

Controllable Text-to-3D Generation via Surface-Aligned Gaussian Splatting

Zhiqi Li^{1,2}, Yiming Chen^{2,3}, Lingzhe Zhao², and Peidong Liu^{2†}

¹ Zhejiang University ² Westlake University ³ Tongji University

Project Page: <https://lizhiqi49.github.io/MVControl/>

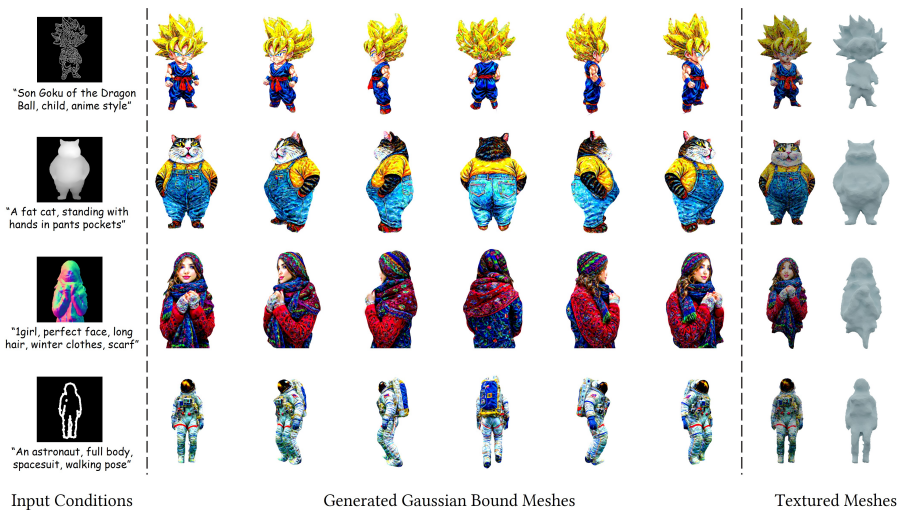


Fig. 1: Given a text prompt and a condition image, our method is able to achieve high-fidelity and efficient controllable text-to-3D generation of Gaussian binded mesh and textured mesh.

Abstract. While text-to-3D and image-to-3D generation tasks have received considerable attention, one important but under-explored field between them is controllable text-to-3D generation, which we mainly focus on in this work. To address this task, 1) we introduce Multi-view ControlNet (MVControl), a novel neural network architecture designed to enhance existing pre-trained multi-view diffusion models by integrating additional input conditions, such as edge, depth, normal, and scribble maps. Our innovation lies in the introduction of a conditioning module that controls the base diffusion model using both local and global embeddings, which are computed from the input condition images and camera poses. Once trained, MVControl is able to offer 3D diffusion guidance for

† Corresponding author.

optimization-based 3D generation. And, **2)** we propose an efficient multi-stage 3D generation pipeline that leverages the benefits of recent large reconstruction models and score distillation algorithm. Building upon our MVControl architecture, we employ a unique hybrid diffusion guidance method to direct the optimization process. In pursuit of efficiency, we adopt 3D Gaussians as our representation instead of the commonly used implicit representations. We also pioneer the use of SuGaR, a hybrid representation that binds Gaussians to mesh triangle faces. This approach alleviates the issue of poor geometry in 3D Gaussians and enables the direct sculpting of fine-grained geometry on the mesh. Extensive experiments demonstrate that our method achieves robust generalization and enables the controllable generation of high-quality 3D content.

Keywords: Controllable 3D Generation · Gaussian Splatting · SuGaR

1 Introduction

Remarkable progress has recently been achieved in the field of 2D image generation, which has subsequently propelled research in 3D generation tasks. This progress is attributed to the favorable properties of image diffusion models [31,44] and differentiable 3D representations [24,38,48,59]. In particular, recent methods based on score distillation optimization (SDS) [42] have attempted to distill 3D knowledge from pre-trained large text-to-image generative models [31,44,50], leading to impressive results [11,29,37,42,56,57,63].

Several approaches aim to enhance generation quality, such as applying multiple optimization stages [11,29], optimizing the diffusion prior with 3D representations simultaneously [52,63], refining score distillation algorithms [22,68], and improving pipeline details [4,20,70]. Another focus is on addressing view-consistency issues by incorporating multi-view knowledge into pre-trained diffusion models [27,31,32,34,49,50]. However, achieving high-quality 3D assets often requires a combination of these techniques, which can be time-consuming. To mitigate this, recent work aims to train 3D generation networks to produce assets rapidly [8,19,21,26,40,55,61]. While efficient, these methods often produce lower quality and less complex shapes due to limitations in training data.

While many works focus on text- or image-to-3D tasks, an important yet under-explored area lies in controllable text-to-3D generation—a gap that this work aims to address. In this work, we propose a new highly efficient controllable 3D generation pipeline that leverages the advantages of both lines of research mentioned in the previous paragraph. Motivated by the achievements of 2D ControlNet [69], an integral component of Stable-Diffusion [44], we propose MVControl, a multi-view variant. Given the critical role of multi-view capabilities in 3D generation, MVControl is designed to extend the success of 2D ControlNet into the multi-view domain. We adopt MVDream [50], a newly introduced multi-view diffusion network, as our foundational model. MVControl is subsequently crafted to collaborate with this base model, facilitating controllable text-to-multi-view image generation. Similar to the approach in [69], we freeze the weights of MVDream and solely focus on training the MVControl component. However, the

conditioning mechanism of 2D ControlNet, designed for single image generation, does not readily extend to the multi-view scenario, making it challenging to achieve view-consistency by directly applying its control network to interact with the base model. Additionally, MVDream is trained on an absolute camera system conflicts with the practical need for relative camera poses in our application scenario. To address these challenges, we introduce a simple yet effective conditioning strategy.

After training MVControl, we can leverage it to establish 3D priors for controllable text-to-3D asset generation. To address the extended optimization times of SDS-based methods, which can largely be attributed to the utilization of NeRF [38]-based implicit representations, we propose employing a more efficient explicit 3D representation, 3D Gaussian [24]. Specifically, we propose a multi-stage pipeline for handling textual prompts and condition images: 1) Initially, we employ our MVControl to generate four multi-view images, which are then inputted into LGM [55], a recently introduced large Gaussian reconstruction model. This step yields a set of coarse 3D Gaussians. 2) Subsequently, the coarse Gaussians undergo optimization using a hybrid diffusion guidance approach, combining our MVControl with a 2D diffusion model. We introduce SuGaR [17] regularization terms in this stage to improve the Gaussians’ geometry. 3) The optimized Gaussians are then transformed into a coarse Gaussian-bound mesh, for further refinement of both texture and geometry. Finally, a high-quality textured mesh is extracted from the refined Gaussian-bound mesh.

In summary, our main contributions are as follows:

- We introduce a novel network architecture designed for controllable fine-grain text-to-multi-view image generation. The model is evaluated across various condition types (edge, depth, normal, and scribble), demonstrating its generalization capabilities;
- We develop a multi-stage yet efficient 3D generation pipeline that combines the strengths of large reconstruction models and score distillation. This pipeline optimizes a 3D asset from coarse Gaussians to SuGaR, culminating in a mesh. Importantly, we are the first to explore the potential of a Gaussian-Mesh hybrid representation in the realm of 3D generation;
- Extensive experimental results showcase the ability of our method to produce high-fidelity multi-view images and 3D assets. These outputs can be precisely controlled using an input condition image and text prompt.

