrate vivid and diversified text-to-3D results generated from our Sherpa3D in the gallery as shown in Figure 1. Then we compare our method with five baseline method: Shap-E [29], DreamFusion [58], Magic3D [39], Fantasia3D [10] and ProlificDreamer [79]. Figure 3 and 4

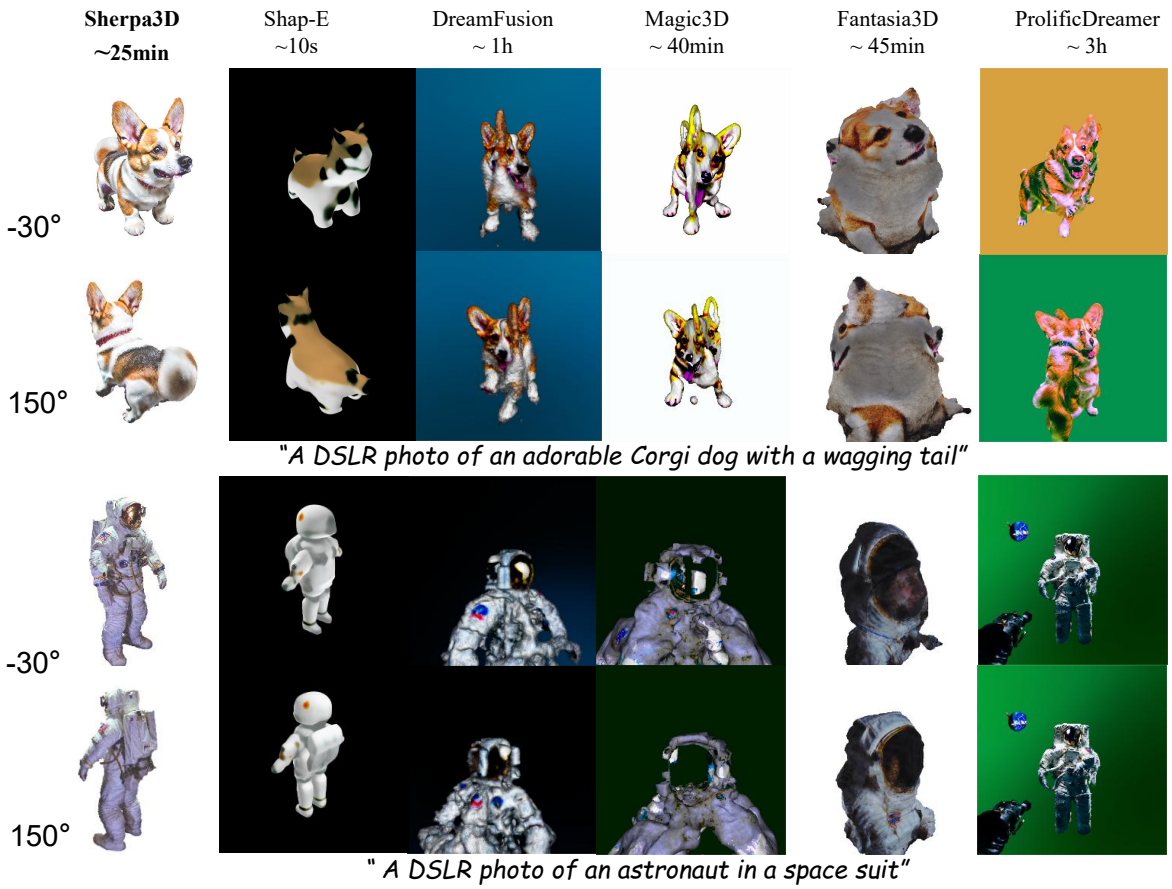


Figure 4. **Qualitative comparisons** with baseline methods across different views (-30° and 150°).

give the comparative results with the same text prompt for each object generation. We observe that the Shap-E [29] only generates coarse shapes while other 2D lifting methods suffer from multi-face problems. In contrast, our Sherpa3D produces high-fidelity 3D assets with compelling texture quality and multi-view consistency. Notably, our framework is more efficient than other baselines with less time to optimize. Specifically, it only takes within 25 minutes from a text prompt to a high-quality 3D model ready to be used in graphic engines.

4.3. Quantitative Comparisons

In Table 1, we report the CLIP R-Precision for Sherpa3D and several baselines. It shows that our method outperforms other baselines consistently across different CLIP models, and approaches the performance of ground truth (GT) images. For the user study, we render 360-degree rotating videos of 3D models generated from a collection of 120 images. Each volunteer is shown 10 samples of rendered video from a random method and rates in two aspects: multi-view consistency and overall generation quality. We collect results from 50 volunteers shown in Table 2. We observe that most users consider our results with much higher viewpoints consistency and overall generation fidelity.

Table 1. **Quantitative comparisons** on generation renderings with text prompts using different CLIP retrieval models. We compared to ground-truth images, Shap-E [29], Dreamfusion [58], Magic3D [39], evaluated on object-centric COCO as in [58].

