Direct2.5: Diverse Text-to-3D Generation via Multi-view 2.5D Diffusion

Yuanxun Lu[1] * Jingyang Zhang[2] Shiwei Li[2] Tian Fang[2] David McKinnon[2]
Yanghai Tsin[2] Long Quan[3] Xun Cao[1] Yao Yao[1] †

[1]Nanjing University
luyuanxun@smail.nju.edu.cn, {caoxun,yaoyao}@nju.edu.cn

[2]Apple
{jingyang_zhang,shiwei,fangtian,dmckinnon,ytsin}@apple.com

[3]The Hong Kong University of Science and Technology
quan@cse.ust.hk

Abstract

*Recent advances in generative AI have unveiled significant potential for the creation of 3D content. However, current methods either apply a pre-trained 2D diffusion model with the time-consuming score distillation sampling (SDS), or a direct 3D diffusion model trained on limited 3D data losing generation diversity. In this work, we approach the problem by employing a multi-view 2.5D diffusion fine-tuned from a pre-trained 2D diffusion model. The multi-view 2.5D diffusion directly models the structural distribution of 3D data, while still maintaining the strong generalization ability of the original 2D diffusion model, filling the gap between 2D diffusion-based and direct 3D diffusion-based methods for 3D content generation. During inference, multi-view normal maps are generated using the 2.5D diffusion, and a novel differentiable rasterization scheme is introduced to fuse the almost consistent multi-view normal maps into a consistent 3D model. We further design a normal-conditioned multi-view image generation module for fast appearance generation given the 3D geometry. Our method is a one-pass diffusion process and does not require any SDS optimization as post-processing. We demonstrate through extensive experiments that, our direct 2.5D generation with the specially-designed fusion scheme can achieve diverse, mode-seeking-free, and high-fidelity 3D content generation in only **10 seconds**. Project page: https://nju-3dv.github.io/projects/direct25.*

1. Introduction

Creating 3D content from generative models has become a heated research topic in the past year, which is key to a variety of downstream applications, including game and film industries, autonomous driving simulation, and virtual reality. Specifically, DreamFusion [28] was proposed to optimize a neural radiance field (NeRF) [24] using a pre-trained 2D text-to-image diffusion model and the score distillation sampling (SDS) technique, showing promising results for text-to-3D generation of arbitrary objects without any 3D data. However, the indirect 3D probability distribution modeling inevitably deteriorates the final generation quality. For example, it has been reported in DreamFusion and its follow-ups [6, 15, 39, 42] that the overall generation success rate is low and the multi-face Janus problem exists.

Another line of work focuses on direct 3D generation by training on large-scale 3D data. For example, [22, 26] apply the probabilistic diffusion model for point cloud generation and [12, 34] model the denoise diffusion process on signed distance field (SDF). These methods usually apply a specific 3D representation and train the denoise diffusion on such representation using a specific 3D dataset, e.g., ShapeNet [3], and show high-quality generation results on objects similar to the training set. However, the scale of the current 3D dataset is still too small when compared with the text-image data [32]. Even with the largest 3D dataset [7] available, it is still challenging to train a 3D diffusion model for diverse text-to-3D generation.

In this work, we instead extend existing text-to-2D models to a denoising diffusion process on multi-view 2.5D depth/normal data. Compared with full 3D representations such as 3D point clouds or meshes, 1) 2.5D information such as depth or normal are much easier to capture or col-

*This project was performed during Yuanxun Lu's internship at Apple.
†Corresponding Author

lect (e.g., depth provided by active sensors); 2) the depth and normal maps perfectly align with the image data, making it possible to adapt and fine-tune a 2.5D model from a pre-trained 2D RGB model. In order to construct full 3D models, 2.5D maps viewed from multiple perspectives are necessary. Therefore, the target diffusion model should be capable of generating multi-view images with content consistency. In practice, we fine-tune existing text-to-image diffusion models on multi-view 2.5D renderings from the Objaverse dataset [7]. On the one hand, the models are adapted to 2.5D information. On the other hand, joint multi-view distribution is captured with the help of structural modification of injecting multi-view information to the self-attention layers. During inference, multi-view images are generated synchronously by common schedulers like DDIM [35], which are then fused directly into a mesh by differentiable rasterization. The whole generation process completes in seconds, which is significantly faster than SDS-based methods that typically take 30 minutes. The system is extensively evaluated with complex text prompts and compared with both SDS-based and direct 3D generation methods, demonstrating the capability of generating 3D textured meshes with complex geometry, diversity, and high fidelity.