2 Related Work

Multi-view Diffusion Models. The success of text-to-image generation via large diffusion models inspires the development of multi-view image generation. Commonly adopted approach is to condition on a diffusion model by an additional input image and target pose [31, 32, 34]. Different from those methods, Chan et al. recently proposes to learn 3D scene representation from a single or multiple input images and then exploit a diffusion model for target novel view

image synthesis [10]. Instead of generating a single target view image, MVDiffusion [58] proposes to generate multi-view consistent images in one feed-forward pass. They build upon a pre-trained diffusion model to have better generalization capability. MVDream [50] introduces a method for generating consistent multi-view images from a text prompt. They achieve this by fine-tuning a pre-trained diffusion model using a 3D dataset. The trained model is then utilized as a 3D prior to optimize the 3D representation through Score Distillation Sampling. Similar work ImageDream [60] substitutes the text condition with an image. While prior works can generate impressive novel/multi-view consistent images, fine-grained control over the generated text-to-multi-view images is still difficult to achieve, as what ControlNet [69] has achieved for text-to-image generation. Therefore, we propose a multi-view ControlNet (i.e. MVControl) in this work to further advance diffusion-based multi-view image generation.

3D Generation Tasks. The exploration of generating 3D models can typically be categorized into two approaches. The first is the SDS-based optimization method, initially proposed by DreamFusion [42], which aims to extract knowledge for 3D generation through the utilization of pre-trained large image models. SDS-based method benefits from not requiring expansive 3D datasets and has therefore been extensively explored in subsequent works [11, 29, 52, 56, 63, 66, 70]. These works often provide valuable insights by developing more sophisticated score distillation loss functions, refining optimization strategies and/or employing better 3D representations, thereby further enhancing the quality of the generated objects. Despite the success achieved by these methods in generating high-fidelity 3D shapes based on textual descriptions, they usually require hours to complete the text-to-3D generation process. In the contrary, feed-forward 3D native methods can produce 3D assets within seconds after training on extensive 3D datasets [14]. Researchers have explored various 3D representations to achieve improved results, such as volumetric representation [6, 15, 28, 64], triangular mesh [13, 16, 54, 67], point cloud [1, 2], implicit neural representation [9, 12, 19, 26, 36, 41, 47, 62], as well as the recent 3D Gaussian [55]. Some of these methods can efficiently generate 3D models that fully satisfy the input conditions. While some methods efficiently generate 3D models that meet input conditions, 3D generative methods, unlike image generative modeling, struggle due to limited 3D training assets. This scarcity hinders their ability to produce high-fidelity and diverse 3D objects. Our method merges both approaches: generating a coarse 3D object with a feed-forward method conditioned on MVControl’s output, then refining it using SDS loss for the final representation.

Optimization based Mesh Generation. The current single-stage mesh generation method, such as MeshDiffusion [33], struggles to produce high-quality mesh due to its highly complex structures. To achieve high grade mesh in both geometry and texture, researchers often turn to multi-stage optimization-based methods [11, 29, 52]. These methods commonly use non-mesh intermediate representations that are easy to process, before transforming them back into meshes with mesh reconstruction methods, which can consume a long optimization time. DreamGaussian [56] refer to a more efficient representation, 3D Gaussian, to ef-

fectively reduce the training time. However, extracting meshes from millions of unorganized tiny 3D Gaussians remains challenging. LGM [55] presents a new mesh extraction method for 3D Gaussian but still relies on implicit representation. In contrast, we adopt a fully explicit representation, a hybrid of mesh and 3D Gaussian as proposed by SuGaR [17]. This approach enables us to achieve high-quality mesh generation within reasonable optimization time.

3 Method

We first review relevant methods, including the 2D ControlNet [69], score distillation sampling [42], Gaussian Splatting [24] and SuGaR [17] in Section 3.1. Then, we analyze the strategy of introducing additional spatial conditioning to MVDream by training a multi-view ControlNet in Section 3.2. Finally in Section 3.3, based on the trained multi-view ControlNet, we propose an efficient 3D generation pipeline, to realize the controllable text-to-3D generation via Gaussian-binded mesh and further textured mesh.

3.1 Preliminary

ControlNet. ControlNet [69] facilitates pretrained large text-to-image diffusion models to accommodate additional input conditions (e.g., canny edges, sketches, depth maps, etc.) alongside text prompts, thereby enabling precise control over generated content. It directly copies the structure and weights of SD’s encoder blocks and mid block as an additional module into the backbone model. The output feature maps of each layer of the ControlNet module’s encoder blocks and mid block are injected into the corresponding symmetric layers of the backbone model’s decoder blocks and mid block using 1x1 convolutions. To preserve the generation capability of the backbone model and ensure smooth training initiation, the 1x1 convolutions are initialized to zero.

Score Distillation Sampling. Score distillation sampling (SDS) [29, 42] utilizes a pretrained text-to-image diffusion model as a prior to guide the generation of text-conditioned 3D assets. Specifically, given a pretrained diffusion model ϵ_ϕ , SDS optimizes the parameters θ of a differentiable 3D representation (e.g., neural radiance field) using the gradient of the loss \mathcal{L}_{SDS} with respect to θ :

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

where $\mathbf{x} = g(\theta, c)$ is an image rendered by g under a camera pose c , $w(t)$ is a weighting function dependent on the timestep t , and z_t is the noisy image input to the diffusion model obtained by adding Gaussian noise ϵ to \mathbf{x} corresponding to the t -th timestep. The primary insight is to enforce the rendered image of the learnable 3D representation to adhere to the distribution of the pretrained diffusion model. In practice, the values of the timestep t and the Gaussian noise ϵ are randomly sampled at every optimization step.

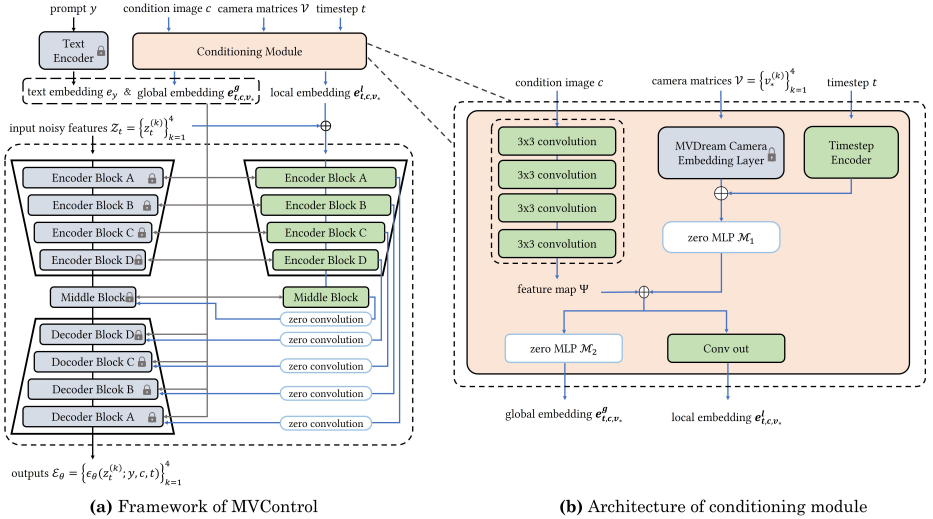


Fig. 2: Overview of proposed method. (a) MVControl consists of a frozen multi-view diffusion model and a trainable MVControl. (b) Our model takes care of all input conditions to control the generation process both locally and globally through a conditioning module. (c) Once MVControl is trained, we can exploit it to serve a hybrid diffusion prior for controllable text-to-3D content generation via SDS optimization procedure.

Gaussian Splatting and SuGaR. Gaussian Splatting [24] represents the scene as a collection of 3D Gaussians, where each Gaussian g is characterized by its center $\mu_g \in \mathbb{R}^3$ and covariance $\Sigma_g \in \mathbb{R}^{3 \times 3}$. The covariance Σ_g is parameterized by a scaling factor $s_g \in \mathbb{R}^3$ and a rotation quaternion $q_g \in \mathbb{R}^4$. Additionally, each Gaussian maintains opacity $\alpha_g \in \mathbb{R}$ and color features $c_g \in \mathbb{R}^C$ for rendering via splatting. Typically, color features are represented using spherical harmonics to model view-dependent effects. During rendering, the 3D Gaussians are projected onto the 2D image plane as 2D Gaussians, and color values are computed through alpha composition of these 2D Gaussians in front-to-back depth order. While the vanilla Gaussian Splatting representation may not perform well in geometry modeling, SuGaR [17] introduces several regularization terms to enforce flatness and alignment of the 3D Gaussians with the object surface. This facilitates extraction of a mesh from the Gaussians through Poisson reconstruction [23]. Furthermore, SuGaR offers a hybrid representation by binding Gaussians to mesh faces, allowing joint optimization of texture and geometry through backpropagation.