Method	R-Precision (%) ↑		
	CLIP B/32	CLIP B/16	CLIP L/14
GT Images	77.3	79.2	-
Shape-E [29]	41.1	42.5	46.4
DreamFusion [58]	70.3	73.2	75.0
Magic3D [39]	71.5	73.8	76.1
Sherpa3D (Ours)	**72.3**	**75.6**	**79.3**

4.4. Ablation Study and Analysis

We carry out ablation studies on the design of our Sherpa3D framework in Figure 5 using an example text prompt "a head of the Terracotta Army". Specifically, we perform ablation on three aspects of our method: structural guidance, semantic guidance, and the step annealing strategy. The re-

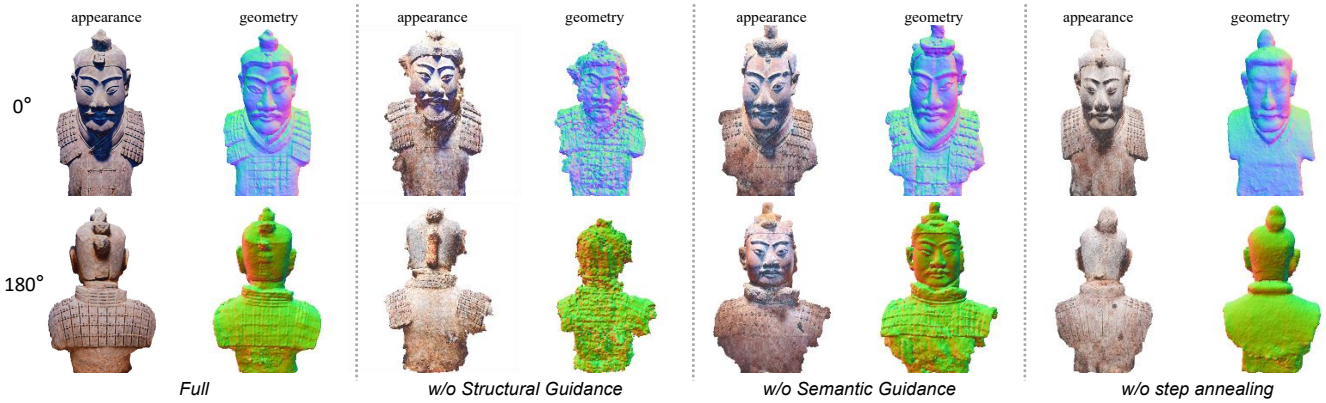


Figure 5. **Ablation study of our method.** The generation is based on the text prompt "a head of the Terracotta Army". We ablate the design choices of structural guidance, semantic guidance (Sec. 3.3), and the step annealing technique (Sec. 3.4).

Table 2. **Quantitative comparisons** on the multi-view consistency and overall generation quality score in a user study, rated on a scale of 1-10, with higher scores indicating better performance.

Method	Multi-view Consistency ↑	Overall Quality ↑
Shap-E [29]	6.09	3.33
DreamFusion [58]	4.58	6.05
Magic3D [39]	5.25	6.73
Fantasia3D [10]	3.83	5.90
ProlificDreamer [79]	5.78	7.02
Sherpa3D(Ours)	**8.95**	**8.74**

Figure 6. Comparison with Zero123 [42]. We use the front view of our generated 3D model as the input of Zero123 with open-vocabulary text prompts.

sults reveal that the omission of any of these elements leads to a degradation in terms of quality and consistency. Notably, the absence of structural guidance leads to a loss of geometric fidelity in the "army", leading to a pockmarked face; without semantic guidance, there's a loss of semantic rationality across different views, resulting in the multi-view Janus problem. The lack of a balanced step annealing results in an excessive influence of guidance with a rough final output. This illustrates the effectiveness of our overall framework (Figure 2), which drives geometric fidelity, multi-view consistency, and optimization balance steered by the 3D guidance and annealing strategy.

To further demonstrate our generalizability, we compare our method in Figure 6 with the Zero123 [42] which uses more 3D data [42] to finetune a 2D diffusion model to be viewpoint-aware. However, such a finetuning-based method easily overfits to 3D training data and suffers from severe performance degradation with unseen input of the training set. In contrast, our method is more generalizable to open-vocabulary text prompts.

5. Conclusion

In this paper, we present Sherpa3D, a new framework that simultaneously achieves high-quality, diversified, and 3D consistent text-to-3D generation. By fully exploiting easily obtained coarse 3D knowledge from the 3D diffusion model, we derive structural guidance and semantic guidance to enhance the prompts and provide continuous guidance with geometric fidelity and 3D coherence throughout the 2D lifting optimization. To further improve the overall performance, we incorporate a step annealing strategy that modulates the impact of 3D guidance and 2D refinement. Therefore, our framework bridges the gap between 2D and 3D diffusion models, preserving multi-view coherent generation while maintaining the generalizability and fidelity of 2D models. Extensive qualitative and quantitative experiments verify the remarkable improvement of our Sherpa3D on text-to-3D generation.

Limitations and future works. Although our Sherpa3D achieves remarkable text-to-3D results, the quality still seems to be limited to the backbone itself as we choose Shap-E [29] and Stable Diffusion v2.1 base model in this work. We expect them to be solved with a larger diffusion model, such as SDXL [1] and DeepFloyd [2]. In future work, we are interested in extending our insight to more creative text-to-4D generation. We believe that Sherpa3D provides a promising research path for user-friendly and more accessible 3D content creation.