To summarize, major contributions of the paper include:

- We propose to approach the 3D generation task by training a multi-view 2.5D diffusion model, which explicitly models the 3D geometry distribution while inheriting a strong generalization ability of the large-scale pre-trained 2D image diffusion.
- We introduce an efficient differentiable rasterization scheme to optimize a textured mesh directly from the multi-view normal maps and RGB images.
- We carefully design a generation pipeline that achieves diverse, mode-seeking-free, and high-fidelity 3D content generation in only 10 seconds.

2. Related Work

2.1. 3D Generation by Score Distillation

Score Distillation [28, 39] is one of the most popular method recently for 3D Generation by pre-trained 2D diffusion models. It distillates the knowledge of image denoising to the optimization process of differentiable rendering systems so that randomly rendered views are gradually refined to describe the input text prompt. There are fundamental problems: 1) 2D diffusion models are not 3D-aware, and the generated samples have multi-face problem as a result; 2) Each optimization step requires single forward of the denoising UNet, making the whole process time consuming; 3) High guidance scale of prompts is preferred for better convergence, which leads to over-saturation of appearance; 4) the optimization is mode-seeking, losing the strong diversity of 2D diffusion model. Follow up works are proposed to solve some of them, but not all. Zero-1-to-3 [16] fine-tunes the 2D diffusion model with multi-view dataset to grant the ability of perspective control and mitigate the problem 1 in image-to-3D task. ProlificDreamer [42] mitigate problem 3 and 4 by utilizing a KL-divergence loss to perform sampling instead of mode-seeking, at the cost of higher time complexity. In this work, we do not apply score distillation and completely separate diffusion process and 3D model optimization. The diffusion can be scheduled and conditioned normally, so that the results have diversity and realistic color. And the 3D model optimization operates on explicit representation so can be finished quickly.

2.2. Direct 3D Diffusion

Fast 3D generation can be achieved by training a direct 3D diffusion model with 3D dataset. One key problem is to choose the 3D representation and design a special encoder/decoder for it. There are some early attempts to train direct 3D models for point cloud [22, 26, 45, 48], mesh [18] and implicit representation like NeRF or SDF [5, 10, 12, 34]. However, they are trained on the limited datasets like ShapeNet [3] which have rather small data size, geometry complexity or category diversity. Recent 3D datasets such as Objaverse [7] dramatically improve the state-of-the-art of 3D dataset, but is still limited compared to 2D image-caption datasets for training 2D diffusion models. In this work, we still use 2D neural network to deal with 2.5D maps, and thus we can perform fine-tuning on existing 2D diffusion models so as to inherit their strong generalization.

2.3. Multi-view Diffusion

Generating multi-view images simultaneously is another strategy to bring 3D-awareness to 2D diffusion models. Two key modifications are proposed to achieve this: 1) Information from other views are concatenated with the current view as keys and queries in the self-attention layers. The gathered information can be from the single projection [36], epipolar lines [17, 37] or all the pixels [33]; 2) The model is fine-tuned on multi-view renderings from 3D dataset like Objaverse [7]. To construct 3D models, previous works either use SDS [33], which is still time-consuming, or image-based reconstruction systems like NeuS [17, 19, 40], which requires at least 10 views to produce reasonable reconstructions. Similar to JointNet [47] which explores the 2.5D domain, we choose to generate multi-view 2.5D maps like normal, so that we can use SDS-free reconstruction while still keep small view numbers.

3. Method

In this section, we introduce our multi-view 2.5D diffusion system, which synchronously generates multi-view 2.5D