3.2 Multi-view ControlNet

Inspired by ControlNet in controlled text-to-image generation and recently released text-to-multi-view image diffusion model (e.g. MVDream), we aim to design a multi-view version of ControlNet (i.e. MVControl) to achieve controlled

text-to-multi-view generation. As shown in Fig. 2, we follow similar architecture style as ControlNet, i.e. a locked pre-trained MVDream and a trainable control network. The main insight is to preserve the learned prior knowledge of MVDream, while train the control network to learn the inductive bias with small amount of data. The control network consists of a conditioning module and a copy of the encoder network of MVDream. Our main contribution lies at the conditioning module and we will detail it as follows.

The conditioning module (Fig. 2b) receives the condition image c , four camera matrices $\mathcal{V}_* \in \mathbb{R}^{4 \times 4 \times 4}$ and timestep t as input, and outputs four local control embeddings e_{t,c,v_*}^l and global control embeddings e_{t,c,v_*}^g . The local embedding is then added with the input noisy latent features $\mathcal{Z}_t \in \mathbb{R}^{4 \times C \times H \times W}$ as the input to the control network, and the global embedding e_{t,c,v_*}^g is injected to each layer of MVDream and MVControl to globally control generation.

The condition image c (i.e. edge map, depth map etc.) is processed by four convolution layers to obtain a feature map Ψ . Instead of using the absolute camera pose matrices embedding of MVDream, we move the embedding into the conditioning module. To make the network better understand the spatial relationship among different views, the relative camera poses with respect to the condition image are used with the camera matrices \mathcal{V}_* . The experimental results also validate the effectiveness of the design. The camera matrices embedding is combined with the timestep embedding, and is then mapped to have the same dimension as the feature map Ψ by a zero-initialized module \mathcal{M}_1 . The sum of these two parts is projected to the local embedding e_{t,c,v_*}^l through a convolution layer.

While MVDream is pretrained with absolute camera poses, the conditioning module exploit relative poses as input. We experimentally find that the network hardly converges due to the mismatch of both coordinate frames. We therefore exploit an additional network \mathcal{M}_2 to learn the transformation and output a global embedding e_{t,c,v_*}^g to replace the original camera matrix embedding of MVDream and add on timestep embeddings of both MVDream and MVControl part, so that semantical and view-dependent features are injected globally.

3.3 Controllable 3D Textured Mesh Generation

In this section, we introduce our highly efficient multi-stage textured mesh generation pipeline: Given a condition image and corresponding description prompt, we first generate a set of coarse 3D Gaussians using LGM [55] with four multi-view images generated by our trained MVControl. Subsequently, the coarse Gaussians undergo refinement utilizing a hybrid diffusion prior, supplemented with several regularization terms aimed at enhancing geometry and facilitating coarse SuGaR mesh extraction. Both the texture and geometry of the extracted coarse SuGaR mesh are refined using 2D diffusion guidance under high resolution, culminating in the attainment of a textured mesh. The overall pipeline is illustrated in Fig. 3.

Coarse Gaussians Initialization. Thanks to the remarkable performance of LGM [55], the images generated by our MVControl model can be directly

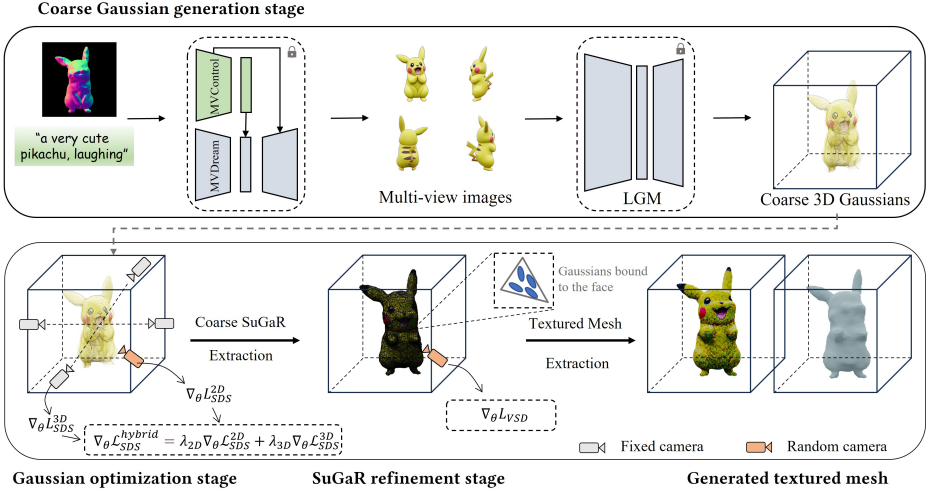


Fig. 3: Proposed 3D generation pipeline. The multi-stage pipeline can efficiently generate high-quality textured meshes starting from a set of coarse Gaussians generated by LGM, with the input being the multi-view images generated by our MVControl. In the second stage, we employ a 2D & 3D hybrid diffusion prior for Gaussian optimization. Finally, in the third stage, we calculate the VSD loss to refine the SuGaR representation.

inputted into LGM to produce a set of 3D Gaussians. However, owing to the low quality of the coarse Gaussians, transferring them directly to mesh, as done in the original paper, does not yield a satisfactory result. Instead, we further apply an optimization stage to refine the coarse Gaussians, with the starting point of optimization either initialized with all the coarse Gaussians’ features or solely their positions.

Gaussian-to-SuGaR Optimization. In this stage, we incorporate a hybrid diffusion guidance from a 2D diffusion model and our MVControl to enhance the optimization of coarse Gaussians θ . MVControl offers robust and consistent geometry guidance across four canonical views \mathcal{V}_* , while the 2D diffusion model contributes fine geometry and texture sculpting under other randomly sampled views $\mathcal{V}_r \in \mathbb{R}^{B \times 4 \times 4}$. Here, we utilize the DeepFloyd-IF base model [3] due to its superior performance in refining coarse geometry. Given a text prompt y and a condition image h , the hybrid SDS gradient $\nabla_{\theta} \mathcal{L}_{SDS}^{hybrid}$ can be calculated as:

$$\nabla_{\theta} \mathcal{L}_{SDS}^{hybrid} = \lambda_{2D} \nabla_{\theta} \mathcal{L}_{SDS}^{2D}(\mathbf{x}_r = g(\theta, \mathcal{V}_r); t, y) + \lambda_{3D} \nabla_{\theta} \mathcal{L}_{SDS}^{3D}(\mathbf{x}_* = g(\theta, \mathcal{V}_*); t, y, h), \quad (2)$$

where λ_1 and λ_2 are the strength of 2D and 3D prior respectively. To enhance the learning of geometry during the Gaussians optimization stage, we employ a Gaussian rasterization engine capable of rendering depth and alpha values [5]. Specifically, in addition to color images, depth \hat{d} and alpha \hat{m} of the scene are also rendered, and we estimate the surface normal \hat{n} by taking the derivative

of \hat{d} . Consequently, the total variation (TV) regularization terms [45] on these components, denoted as \mathcal{L}_{TV}^d and \mathcal{L}_{TV}^n , are calculated and incorporated into the hybrid SDS loss. Furthermore, as the input conditions are invariably derived from existing images, a foreground mask m_{gt} is generated during the intermediate process. Therefore, we compute the mask loss $\mathcal{L}_{mask} = \text{MSE}(\hat{m}, m_{gt})$ to ensure the sparsity of the scene. Thus, the total loss for Gaussian optimization is expressed as:

$$\mathcal{L}_{GS} = \mathcal{L}_{SDS}^{hybrid} + \lambda_1 \mathcal{L}_{TV}^d + \lambda_2 \mathcal{L}_{TV}^n + \lambda_3 \mathcal{L}_{mask}, \quad (3)$$

where $\lambda_{k=1,2,3}$ are the weights of depth TV loss, normal TV loss and mask loss respectively. Following the approach in [11], we alternately utilize RGB images or normal maps as input to the diffusion models when calculating SDS gradients. After a certain number of optimization steps N_1 , we halt the split and pruning of Gaussians. Subsequently, we introduce SuGaR regularization terms [17] as new loss terms to \mathcal{L}_{GS} to ensure that the Gaussians become flat and aligned with the object surface. This process continues for an additional N_2 steps, after which we prune all points whose opacity is below a threshold $\bar{\sigma}$.