References

[1] stable-diffusion-xl-base-1.0. https://huggingface.co/stabilityai/stable-diffusion-xl-base-1.0. Accessed: 2023-08-29. 8

[2] Deepfloyd. https://huggingface.co/DeepFloyd. Accessed: 2023-08-25. 8

[3] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds. In *ICML*, pages 40–49. PMLR, 2018. 3

[4] Mohammadreza Armandpour, Huangjie Zheng, Ali Sadeghian, Amir Sadeghian, and Mingyuan Zhou. Re-imagine the negative prompt algorithm: Transform 2d diffusion into 3d, alleviate janus problem and beyond. *arXiv preprint arXiv:2304.04968*, 2023. 2, 3

[5] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021. 3

[6] Eric R Chan, Marco Monteiro, Petr Kellnhofer, Jiajun Wu, and Gordon Wetzstein. pi-gan: Periodic implicit generative adversarial networks for 3d-aware image synthesis. In *CVPR*, pages 5799–5809, 2021. 3

[7] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *CVPR*, pages 16123–16133, 2022. 3

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

[9] Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. *arXiv preprint arXiv:2303.11396*, 2023. 5, 13

[10] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023. 2, 3, 5, 6, 8, 13, 14

[11] 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. *arXiv preprint arXiv:2307.05663*, 2023. 2

[12] 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, 3, 14

[13] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[14] Lijun Ding and Ardeshir Goshtasby. On the canny edge detector. *Pattern recognition*, 34(3):721–725, 2001. 5

[15] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *3DV*, pages 402–411. IEEE, 2017. 3

[16] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *NeurIPS*, 35:31841–31854, 2022. 3

[17] Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 6(3):e30, 2021. 5

[18] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. *arXiv preprint arXiv:2110.08985*, 2021. 3

[19] Yuan-Chen Guo, Ying-Tian Liu, Ruizhi Shao, Christian Laforte, Vikram Voleti, Guan Luo, Chia-Hao Chen, Zi-Xin Zou, Chen Wang, Yan-Pei Cao, and Song-Hai Zhang. threestudio: A unified framework for 3d content generation. https://github.com/threestudio-project/threestudio, 2023. 6

[20] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation, 2023. 2, 3

[21] Zekun Hao, Arun Mallya, Serge Belongie, and Ming-Yu Liu. Gancraft: Unsupervised 3d neural rendering of minecraft worlds. In *ICCV*, pages 14072–14082, 2021. 3

[22] Philipp Henzler, Niloy J Mitra, and Tobias Ritschel. Escaping plato's cave: 3d shape from adversarial rendering. In *ICCV*, pages 9984–9993, 2019. 3

[23] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 13

[24] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 3

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2, 13

[26] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. *arXiv preprint arXiv:2303.11989*, 2023. 2

[27] Yukun Huang, Jianan Wang, Yukai Shi, Xianbiao Qi, Zheng-Jun Zha, and Lei Zhang. Dreamtime: An improved optimization strategy for text-to-3d content creation. *arXiv preprint arXiv:2306.12422*, 2023. 3, 5, 14

[28] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876, 2022. 2, 6

[29] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 2, 3, 6, 7, 8, 14

[30] Nick Kanopoulos, Nagesh Vasanthavada, and Robert L Baker. Design of an image edge detection filter using the sobel operator. *IEEE Journal of solid-state circuits*, 23(2): 358–367, 1988. 5

[31] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 34:21696–21707, 2021. 13

[32] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5, 13

[33] Matthias Labschütz, Katharina Krösl, Mariebeth Aquino, Florian Grashäftl, and Stephanie Kohl. Content creation for a 3d game with maya and unity 3d. *Institute of Computer Graphics and Algorithms, Vienna University of Technology*, 6:124, 2011. 2

[34] Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020. 4

[35] Jiabao Lei, Yabin Zhang, Kui Jia, et al. Tango: Text-driven photorealistic and robust 3d stylization via lighting decomposition. *Advances in Neural Information Processing Systems*, 35:30923–30936, 2022. 5, 13

[36] Chenghao Li, Chaoning Zhang, Atish Waghwase, Lik-Hang Lee, Francois Rameau, Yang Yang, Sung-Ho Bae, and Choong Seon Hong. Generative ai meets 3d: A survey on text-to-3d in aigc era. *arXiv preprint arXiv:2305.06131*, 2023. 2

[37] Weiyu Li, Rui Chen, Xuelin Chen, and Ping Tan. Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. *arXiv preprint arXiv:2310.02596*, 2023. 2, 4

[38] Yuhan Li, Yishun Dou, Yue Shi, Yu Lei, Xuanhong Chen, Yi Zhang, Peng Zhou, and Bingbing Ni. Focaldreamer: Text-driven 3d editing via focal-fusion assembly. *arXiv preprint arXiv:2308.10608*, 2023. 3

[39] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 300–309, 2023. 2, 3, 6, 7, 8, 14

[40] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014. 6

[41] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Zexiang Xu, Hao Su, et al. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *arXiv preprint arXiv:2306.16928*, 2023. 3

[42] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9298–9309, 2023. 2, 3, 8, 14

[43] 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. 3

[44] 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. *arXiv preprint arXiv:2310.15008*, 2023. 3

[45] Jonathan Lorraine, Kevin Xie, Xiaohui Zeng, Chen-Hsuan Lin, Towaki Takikawa, Nicholas Sharp, Tsung-Yi Lin, Ming-Yu Liu, Sanja Fidler, and James Lucas. Att3d: Amortized text-to-3d object synthesis. *arXiv preprint arXiv:2306.07349*, 2023. 3

[46] Sebastian Lunz, Yingzhen Li, Andrew Fitzgibbon, and Nate Kushman. Inverse graphics gan: Learning to generate 3d shapes from unstructured 2d data. *arXiv preprint arXiv:2002.12674*, 2020. 3

[47] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *CVPR*, pages 2837–2845, 2021. 3