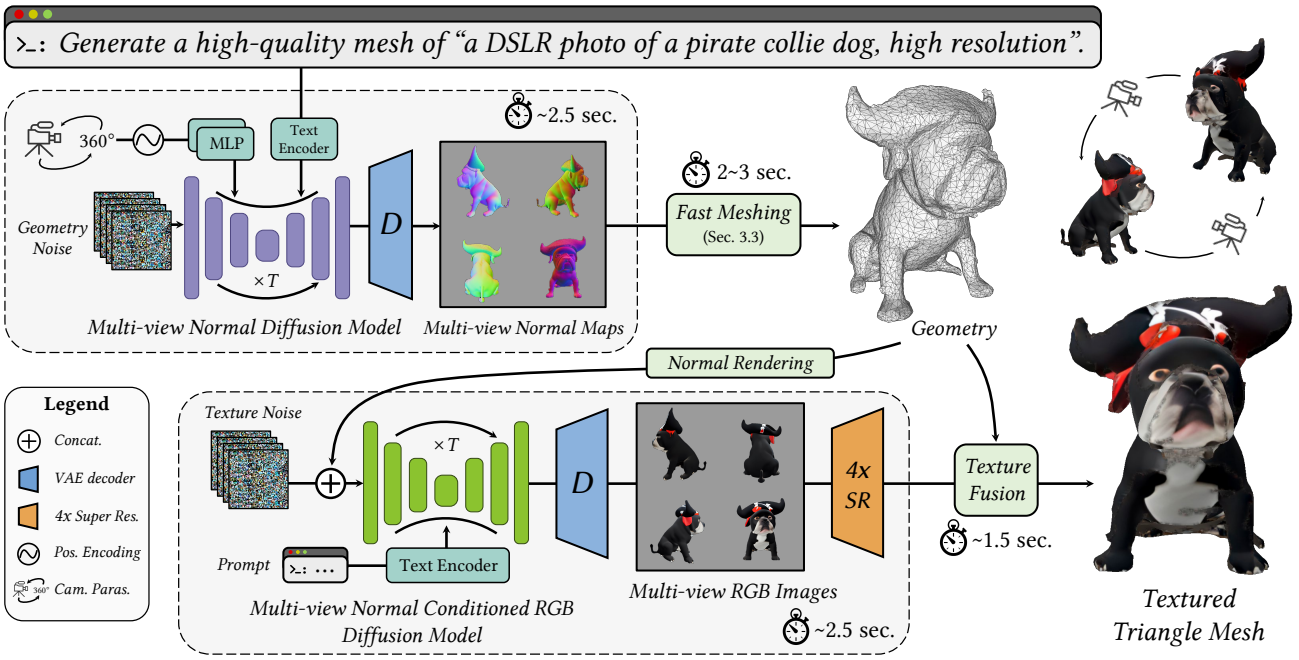


Figure 1. Overview of our text-to-3D content generation system. The generation is a two-stage process, first generating geoemtry and then appearance. Specifically, the system is composed of the following steps: 1) a single denoising process to simultaneously generate 4 normal maps; 2) fast mesh optimization by differentiable rasterization; 3) a single denoising process to generate 4 images conditioned on rendered normal maps; 4) texture construction from multi-view images. The whole generation process only takes 10 seconds.

geometry images, i.e., normal maps, and corresponding texture maps given a text prompt as input for 3D content generation (Fig. 1). Our method is efficient enough to generate various results in only 10 seconds. In Sec. 3.1, we first briefly review the 2D diffusion model and formulate the multi-view 2.5D adaptation. We then illustrate the cross-view attention which enhances the multi-view consistency in Sec. 3.2. In Sec. 3.3, we describe how to produce the final 3D model from generated 2.5D geometry images, and finally in Sec. 3.4, we demonstrate how to synthesize the texture maps given the generated normal maps, and construct the high-quality final textured triangle mesh.

3.1. Diffusion Models and 2.5D Adaptation

Diffusion models learn a conversion from an isotropic Gaussian distribution to the target distribution (e.g. image spaces) via iterative denoising operations. We build our system on latent diffusion models (LDM), which contains a variational autoencoder (VAE) including an encoder and a decoder, a denoising network, and a condition input encoder. Compared to original diffusion models, LDM conducts the whole diffusion process in the latent image space and greatly improves efficiency and quality. Specifically, during the forward process, a noisy latent at time t is sampled in the latent space and is gradually degraded by noise which makes it indistinguishable from the Gaussian noise, while the denoising process reverses the process, which iteratively predicts and remove the noise to get the real images.