SuGaR Refinement. Following the official pipeline of [17], we transfer the optimized Gaussians to a coarse mesh. For each triangle face, a set of new flat Gaussians is bound. The color of these newly bound Gaussians is initialized with the colors of the triangle vertices. The positions of the Gaussians are initialized with predefined barycentric coordinates, and rotations are defined as 2D complex numbers to constrain the Gaussians within the corresponding triangles. Different from the original implementation, we initialize the learnable opacities of the Gaussians with a large number, specifically 0.9, to facilitate optimization at the outset. Given that the geometry of the coarse mesh is nearly fixed, we replace the hybrid diffusion guidance with solely 2D diffusion guidance computed using Stable Diffusion [44] to achieve higher optimization resolution. Additionally, we employ Variational Score Distillation (VSD) [63] due to its superior performance in texture optimization. Similarly, we render the depth \hat{d}' and alpha \hat{m}' through the bound Gaussians. However, in contrast, we can directly render the normal map \hat{n}' using mesh face normals. With these conditions, we calculate the TV losses, \mathcal{L}_{TV}^d and \mathcal{L}_{TV}^n , and the mask loss \mathcal{L}'_{mask} similarly to the previous section. The overall loss for SuGaR refinement is computed as:

$$\mathcal{L}_{SuGaR} = \mathcal{L}_{VSD} + \lambda'_1 \mathcal{L}'_{TV}^d + \lambda'_2 \mathcal{L}'_{TV}^n + \lambda'_3 \mathcal{L}'_{mask}, \quad (4)$$

where $\lambda'_{k=1,2,3}$ represent the weights of the different loss terms, respectively.

4 Experiments

4.1 Implementation Details

Training Data. We employ the multi-view renderings from the publicly available large 3D dataset, Objaverse [14], to train our MVControl. Initially, we



Fig. 4: Comparison of Multi-view image generation w/ and w/o our MV-Control. MVDream generation results with and without our MVControl attached with edge map and normal map as input condition respectively.

preprocess the dataset by removing all samples with a CLIP-score lower than 22, based on the labeling criteria from [53]. This filtering results in approximately 400k remaining samples. Instead of utilizing the names and tags of the 3D assets, we employ the captions from [35] as text descriptions for our retained objects. Following the approach in [50], our network is trained on both 2D and 3D datasets. Specifically, we randomly sample images from the AES v2 subset of LAION [46] with a 30% probability during training to ensure the network retains its learned 2D image priors. Additionally, we sample from our curated 3D/multi-view image dataset with a 70% probability to learn 3D knowledge. The process of preparing different types of condition images will be detailed in our appendix.

Training details of MVControl. We initialize our network by leveraging the weights of the pretrained MVDream and ControlNet. All connections between the locked and trainable networks are initialized to zero. MVControl is trained at a resolution of 256, with a chosen batch size of 2560 images. The model undergoes fine-tuning for 50,000 steps using a conservative learning rate of $4e^{-5}$, employing the AdamW optimizer [25] on 8 Nvidia Tesla A100 GPUs. Similar to the approach in [69], we randomly drop the text prompt as empty with a 50% chance during training to facilitate classifier-free learning and enhance the model’s understanding of input condition images.

3D Generation. The coarse Gaussian optimization stage comprises a total of 3000 steps. During the initial 1500 steps, we perform straightforward 3D Gaussian optimization with split and prune. After step 1500, we cease Gaussian split and prune and instead introduce SuGaR [17] regularization terms to refine the scene. We set the number of nearest neighbors to 16, updating them every 50 steps. During training’s final phase, we prune all Gaussians with opacity below $\bar{\sigma} = 0.5$. We use $\lambda_{2D} = 0.1$ and $\lambda_{3D} = 0.01$ for 2D and 3D diffusion guidance, with resolutions of 512 and 256 respectively. For coarse SuGaR extraction, we target $2e5$ vertices, binding 6 Gaussians to each triangle. In the SuGaR refinement stage, we optimize the scene for 5000 steps with VSD loss [63].

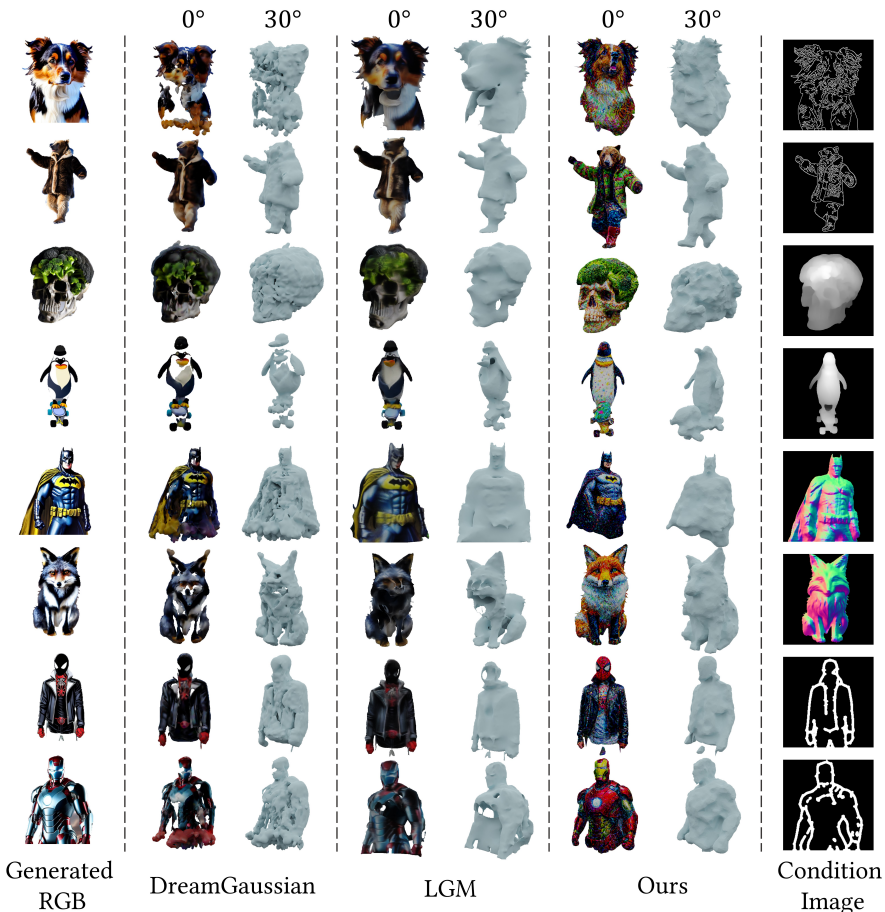


Fig. 5: Comparison with baseline 3D generation methods. Our method yields more delicate texture, and generates much better meshes than the compared methods.

4.2 Qualitative Comparisons

Multi-view Image Generation. To assess the controllability of our MVControl, we conduct experiments on MVDream both with and without MVControl attached. In the first case, MVDream fails to generate the correct contents according to the given prompt, producing a standing cat without clothes, which contradicts the prompt. In contrast, with the assistance of MVControl, it successfully generates the correct contents. The second case also demonstrates that our MVControl effectively controls the generation of MVDream, resulting in highly view-consistent multi-view images.