[48] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *arXiv preprint arXiv:2306.07279*, 2023. 5

[49] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673, 2023. 2, 3

[50] 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. In *European Conference on Computer Vision (ECCV)*, pages 405–421. Springer, 2020. 3

[51] Kaichun Mo, Paul Guerrero, Li Yi, Hao Su, Peter Wonka, Niloy Mitra, and Leonidas J Guibas. Structurenet: Hierarchical graph networks for 3d shape generation. *arXiv preprint arXiv:1908.00575*, 2019. 3

[52] Nasir Mohammad Khalid, Tianhao Xie, Eugene Belilovsky, and Tiberiu Popa. Clip-mesh: Generating textured meshes from text using pretrained image-text models. In *SIGGRAPH Asia 2022 conference papers*, pages 1–8, 2022. 2

[53] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)*, 41(4):1–15, 2022. 3

[54] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts, 2022. 2, 3

[55] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 2

[56] Roy Or-El, Xuan Luo, Mengyi Shan, Eli Shechtman, Jeong Joon Park, and Ira Kemelmacher-Shlizerman. Stylesdf: High-resolution 3d-consistent image and geometry generation. In *CVPR*, pages 13503–13513, 2022. 3

[57] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021. 6

[58] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. 2, 3, 6, 7, 8, 13, 14

[59] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 5

[60] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023. 3

[61] 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. 2, 13

[62] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 2, 3, 5, 6, 14

[63] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding, 2022. 2, 3, 13

[64] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 2

[65] Katja Schwarz, Axel Sauer, Michael Niemeyer, Yiyi Liao, and Andreas Geiger. Voxgraf: Fast 3d-aware image synthesis with sparse voxel grids. *NeurIPS*, 35:33999–34011, 2022. 3

[66] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, 2018. 2

[67] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. *Advances in Neural Information Processing Systems*, 34:6087–6101, 2021. 3

[68] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023. 3

[69] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. 2, 3, 4, 14

[70] Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. Text-to-4d dynamic scene generation. *arXiv preprint arXiv:2301.11280*, 2023. 3

[71] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 13

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

[73] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 13

[74] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 13

[75] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 3

[76] Christina Tsalicoglou, Fabian Manhardt, Alessio Tonioni, Michael Niemeyer, and Federico Tombari. Textmesh: Generation of realistic 3d meshes from text prompts. *arXiv preprint arXiv:2304.12439*, 2023. 3

[77] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A. Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation, 2022. 2, 3, 13

[78] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021. 3

[79] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023. 2, 3, 6, 8, 14

[80] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022. 2

[81] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in neural information processing systems*, 29, 2016. 2

[82] Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian,

et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 803–814, 2023. 3

[83] Jiayu Yang, Ziang Cheng, Yunfei Duan, Pan Ji, and Hongdong Li. Consistnet: Enforcing 3d consistency for multi-view images diffusion. *arXiv preprint arXiv:2310.10343*, 2023. 3

[84] Chaohui Yu, Qiang Zhou, Jingliang Li, Zhe Zhang, Zhibin Wang, and Fan Wang. Points-to-3d: Bridging the gap between sparse points and shape-controllable text-to-3d generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6841–6850, 2023. 3

[85] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *arXiv preprint arXiv:2301.11445*, 2023. 3

[86] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2

[87] Yuxuan Zhang, Wenzheng Chen, Huan Ling, Jun Gao, Yinan Zhang, Antonio Torralba, and Sanja Fidler. Image gans meet differentiable rendering for inverse graphics and interpretable 3d neural rendering. *arXiv preprint arXiv:2010.09125*, 2020. 3

[88] Xin-Yang Zheng, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Heung-Yeung Shum. Locally attentional sdf diffusion for controllable 3d shape generation. *arXiv preprint arXiv:2305.04461*, 2023. 3

[89] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018. 13

[90] Joseph Zhu and Peiye Zhuang. Hifa: High-fidelity text-to-3d with advanced diffusion guidance. *arXiv preprint arXiv:2305.18766*, 2023. 3

[91] Jingyu Zhuang, Chen Wang, Lingjie Liu, Liang Lin, and Guanbin Li. Dreameditor: Text-driven 3d scene editing with neural fields. *arXiv preprint arXiv:2306.13455*, 2023. 3

Sherpa3D: Boosting High-Fidelity Text-to-3D Generation via Coarse 3D Prior

Supplementary Material

6. More Discussion of Preliminaries

In this section, we provide more preliminaries and details of our implementation for Score Distillation Sampling (SDS).

6.1. Diffusion Models

The diffusion model, which is a type of likelihood-based generative model used to learn data distributions, has been studied extensively in recent years [25, 71–74]. Given an underlying data distribution $q_0(\boldsymbol{x})$, a diffusion model composes two processes: (a) a forward process $\{q_t\}_{t\in[0,1]}$ to gradually add noise to the data point $\boldsymbol{x}_0 \sim q_0(\boldsymbol{x}_0)$; (b) a reverse process $\{p_t\}_{t\in[0,1]}$ to denoise data (*e.g.*, generation). Specifically, the forward process is defined by $q_t\left(\boldsymbol{x}_t \mid \boldsymbol{x}_0\right) := \mathcal{N}\left(\alpha_t \boldsymbol{x}_0, \sigma_t^2 \boldsymbol{I}\right)$ and $q_t\left(\boldsymbol{x}_t\right) := \int q_t\left(\boldsymbol{x}_t \mid \boldsymbol{x}_0\right) q_0\left(\boldsymbol{x}_0\right) \mathrm{d}\boldsymbol{x}_0$, where $\alpha_t, \sigma_t > 0$ are hyperparameters. On the other hand, the reverse process is described with the transition kernel $p_t(\boldsymbol{x}_{t-1} \mid \boldsymbol{x}_t) := \mathcal{N}(\mu_\phi(\boldsymbol{x}_t, t), \sigma_t^2 \boldsymbol{I})$ from $p_1(\boldsymbol{x}_1) := \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The training objective is to optimize μ_ϕ by maximizing a variational lower bound of a log-likelihood. In practice, μ_ϕ is reparameterized as a denoising network $\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t, t)$ [25] to predict the noise added to the clean data $\boldsymbol{x}_0$, which is trained by minimizing the MSE criterion [25, 31]:

$$\mathcal{L}_{\text{Diff}}\left(\phi\right) := \mathbb{E}_{\boldsymbol{x}_0, t, \boldsymbol{\epsilon}}\left[\omega(t)\left\|\boldsymbol{\epsilon}_\phi\left(\alpha_t \boldsymbol{x}_0 + \sigma_t \boldsymbol{\epsilon}\right) - \boldsymbol{\epsilon}\right\|_2^2\right], \quad (12)$$

where $\omega(t)$ is the time-dependent weights. Besides, the noise prediction network $\boldsymbol{\epsilon}_\phi$ can be applied for approximating the score function [73] of the perturbed data distribution $q(\boldsymbol{x}_t)$, which is defined as the gradient of the log-density:

$$\nabla_{\boldsymbol{x}_t} \log q_t\left(\boldsymbol{x}_t\right) \approx -\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t, t\right) / \sigma_t. \quad (13)$$

This means that the diffusion model can estimate a direction that guides $\boldsymbol{x}_t$ towards a high-density region of $q(\boldsymbol{x}_t)$, which is the key idea Score Distillation Sampling (SDS) [58, 77] for optimizing the 3D scene via well 2D pre-trained models.

6.2. SDS with Classifier-Free Guidance

As one of the most successful applications of diffusion models, text-to-image generation [61–63] generate samples $\boldsymbol{x}$ based on the text prompt y, which is also fed into the $\boldsymbol{\epsilon}_\phi$ as input, denoted as $\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y\right)$. An important technique to improve the performance of these models is Classifier-Free Guidance (CFG) [23]. CFG modifies the original model by adding a guidance term, *i.e.*, $\hat{\boldsymbol{\epsilon}}_\phi(\boldsymbol{x}_t; y, t) := (1+s)\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; y, t) - s\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; t, \varnothing)$, where $s > 0$ is the guidance weight that controls the balance between fidelity and diversity, while $\varnothing$ denotes the "empty" text prompt for the unconditional case. Recall the SDS gradient form to update θ:

$$\nabla_\theta \mathcal{L}_{\text{SDS}}(\phi, \boldsymbol{x}) = \mathbb{E}_{t,\epsilon}\left[\omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; y, t\right) - \epsilon\right)\frac{\partial \boldsymbol{x}}{\partial \theta}\right], \quad (14)$$

and denote $\delta_{\boldsymbol{x}}(\boldsymbol{x}_t; y, t) := \epsilon_\phi(\boldsymbol{x}_t; y, t) - \boldsymbol{\epsilon}$. In principle, $\boldsymbol{\epsilon}(\boldsymbol{x}_t; y, t)$ should represent the pure text-conditioned score function in Eq. (14). But in practice, CFG is employed in it with a guidance weight s to achieve high-quality results, where we rewrite

$$\delta_{\boldsymbol{x}}(\boldsymbol{x}_t; y, t) = [\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; y, t) - \boldsymbol{\epsilon}] + s[\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; y, t) - \boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; t, \varnothing)]. \quad (15)$$

As DreamFusion [58] uses $s = 100$ for high fidelity, our implementation adopts $s = 50$ with the enhancement of structural and semantic guidance to preserve some diversity. The two types of guidance can also be seen as another form of prompt guidance that is more generalizable and robust. Therefore, there is a gap between the original formulation in Eq. (14) and the practical coding implementation in Eq. (15).

7. Additional Implementation Details

Training details. Our geometry model $\mathcal{F}_\theta$ and appearance model $\mathcal{T}_\eta$ is approximated by three-layer MLPs and we apply adam [32] optimizer to update them with an initial learning rates of 1×10^{-3} to decaying to 5×10^{-4}. In particular, our method is optimized for 2500 iterations about 15 minutes to learn $\mathcal{F}_\theta$ and 2500 iterations about 10 minutes to learn $\mathcal{T}_\eta$. For geometry modeling, we utilize the Open3D library [89] to calculate the signed distance function (SDF) value for each point in Equations 2 and 3 in the main paper. In our experiments, the DMTet-based coarse 3D prior building stage is critical as it not only provides coarse 3D knowledge with consistency but also boosts the speed of the convergence of generation. For appearance modeling, since our focus in this paper is to fully exploit easily obtained coarse 3D knowledge that serves as guidance for 2D lifting optimization (as discussed in Section 3.3 of our paper), we do not design a specific appearance model for our framework. Note that our geometry model is plug and play and we can leverage different models [9, 10, 35], we leverage the same PBR materials approach in Fantasia3D [10] to achieve photorealistic surface renderings and better aligns with our geometry modeling.