In this work, we extend 2D text-to-image diffusion models to generate multi-view geometry images. By fine-tuning a pre-trained 2D diffusion model using our 2.5D image dataset, we are able to inherit the generalization and also obtain the expressive generation ability for multi-view 2.5D geometry images. Let $(\boldsymbol{\mathcal{X}}, c)$ be 3D data with caption from training dataset, $x_i \in \boldsymbol{\mathcal{X}}$ be multi-view renderings, $x_{i,t}$ be views corrupted by independent noise $\epsilon_i \in \boldsymbol{\mathcal{E}}$ at time t. The denoising neural network ϵ_θ is trained by

$$L = \mathbb{E}_{(\boldsymbol{\mathcal{X}},c);\boldsymbol{\mathcal{E}}\sim N(0,1);t} \sum_{x_i\in\boldsymbol{\mathcal{X}};\epsilon_i\in\boldsymbol{\mathcal{E}}} \|\epsilon_i - \epsilon_\theta(x_{i,t}, c, t)\|_2^2. \tag{1}$$

3.2. Cross-view Attention

Before fine-tuning, the multiple images generated from the base model for the same text prompt are not guaranteed to describe the same object because they are initiated from different noise maps and are denoised independently. We use a solution similar to [33]: we add data communication among the diffusion processes and fine-tune the model on multi-view image dataset to learn multi-view conditioning. Implementation-wise, we synchronize all the diffusion processes. When the calculation reaches a self-attention layer, we gather all the intermediate results as queries and values instead of just using the results from the current branch. Because images are treated as sequential inputs, the additional

Algorithm 1: *Multi-view Geometry Optimization*

Input: Multi-view normal maps I_i and camera parameters π_i, where $i \in \{0, 1, 2, 3\}$
Output: $M = (V, F)$ output triangle mesh
Parameters:
T: max number of optimization iterations
$\lambda_\alpha, \lambda_{nc}$: weights for alpha and normal consistency loss

$V_{occ} \leftarrow$ InitOccupancyVolume
for $i \in \{0, 1, 2, 3\}$ **do**
  Compute alpha mask $\alpha_i \leftarrow$ thresholding(I_i)
  Update $V_{occ} \leftarrow$ SpaceCarving(α_i, π_i)
end

$M \leftarrow$ MarchingCubes(V_{occ})
$M \leftarrow$ MeshSimplification(M)

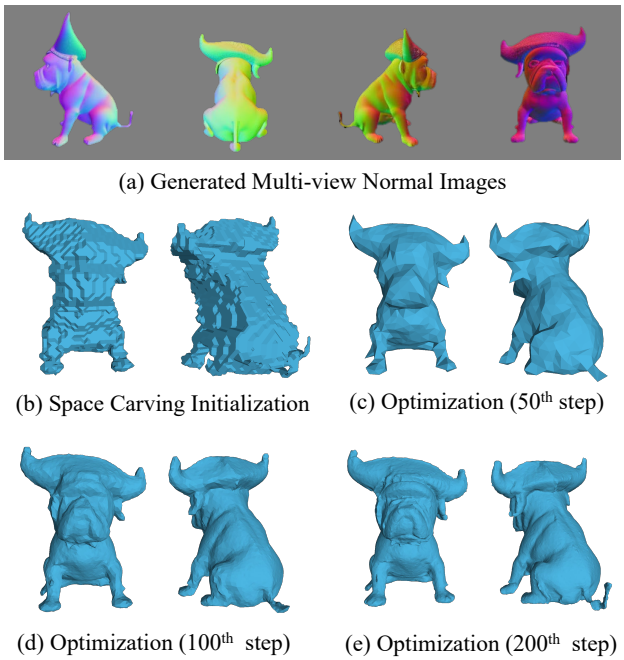
for *iter* $\leftarrow T$ **do**
  $\hat{I}, \hat{\alpha} \leftarrow$ DifferentiableRender(M, π)
  $loss \leftarrow \mathcal{L}_n(I, \hat{I}) + \lambda_\alpha \mathcal{L}_\alpha(\alpha, \hat{\alpha}) + \lambda_{nc}\mathcal{L}_{nc}(M)$
  Optimize$(loss)$
  $M \leftarrow$ Remesh(M)
end

information can be simply concatenated together without introducing more trainable parameters. This architecture ensures that the diffusion processes are mutually conditioned, which serves as a structural prerequisite for multi-view consistent generation.