3D Gaussian-based Mesh Generation. Given that our 3D generation pipeline aims to produce textured mesh from 3D Gaussians, we compare our method with recent Gaussian-based mesh generation approaches, DreamGaussian [56] and

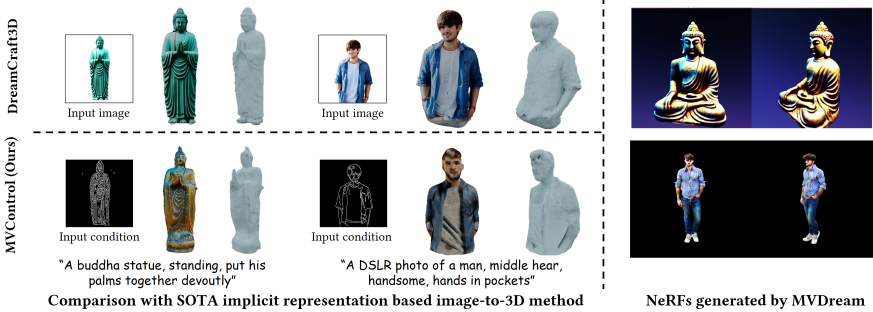


Fig. 6: Comparison with implicit representation based 3D generation methods. We compare our model with the state-of-the-art image-to-3D method DreamCraft3D [52] and our base model MVDream [50].

LGM [55], both of which can be conditioned on RGB images. For a fair comparison, we generate a 2D RGB image using the 2D ControlNet as the condition for the compared methods. As illustrated in Fig. 5, DreamGaussian struggles to generate the geometry for most of the examples, resulting in many broken and hollow areas in the generated meshes. LGM performs better than DreamGaussian, however, its extracted meshes lack details and still contain broken areas in some cases. In contrast, our method produces fine-grain meshes with more delicate textures, even without an RGB condition. Due to space limitations, the textual prompts are not provided in Fig. 5, and we will include them in our appendix. All the images in the figure are rendered using Blender.

Implicit Representation-based Mesh Generation. To provide a comprehensive evaluation of our model, we extend the application of MVCControl to 3D generation tasks based on implicit representations. We compare our approach with the state-of-the-art image-to-3D method, DreamCraft3D [52]. We achieve this by employing a straightforward coarse-to-fine optimization procedure. Initially, we generate a coarse Neus [59] model, followed by its transfer to a coarse DMTet [48] for further geometry optimization. The final stage focuses on texture refinement, with the geometry fixed. The hybrid diffusion guidance is utilized in all three stages, with varying weight terms. As illustrated in Fig. 6, our method is capable of generating 3D assets with geometry and texture comparable to DreamCraft3D, even in the absence of RGB signals. Furthermore, we conducted experiments on our base model, MVDream, using the same prompt. The results demonstrate that the prompt alone cannot precisely control the generation of MVDream (evidenced by the sitting buddha conflicting with the prompt) without the use of our MVCControl.

4.3 Quantitative Comparisons

In this section, we adopt CLIP-score [39] to evaluate both the compared methods and our method. Given that our task involves two conditions, namely the prompt and the condition image of the reference view, we calculate both image-text and

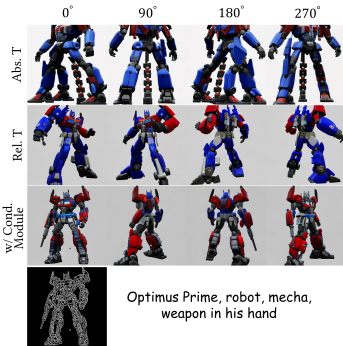


Fig. 7: Ablation study on conditioning module.



"A lovely rabbit sat in a bucket, showing itself, adorable"

input

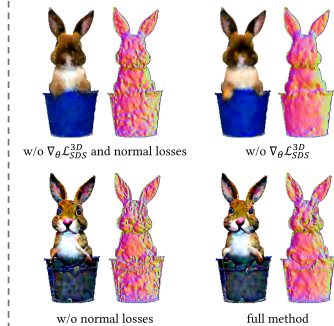


Fig. 8: Ablation study on Gaussian optimization stage.

image-image similarities. For each object, we uniformly render 36 surrounding views. The image-text similarity, denoted as CLIP-T, is computed by averaging the similarities between each view and the given prompt. Similarly, the image-image similarity, referred to as CLIP-I, is the mean similarity between each view and the reference view. The results, calculated for a set of 60 objects, are reported in Table 1. When employing our method, the condition type used for each object is randomly sampled from edge, depth, normal, and scribble map. Additionally, the RGB images for DreamGaussian and LGM are generated using 2D ControlNet with the same condition image and prompt.

	CLIP-T \uparrow	CLIP-I \uparrow
DreamGaussian [56]	0.200	0.847
LGM [55]	0.228	0.872
MVControl(Ours)	0.245	0.897

Table 1: Quantitative Comparison. We calculate both image-text and image-image similarity with CLIP score.

4.4 Ablation Study

Conditioning Module of MVControl. We evaluate the training of our model under three different settings to introduce camera condition: 1) we utilize the absolute (world) camera system (i.e., Abs. T) as MVDream [50] does, without employing our designed conditioning module (retaining the same setup as 2D ControlNet); 2) we adopt the relative camera system without employing the conditioning module; 3) we employ the complete conditioning module. The experimental results, depicted in Fig. 7, demonstrate that only the complete conditioning module can accurately generate view-consistent multi-view images that adhere to the descriptions provided by the condition image.

Hybrid Diffusion Guidance. We conduct ablation study on hybrid diffusion guidance utilized in the Gaussian optimization stage. As illustrated in Fig. 8 (top right), when excluding $\nabla_{\theta} \mathcal{L}_{SDS}^{3D}$ provided by our MVControl, the generated

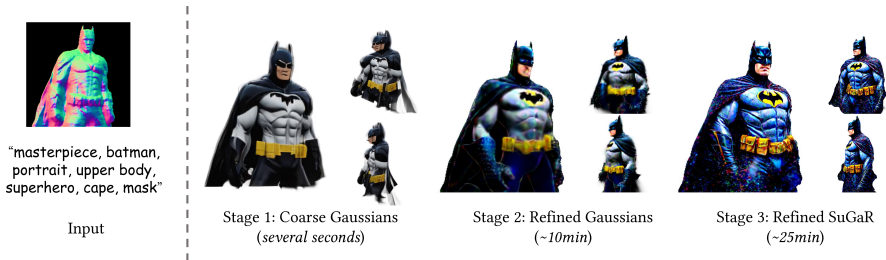


Fig. 9: Ablation study on multi-stage optimization. We conduct the ablation study under different optimization stages. Both Gaussian optimization and SuGaR refinement stage promote the quality largely. We also report the consumed time of every stage.

3D Gaussians lack texture details described in the given condition edge map. For instance, the face of the rabbit appears significantly blurrier without $\nabla_{\theta} \mathcal{L}_{SDS}^{3D}$.

Losses on Rendered Normal Maps. The normal-related losses in our method are alternately calculated using SDS loss with the normal map as input in stage 2, and the normal TV regularization term. We conduct experiments by dropping all of them in stage 2, and the results are illustrated in Fig. 8 (bottom left). Compared to our full method, the surface normal of 3D Gaussians deteriorates without the normal-related losses.

Multi-stage Optimization. We also assess the impact of different optimization stages, as shown in Fig. 9. Initially, in stage 1, the coarse Gaussians exhibit poor geometry consistency. However, after the Gaussian optimization stage, they become view-consistent, albeit with blurry texture. Finally, in the SuGaR refinement stage, the texture of the 3D model becomes fine-grained and of high quality.