Hyperparameter settings. We select the camera positions (r, κ, φ) in the spherical coordinate system, where r denote

radius, κ is the elevation and φ is the azimuth angle respectively. Specifically, we sample random camera poses at a fixed $r = 2.5$ with the $\kappa \in [-30°, 30°]$. In a batch of $b \times l$ images, we partition φ into l intervals in $[-180°, 180°]$ and uniformly sample b azimuth angles in each interval. For structural guidance, we set $\sigma = 1$ in Eq. (4) in the main paper as the standard deviation of the Gaussian filter. We tune λ_{struc} and λ_{sem} in $\{0.01, 0.1, 1, 5, 10, 20, 30, 100\}$. We find that often $\lambda_{\text{struc}} = 10$ and $\lambda_{\text{sem}} = 30$ works well with $\beta = 0.5$ in the step annealing technique, which may balance the magnitude of SDS losses and better guide the 2D lifting to refine the 3D contents with multi-view coherence. We assigned the value of m to the epoch at around 1000 iterations. For the guidance weight $\omega(t)$, we follow the DreamTime [27] to achieve higher fidelity results. Our codes for implementation will be available upon acceptance.

8. Additional Experiments and Analysis

8.1. Additional User Study

To further demonstrate the effectiveness and impressive visualization results of our Sherpa3D, we conducted a more intuitive user study (Figure 7) on 20 text prompts of five baselines (ShapE [29], DreamFusion [58], Magic3D [39], ProlificDreamer [79], Fantasia3D [10]) and ours. The study engaged 50 volunteers to assess the generated results in 20 rounds. In each round, they were asked to select the 3D model they preferred the most, based on quality, creativity, alignment with text prompts, and consistency. We also compare our method with recent finetuning-based techniques, such as Zero123 [42] and MVDream [69], which utilize more 3D data [12] to retrain a costly 3D aware diffusion model from Stable Diffusion [62]. We use the same text prompts and settings as mentioned above. As shown, we

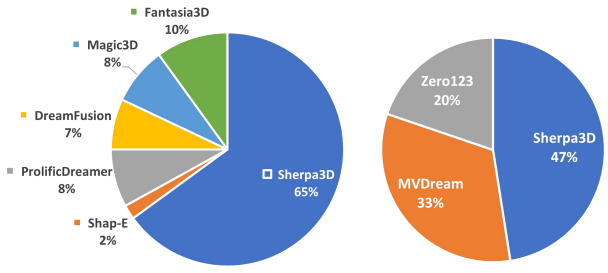


Figure 7. **User study** of the rate from volunteers' preference for each method in the inset pie chart.

observe that Sherpa3D is preferable (65%) by the raters on average. In other words, our model is preferred over the best of all baselines in most cases. What's more, our Sherpa3D also outperforms than fine-tuning based method in terms of overall performance as they easily suffer from styles (lightning, texture) overfitting [42, 69]. We believe this is strong proof of the robustness and quality of our proposed method.

8.2. More Qualitative Results

Sherpa3D. In Figure 10, 11, 12, we present more text-to-3D results obtained with Sherpa3D, which can generate high-fidelity, diverse, and 3D-consistent results within 25 minutes. Besides the impressive 3D consistency and high fidelity, we can also change the style of generated 3D content (Figure 8) by only modifying a small part of the prompt, while preserving the basic structure of 3D content, which is more convenient for users to flexibly edit generated objects.

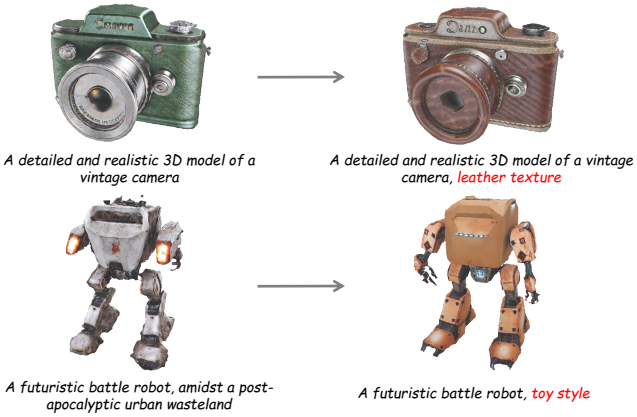


Figure 8. Sherpa3D can be used for flexible editing through a small part of the prompt modification.

More comparison results. We provide more comparisons with baselines in Figure 13, 14. To further demonstrate the robustness and generalization of our method, we compare our Sherpa3D with Zero123 [42] and MVDream [69] in Figure 9. Although the concurrent work MVDream and Zero123 can also resolve the multi-view inconsistency issues via fine-tuning a costly viewpoints-aware model, we observe that it is prone to overfit the limited 3D data [12]. Specifically, MVDream generates strange color styles while Zero123 fails in such open-vocabulary prompts.

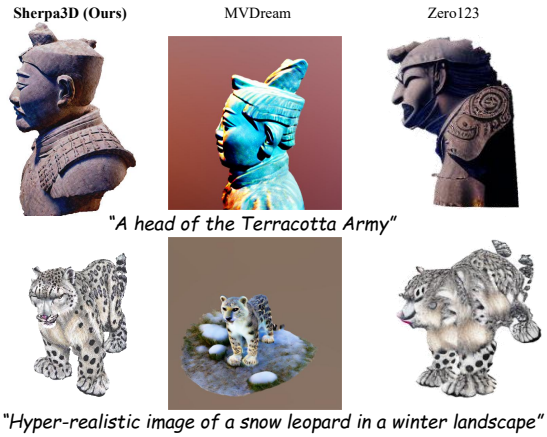


Figure 9. Comparison with MVDream [69] and Zero123 [42].