3.3. Explicit Multi-view 2.5D Fusion

There are various approaches available for constructing a 3D model from multi-view observations. Among them, image-based 3D reconstruction methods such as multi-view stereo [9, 43, 44, 46] or NeRF [1, 8, 24, 25] requires at least 10 images for high-fidelity reconstruction, which pose significant computational challenges in multi-view diffusion scenarios. However, by taking benefits from 2.5D information, one could effectively reduce this requirement. In practice, we generate 4 normal maps aligned with world coordinates from different viewpoints (front, left, right, and back). To fuse these observations into a triangle mesh, we explore the insight of geometry optimization from an initialized mesh via differentiable rasterization. This optimization, which is independent of neural network inference, achieves convergence rapidly within seconds (see Alg. 1).

Space Carving Initialization. A simplistic and straightforward approach would be to initialize the shape using basic geometric primitives like spheres and cubes and optimize. However, this often introduces significant challenges during the latter geometry optimization, particularly when the target shape's topology diverges significantly from these elementary forms. To tackle this challenge, we employ the space carving algorithm [13] for shape topology initialization. Besides, it also provides a good initialization for latter



(a) Generated Multi-view Normal Images

(b) Space Carving Initialization (c) Optimization (50th step)

(d) Optimization (100th step) (e) Optimization (200th step)

Figure 2. Illustration of explicit geometry optimization. (a) is the generated normal images given a prompt "a DSLR photo of a pirate collie dog, high resolution". (b) shows the space carving initialization results mesh in the front and side views. (c), (d), (e) present the intermediate optimization states at 50, 100, 200 steps, separately. As shown, 200 steps are enough to reconstruct the fine details like the skin folds of the dog's face and the thin dog tail.

geometry optimization. Fig. 2 (a) shows the space carving results. Specifically, this process begins by segregating the background normal maps through a simple value thresholding. Subsequently, a volume in the interested space is created, and each voxel is projected onto the images using the camera parameters, determining whether the corresponding pixel is part of the object or the background. By gathering all projections under different views, we construct an occupancy volume, in which a voxel's occupancy is set to 0 (indicating emptiness) if all of its projections belong to the background, and 1 (indicating occupancy) otherwise. Finally, we apply the marching cubes [20] on the occupancy volume to extract the zero level-set surface to form the initialized shape. This technique not only effectively preserves the topology, but also provides a rough shape estimation generated from the multi-view normal images.

Optimization via Differentiable Rasterization. Once we have obtained the initialized geometry, we further refine the mesh details based on observational data. This refinement is mathematically formulated as an optimization problem, targeting the triangle triangle vertices V and faces F. As illustrated in Alg. 1 and Fig. 2, we first simply the marching cube-generated mesh to a lower face number, which is found to help accelerate and improve the optimization. In each optimization step, we optimize the model by minimiz-