5 Conclusion

In this work, we delve into the important yet under-explored field of controllable 3D generation. We present a novel network architecture, MVControl, for controllable text-to-multiview image generation. Our approach features a trainable control network that interacts with the base image diffusion model to enable controllable multi-view image generation. Once it is trained, our network offers 3D diffusion guidance for controllable text-to-3D generation using a hybrid SDS gradient alongside another 2D diffusion model. We propose an efficient multi-stage 3D generation pipeline using both feed-forward and optimization-based methods. Our pioneering use of SuGaR—an explicit representation blending mesh and 3D Gaussians—outperforms previous Gaussian-based mesh generation approaches. Experimental results demonstrate our method’s ability to produce controllable, high-fidelity text-to-multiview images and text-to-3D assets. Furthermore, tests across various conditions show our method’s generalization capabilities. We believe our network has broader applications in 3D vision and graphics beyond controllable 3D generation via SDS optimization.

Appendix

A Introduction

In this supplementary material, we offer additional details regarding our experimental setup and implementation. Subsequently, we present more qualitative results showcasing the performance and diversity of our method with various types of condition images as input.

B Implementation Detail

B.1 Training Data

Multi-view Images Dataset. As described in our main paper, we filter the number of objects in Objaverse [14] to approximately 400k. For each retained sample, we first normalize its scene bounding box to a unit cube centered at the world origin. Subsequently, we sample a random camera setting by uniformly selecting the camera distance between 1.4 and 1.6, the angle of Field-of-View (FoV) between 40 and 60 degrees, the degree of elevation between 0 and 30 degrees, and the starting azimuth angle between 0 and 360 degrees. Under the random camera setting, multi-view images are rendered at a resolution of 256 under 4 canonical views at the same elevation starting from the sampled azimuth. During training, one of these views is chosen as the reference view corresponding to the condition image. We repeat this procedure three times for each object.

Canny Edges. We apply the Canny edge detector [7] with random thresholds to all rendered images to obtain the Canny edge conditions. The lower threshold is randomly selected from the range [50, 125], while the upper threshold is chosen from the range [175, 250].

Depth Maps. We use the pre-trained depth estimator, Midas [43], to estimate the depth maps of rendered images.

Normal Maps. We compute normal map estimations of all rendered images by computing normal-from-distance on the depth values predicted by Midas.

User Scribble. We synthesize human scribbles from rendered images by employing an HED boundary detector [65] followed by a set of strong data augmentations, similar to those described in [69].

B.2 Training Details of MVControl

While our base model, MVDream [50], is fine-tuned from Stable Diffusion v2.1 [44], we train our multi-view ControlNet models from publicly available 2D ControlNet checkpoints¹ adapted to Stable Diffusion v2.1 for consistency. The models are trained on an 8×A100 node, where we have 160 (40×4) images on each GPU. With a gradient accumulation of 2 steps, we achieve a total batch size of 2560 images. We utilize a constant learning rate of 4×10^{-5} with 1000 steps of warm-up.

¹<https://huggingface.co/thibaud>

B.3 Implementation Details of 3D Generation

In our coarse Gaussian generation stage, the multi-view images are generated with our MVControl attached to MVDream using a 30-step DDIM sampler [51] with a guidance scale of 9 and a negative prompt "ugly, blurry, pixelated obscure, unnatural colors, poor lighting, dull, unclear, cropped, lowres, low quality, artifacts, duplicate". In the Gaussian optimization stage, the Gaussians are densified and pruned every 300 steps before step 1500. The 3D SDS ($\nabla_{\theta}\mathcal{L}_{SDS}^3$) is computed with a guidance scale of 50 using the CFG rescale trick [30], and $\nabla_{\theta}\mathcal{L}_{SDS}^2$ is computed with a guidance scale of 20. The $\nabla_{\theta}\mathcal{L}_{VSD}$ in the SuGaR refinement stage is computed with a guidance scale of 7.5. These score distillation terms also incorporate the aforementioned negative prompt. Our implementation is based on the threestudio project [18]. All testing images for condition image extraction are downloaded from civitai.com.

C Additional Qualitative Results

C.1 Diversity of MVControl

Similar to 2D ControlNet [69], our MVControl can generate diverse multi-view images with the same condition image and prompt. Some of the results are shown in Fig. 10.

C.2 Textured Meshes

We also provide additional generated textured mesh. The images under several views are shown in Figs. 11 to 14. Please refer to our [project page](#) for video and interactive mesh results.

D Textual Prompts for 3D Comparison

Here we provide the missing textual prompts in Fig. 5 of our main paper as below:

1. "RAW photo of A charming long brown coat dog, border collie, head of the dog, upper body, dark brown fur on the back,shelti,light brown fur on the chest,ultra detailed, brown eye"
2. "Wild bear in a sheepskin coat and boots, open-armed, dancing, boots, patterned cotton clothes, cinematic, best quality"
3. "Skull, masterpiece, a human skull made of broccoli"
4. "A cute penguin wearing smoking is riding skateboard, Adorable Character, extremely detailed"
5. "Masterpiece, batman, portrait, upper body, superhero, cape, mask"
6. "Ral-chrome, fox, with brown orange and white fur, seated, full body, adorable"
7. "Spiderman, mask, wearing black leather jacket, punk, absurdres, comic book"
8. "Marvel iron man, heavy armor suit, futuristic, very cool, slightly sideways, portrait, upper body"

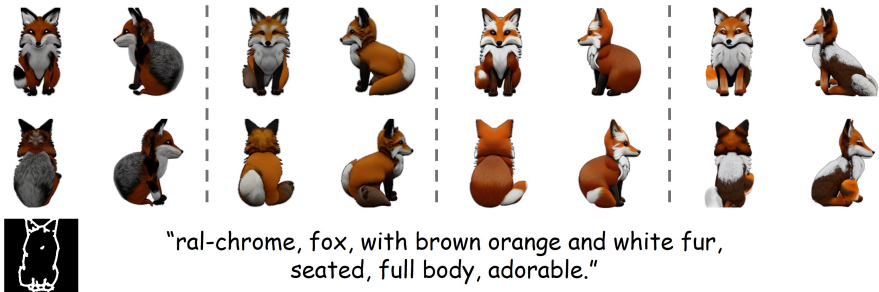
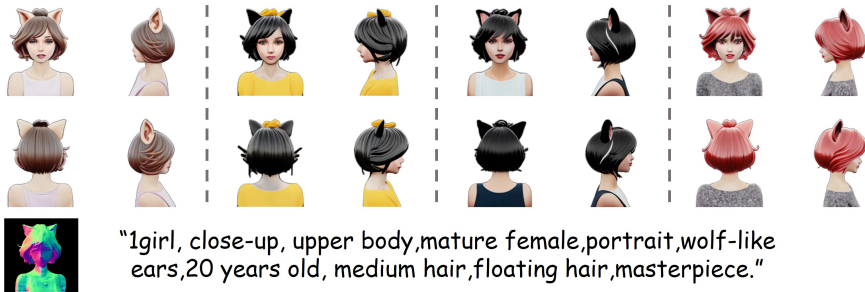
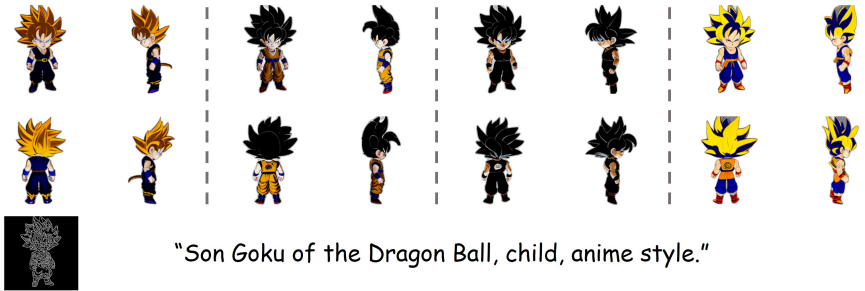
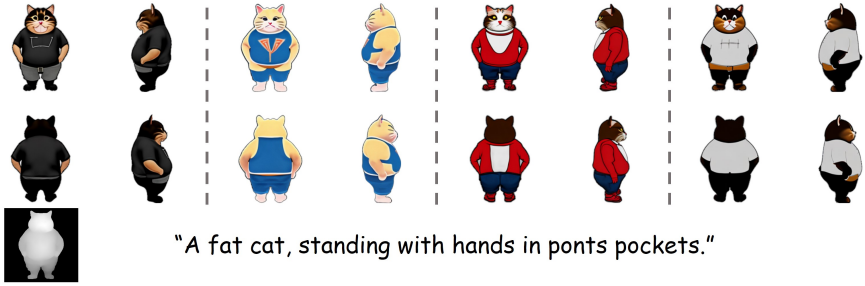


Fig. 10: Diverse generated multi-view images of MVCControl. Our MVCControl can generate diverse multi-view images with same condition image and prompt.