"A statue of a angel"

"A futuristic battle robot, heavily armed, amidst a post-apocalyptic urban wasteland"

"A cybernetic biomechanical arm, with a blend of organic and mechanical elements"

"A luxurious sky-blue leather handbag with a sleek and elegant design, highlighted by its vibrant blue color"

"Iron Man in his state-of-the-art suit, confidently standing, looking ahead, ready for action"

"Commercial airliner in flight, sleek and modern design"

Figure 10. More generated results using our Sherpa3D within 25 minutes. Our work can generate high-fidelity and diversified 3D results from various text prompts, free from the multi-view inconsistency problem.

"Detailed portrait of a noble knight, full armor, intricate helmet design"

"Hyper-realistic image of a snow leopard, capturing its camouflage and majestic stance"

"A detailed and realistic 3D model of a vintage camera"

"Spaceship,futuristic design,sleek metal,glowing thrusters, flying in space"

"A DLSR Photo of the Leaning Tower of Pisa"

"An ultra-detailed illustration of a mythical Phoenix, rising from ashes, vibrant feathers in a fiery palette"

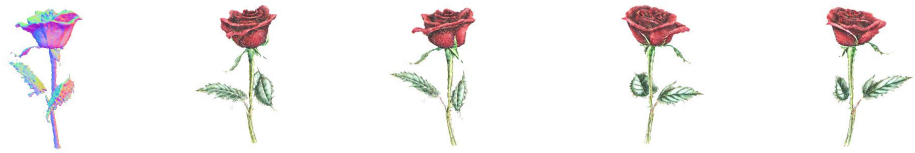
Figure 11. More generated results using our Sherpa3D within 25 minutes. Our work can generate high-fidelity and diversified 3D results from various text prompts, free from the multi-view inconsistency problem.

"A carved wooden Bodhisattva from China's Song dynasty"

"A futuristic-style motorcycle with sleek design, neon lights, and a sci-fi aesthetic in an urban setting"

"Vintage wooden race car, polished mahogany finish, classic design with spoked wheels"

"A head of the Terracotta Army"



"A blooming red rose, with velvety petals, delicate green leaves, and a captivating fragrance that fills the air"

"A DSLR photo of an adorable Corgi dog with a wagging tail"

Figure 12. More generated results using our Sherpa3D within 25 minutes. Our work can generate high-fidelity and diversified 3D results from various text prompts, free from the multi-view inconsistency problem.