A lovely rabbit sat in a bucket, showing itself, adorable.



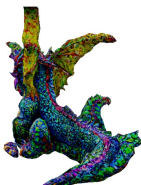
A light white statue of angel with two wings, the angel is sitting on her knees, praying with two hands put together.



A child astronaut, a cute children wearing space suit, portrait, upper body, digital art, adorable, highest details.



Arthur Morgan, cowboy, he is riding on a horse, lonely, red dead redemption 2 concept art.



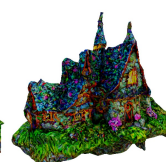
A baby dragon, digital art, masterpiece, its tail cocked to one side.



1girl, full body of virgin mary, holy, open-armed, in digital art illustration style, hyper detailed.

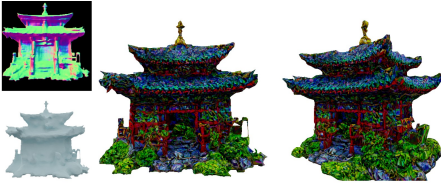


1girl, beautiful, photorealistic, portrait, upper body, middle hair, 20 years old, off-shoulder blouse, extremely detailed.



A witch house, gruesome, delicate, pointy roof, masterpiece, 3d asset.

Fig. 11: Additional qualitative results with Canny edge as input.



A Japanese temple, 3d asset, video game asset, concept art.



A very cute pikachu, real life, laughing, full body, adorable, highly detailed.



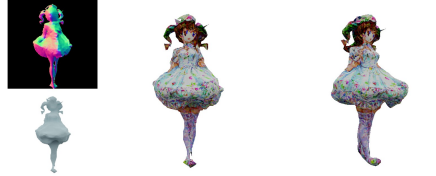
1girl, close-up, upper body, mature female, portrait, wolf-like ears, 20 years old, medium hair, floating hair, masterpiece.



A cute panda, sitting, adorable, extremely detailed.



Masterpiece, batman, portrait, upper body, superhero, cape, mask.



1girl, wearing white stockings and white dress, holding the hemline with two hands, double ponytail, small top hat, full body, anime style, beautiful.



1girl, full body armor, full face helmet, posed, extremely detailed.



A cat holding a baseball bat, full body, standing, very cute, masterpiece, best quality.

Fig. 12: Additional qualitative results with normal map as input.



Luxurious multi-layered burgers, delicious, extremely detailed.



A fat cat, standing with hands in pants pockets.



1girl, batgirl, mask, long hair, domino mask, belt, superhero, black bodysuit, slim, belt buckle, horns, bat print, curvy, narrow waist.



A cute cat wearing white suit, pink bowknot on its head, standing, full body, hello kitty shoes.



1girl, a girl is dancing ballet, two legs crossed, wearing a dance skirt, beautiful, delicate face.



Doll, full body, Obama, toy, best quality.



A cute penguin wearing smoking is riding skateboard, Adorable Character, extremely detailed.



Skull, masterpiece, a human skull made of broccoli.

Fig. 13: Additional qualitative results with depth map as input.



A bottle, blue and white porcelain, intricate patterns, extremely detailed.



A portrait of an alpaca, digital art, wearing black eye patch on face, anime style.



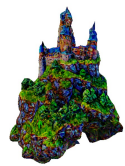
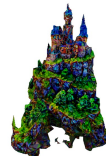
A white owl doll, metallic, polygonal, best quality.



A vanilla coffee ice cream cone with whipped cream, colorful, delicious, extremely detailed.



An astronaut, full body, spacesuit, walking pose.



A gorgeous castle on a hill, realistic materials, realistic textures, 3d rendering, game art.



A DSLR photo of 1girl, portrait, beautiful, stand, pose, front full body, laughing, very long hoodie, hands in hoodie pocket, air bangs, 20 years old



A crown with diamonds, golden, delicate.

Fig. 14: Additional qualitative results with user scribble as input.

References

1. Achlioptas, P., Diamanti, O., Mitliagkas, I., Guibas, L.: Learning representations and generative models for 3d point clouds. In: International conference on machine learning. pp. 40–49. PMLR (2018) [4](#)
2. Albert Pumarola, Stefan Popov, F.M.N., Ferrari, V.: C-Flow: Conditional generative flow models for images and 3D point clouds. In: CVPR (2020) [4](#)
3. Alex, S., Misha, K., Daria, B., Christoph, S., Ksenia, I., Nadiia, K.: Deepfloyd if: A modular cascaded diffusion model. <https://github.com/deep-floyd/IF/tree/develop> (2023) [8](#)
4. Armandpour, M., Zheng, H., Sadeghian, A., Sadeghian, A., Zhou, M.: Re-imagine the negative prompt algorithm: Transform 2d diffusion into 3d, alleviate janus problem and beyond. arXiv preprint arXiv:2304.04968 (2023) [2](#)
5. ashawkey: Differential gaussian rasterization. <https://github.com/ashawkey/diff-gaussian-rasterization> (2023) [8](#)
6. Brock, A., Lim, T., Ritchie, J.M., Weston, N.: Generative and discriminative voxel modeling with convolutional neural networks. arXiv preprint arXiv:1608.04236 (2016) [4](#)
7. Canny, J.: A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence (6), 679–698 (1986) [15](#)
8. Cao, Z., Hong, F., Wu, T., Pan, L., Liu, Z.: Large-vocabulary 3d diffusion model with transformer. arXiv preprint arXiv:2309.07920 (2023) [2](#)
9. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., Mello, S.D., Gallo, O., Guibas, L., Tremblay, J., Khamis, S., Karras, T., Wetzstein, G.: Efficient Geometry Aware 3D Generative Adversarial Networks. In: CVPR (2022) [4](#)
10. Chan, E.R., Nagano, K., Chan, M.A., Bergman, A.W., Park, J.J., Levy, A., Aittala, M., Mello, S.D., Karras, T., Wetzstein, G.: Generative novel view synthesis with 3D aware diffusion models. In: ICCV (2023) [4](#)
11. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. arXiv preprint arXiv:2303.13873 (2023) [2](#), [4](#), [9](#)
12. Chen, Z., Zhang, H.: Learning implicit fields for generative shape modeling. In: CVPR (2019) [4](#)
13. Dario Pavllo, Jonas Kohler, T.H., Lucchi, A.: Learning generative models of textured 3D meshes from real-world images. In: ICCV (2021) [4](#)
14. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., Vanderbilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13142–13153 (2023) [4](#), [9](#), [15](#)
15. Gadelha, M., Maji, S., Wang, R.: 3d shape induction from 2d views of multiple objects. In: 2017 International Conference on 3D Vision (3DV). pp. 402–411. IEEE (2017) [4](#)
16. Gao, L., Yang, J., Wu, T., Yuan, Y., Fu, H., Lai, Y., Zhang, H.: SDM-Net: Deep generative network for structured deformable mesh. In: ACM TOG (2019) [4](#)
17. Guédon, A., Lepetit, V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. arXiv preprint arXiv:2311.12775 (2023) [3](#), [5](#), [6](#), [9](#), [10](#)
18. Guo, Y.C., Liu, Y.T., Shao, R., Laforte, C., Voleti, V., Luo, G., Chen, C.H., Zou, Z.X., Wang, C., Cao, Y.P., Zhang, S.H.: threestudio: A unified framework for 3d content generation. <https://github.com/threestudio-project/threestudio> (2023) [16](#)