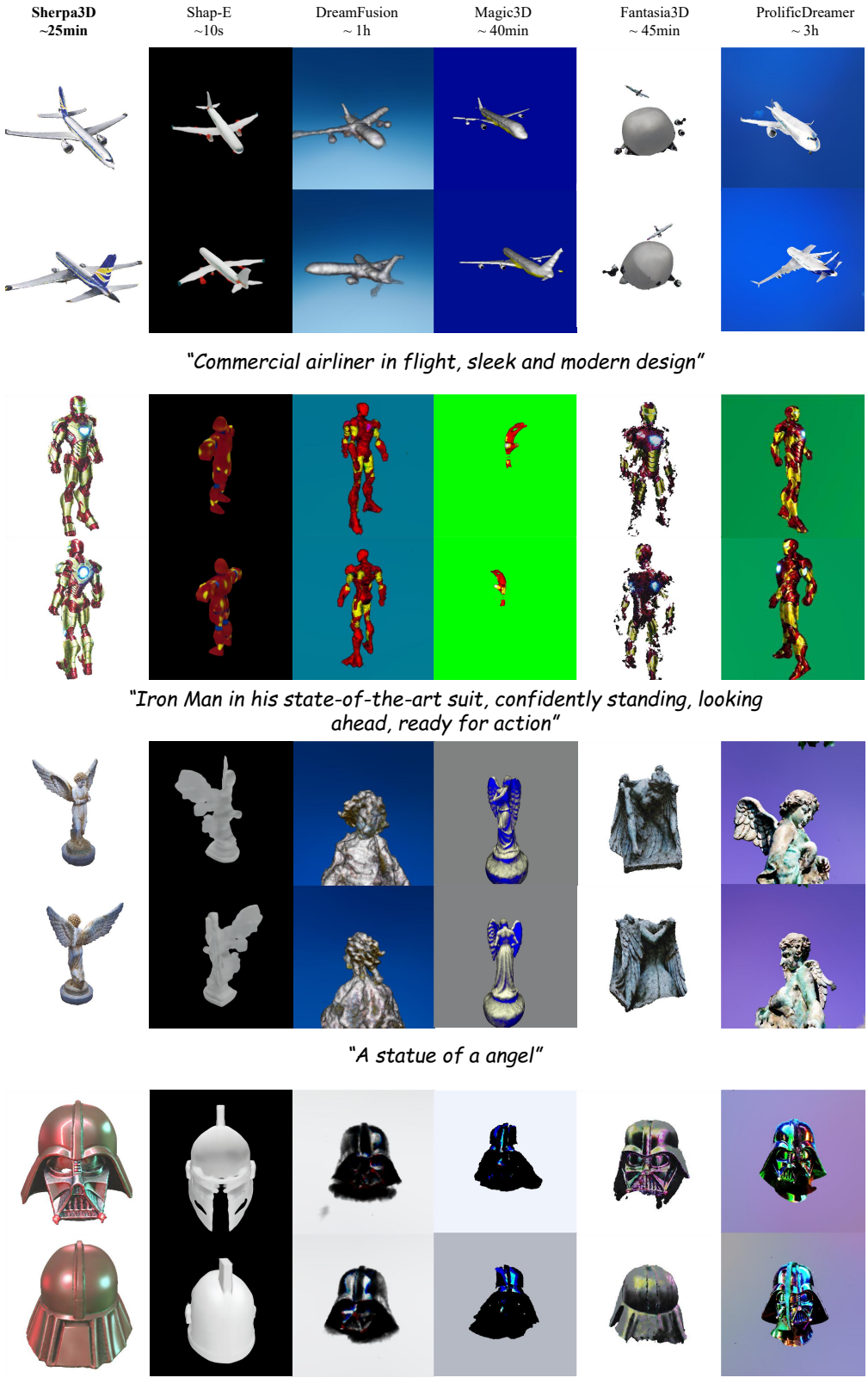


Figure 13. Qualitative comparisons with baseline methods across different views. All methods use *stabilityai/stable-diffsuion-2-1-base* for fair comparison. We observe that baselines suffer from severe multi-face issues while Sherpa3D achieves better quality and 3D coherence.

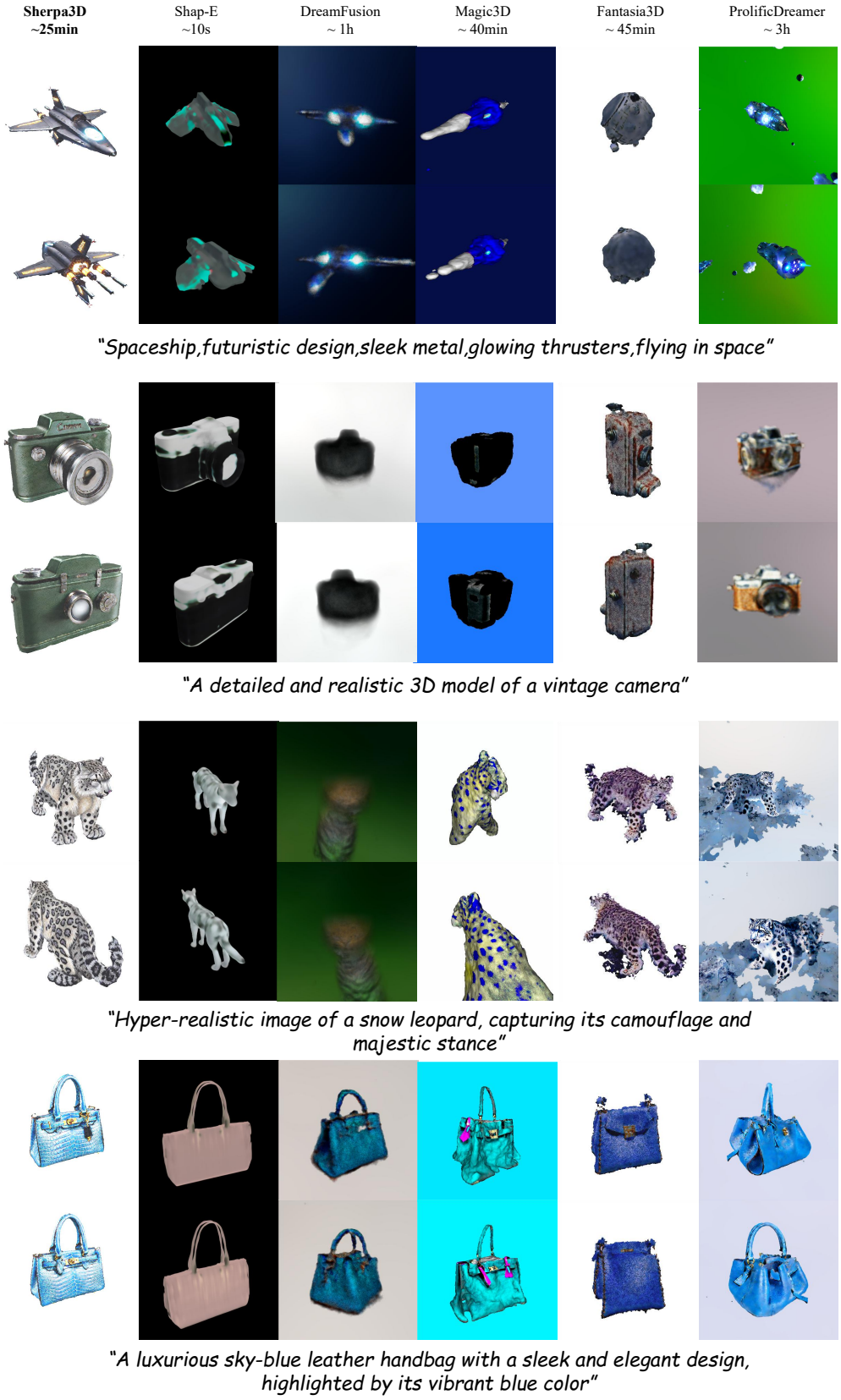


Figure 14. Qualitative comparisons with baseline methods across different views. All methods use *stabilityai/stable-diffsuion-2-1-base* for fair comparison. We observe that baselines suffer from severe multi-face issues while Sherpa3D achieves better quality and 3D coherence.