19. 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) [2](#), [4](#)
20. Huang, Y., Wang, J., Shi, Y., Qi, X., Zha, Z.J., Zhang, L.: Dreamtime: An improved optimization strategy for text-to-3d content creation. arXiv preprint arXiv:2306.12422 (2023) [2](#)
21. Jun, H., Nichol, A.: Shap-e: Generating conditional 3d implicit functions. arXiv preprint arXiv:2305.02463 (2023) [2](#)
22. Katzir, O., Patashnik, O., Cohen-Or, D., Lischinski, D.: Noise-free score distillation. arXiv preprint arXiv:2310.17590 (2023) [2](#)
23. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Proceedings of the fourth Eurographics symposium on Geometry processing. vol. 7, p. 0 (2006) [6](#)
24. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics (ToG) **42**(4), 1–14 (2023) [2](#), [3](#), [5](#), [6](#)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) [10](#)
26. Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., Bi, S.: Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. arXiv preprint arXiv:2311.06214 (2023) [2](#), [4](#)
27. Li, W., Chen, R., Chen, X., Tan, P.: Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. arXiv preprint arXiv:2310.02596 (2023) [2](#)
28. Li, X., Dong, Y., Peers, P., Tong, X.: Synthesizing 3d shapes from silhouette image collections using multi-projection generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5535–5544 (2019) [4](#)
29. 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) [2](#), [4](#), [5](#)
30. Lin, S., Liu, B., Li, J., Yang, X.: Common diffusion noise schedules and sample steps are flawed. arXiv preprint arXiv:2305.08891 (2023) [16](#)
31. 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) [2](#), [3](#)
32. 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) [2](#), [3](#)
33. Liu, Z., Feng, Y., Black, M.J., Nowrouzezahrai, D., Paull, L., Liu, W.: Meshdiffusion: Score-based generative 3d mesh modeling. arXiv preprint arXiv:2303.08133 (2023) [4](#)
34. Long, X., Guo, Y.C., Lin, C., Liu, Y., Dou, Z., Liu, L., Ma, Y., Zhang, S.H., Habermann, M., Theobalt, C., et al.: Wonder3d: Single image to 3d using cross-domain diffusion. arXiv preprint arXiv:2310.15008 (2023) [2](#), [3](#)
35. Luo, T., Rockwell, C., Lee, H., Johnson, J.: Scalable 3d captioning with pretrained models. arXiv preprint arXiv:2306.07279 (2023) [10](#)
36. Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A.: Occupancy Networks: Learning 3D reconstruction in function space. In: CVPR (2019) [4](#)

37. Metzger, G., Richardson, E., Patashnik, O., Giryas, R., Cohen-Or, D.: Latent-nerf for shape-guided generation of 3d shapes and textures. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12663–12673 (2023) [2](#)
38. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* **65**(1), 99–106 (2021) [2](#), [3](#)
39. Mohammad Khalid, N., Xie, T., Belilovsky, E., Popa, T.: Clip-mesh: Generating textured meshes from text using pretrained image-text models. pp. 1–8 (2022) [12](#)
40. Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., Chen, M.: Point-e: A system for generating 3d point clouds from complex prompts. arXiv preprint arXiv:2212.08751 (2022) [2](#)
41. Park, J.J., Florence, P., Straub, J., Newcombe, R., Lovegrove, S.: DeepSDF: Learning continuous signed distance functions for shape representation. In: CVPR (2019) [4](#)
42. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. In: ICLR (2023) [2](#), [4](#), [5](#)
43. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12179–12188 (2021) [15](#)
44. 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) [2](#), [9](#), [15](#)
45. Rudin, L.I., Osher, S.: Total variation based image restoration with free local constraints. In: Proceedings of 1st international conference on image processing. vol. 1, pp. 31–35. IEEE (1994) [9](#)
46. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., 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) [10](#)
47. Schwarz, K., Sauer, A., Niemeyer, M., Liao, Y., , Geiger, A.: VoxGRAF: Fast 3D-aware image synthesis with sparse voxel grids (2022) [4](#)
48. 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) [2](#), [12](#)
49. Shi, R., Chen, H., Zhang, Z., Liu, M., Xu, C., Wei, X., Chen, L., Zeng, C., Su, H.: Zero123++: a single image to consistent multi-view diffusion base model. arXiv preprint arXiv:2310.15110 (2023) [2](#)
50. 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) [2](#), [4](#), [10](#), [12](#), [13](#), [15](#)
51. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020) [16](#)
52. Sun, J., Zhang, B., Shao, R., Wang, L., Liu, W., Xie, Z., Liu, Y.: Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. arXiv preprint arXiv:2310.16818 (2023) [2](#), [4](#), [12](#)
53. Sun, Q., Li, Y., Liu, Z., Huang, X., Liu, F., Liu, X., Ouyang, W., Shao, J.: Unig3d: A unified 3d object generation dataset. arXiv preprint arXiv:2306.10730 (2023) [10](#)
54. Tan, Q., Gao, L., Lai, Y., Xia, S.: Variational autoencoders for deforming 3D mesh models. In: CVPR (2018) [4](#)

55. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: Lgm: Large multi-view gaussian model for high-resolution 3d content creation. arXiv preprint arXiv:2402.05054 (2024) [2](#), [3](#), [4](#), [5](#), [7](#), [12](#), [13](#)
56. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. arXiv preprint arXiv:2309.16653 (2023) [2](#), [4](#), [11](#), [13](#)
57. Tang, J., Wang, T., Zhang, B., Zhang, T., Yi, R., Ma, L., Chen, D.: Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. arXiv preprint arXiv:2303.14184 (2023) [2](#)
58. Tang, S., Zhang, F., Chen, J., Wang, P., Furukawa, Y.: Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion (2023) [4](#)
59. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. arXiv preprint arXiv:2106.10689 (2021) [2](#), [12](#)
60. Wang, P., Shi, Y.: Imagedream: Image-prompt multi-view diffusion for 3d generation. arXiv preprint arXiv:2312.02201 (2023) [4](#)
61. Wang, P., Tan, H., Bi, S., Xu, Y., Luan, F., Sunkavalli, K., Wang, W., Xu, Z., Zhang, K.: Pflrm: Pose-free large reconstruction model for joint pose and shape prediction. arXiv preprint arXiv:2311.12024 (2023) [2](#)
62. Wang, T., Zhang, B., Zhang, T., Gu, S., Bao, J., Baltrusaitis, T., Shen, J., Chen, D., Wen, F., Chen, Q., Guo, B.: Rodin: A generative model for sculpting 3D digital Avatars using diffusion. In: CVPR (2023) [4](#)
63. 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. arXiv preprint arXiv:2305.16213 (2023) [2](#), [4](#), [9](#), [10](#)
64. Wu, J., Zhang, C., Xue, T., Freeman, B., Tenenbaum, J.: Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in neural information processing systems* **29** (2016) [4](#)
65. Xie, S., Tu, Z.: Holistically-nested edge detection. In: Proceedings of the IEEE international conference on computer vision. pp. 1395–1403 (2015) [15](#)
66. Yi, T., Fang, J., Wu, G., Xie, L., Zhang, X., Liu, W., Tian, Q., Wang, X.: Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. arXiv preprint arXiv:2310.08529 (2023) [4](#)
67. Youwang, K., Ji-Yeon, K., Oh, T.H.: CLIP-Actor: text driven recommendation and stylization for animating human meshes. In: ECCV (2022) [4](#)
68. Yu, X., Guo, Y.C., Li, Y., Liang, D., Zhang, S.H., Qi, X.: Text-to-3d with classifier score distillation. arXiv preprint arXiv:2310.19415 (2023) [2](#)
69. 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) [2](#), [4](#), [5](#), [10](#), [15](#), [16](#)
70. Zhu, J., Zhuang, P., Koyejo, S.: Hifa: High-fidelity text-to-3d generation with advanced diffusion guidance. In: The Twelfth International Conference on Learning Representations (2023) [2](#), [4](#)