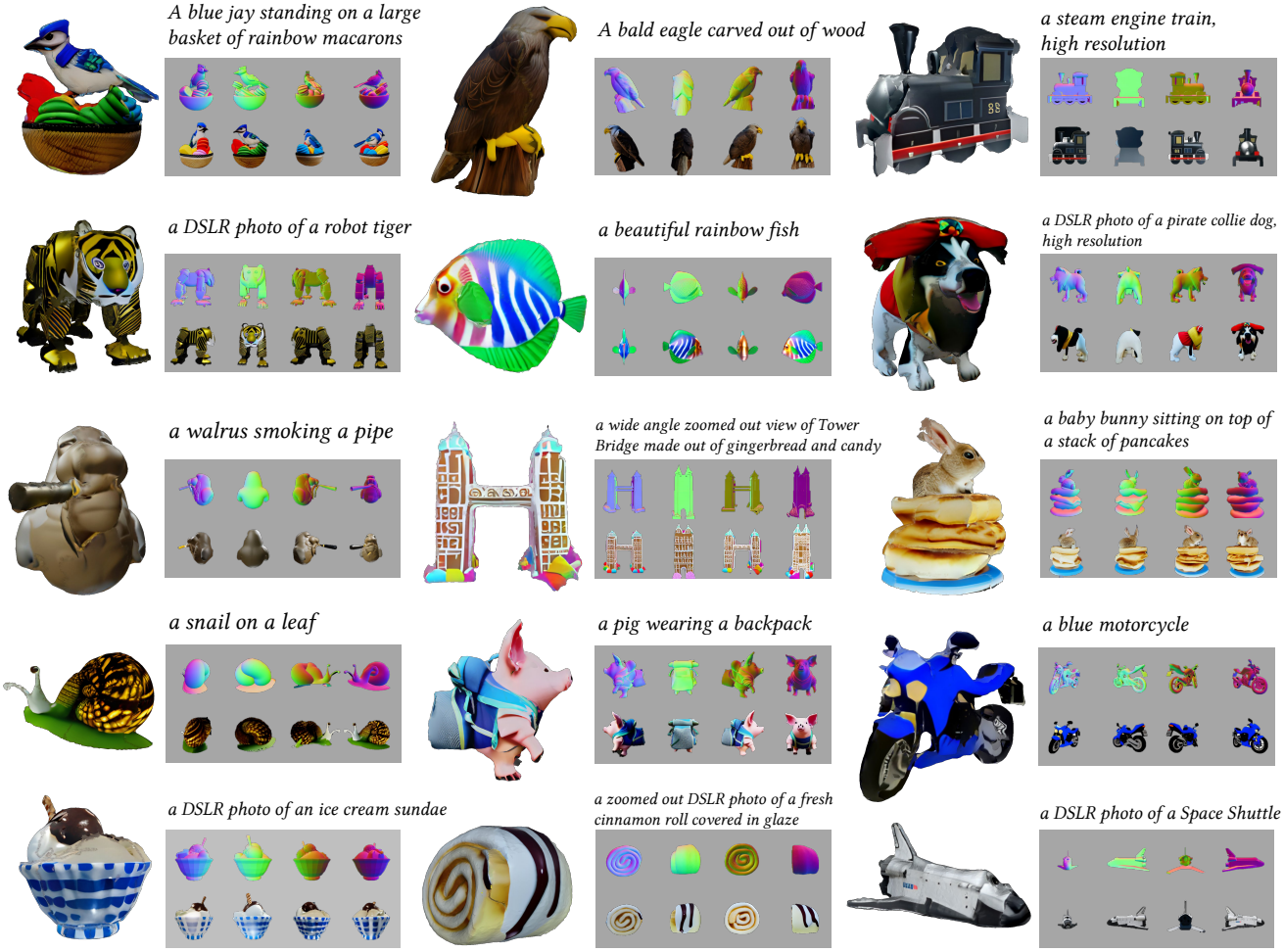


Figure 3. A gallery of our text-to-3d generation results. Given text prompts as description input, our method outputs high-quality textured triangle mesh in only 10 seconds. Note that the prompts are not from the training set. Best viewed zoomed in.

ing the L_1 loss between the rendered results and observations, as well as a normal consistency regularization. The loss function could be written as follows:

$$\mathcal{L}_V = \mathcal{L}_n + \lambda_\alpha \mathcal{L}_\alpha + \lambda_{nc} \mathcal{L}_{nc}, \tag{2}$$

where $\mathcal{L}_n = \frac{1}{4}\sum_i^4 ||n_i - \hat{n}_i||_1$ is the normal rendering loss. It measures the mean L_1 distance between rendered normal maps n and the observations $\hat{n}$ under different camera viewpoints $i \in \{0, 1, 2, 3\}$. Similarly, $L_\alpha = \frac{1}{4}\sum_i^4 ||\alpha_i - \hat{\alpha}_i||_1$ is the alpha mask loss, which computes the difference between rasterized object mask α and the observed $\hat{\alpha}$, and the latter could be obtained by a simple value thresholding $\delta = 0.05$ in the generated normal maps.

We additionally integrate a normal consistency term, denoted as $\mathcal{L}_{nc}$ to regularize the mesh. Specifically, this regularization is designed to smooth the mesh on a global scale by minimizing the negative cosine similarity between connected face normals. The hyperparameters $\lambda_\alpha, \lambda_{nc}$ which control the different weights for alpha mask loss and normal consistency regularization are set to 1 and 0.1 respectively. We adopt the nvdiffrast library [14] for differentiable rasterization.

After each optimization step, we further perform remeshing by merging or splitting triangle faces using the strategy from [27]. During experiments, we empirically found that only about 200 optimization steps are enough to generate a high-quality geometry mesh, which takes only around 2 to 3 seconds. As shown in the fig. 2 (c-e), the dog shape has been well optimized at around 200 steps.

3.4. Texture Synthesis

Texturing the mesh is another crucial step in achieving a high-quality result. Similar to the geometry generation, we initially synthesized multi-view texture maps, which were then applied to the generated geometry. In practice, another multi-view diffusion model generates the corresponding multi-view texture maps, conditioned on text prompts and the multi-view normal images.

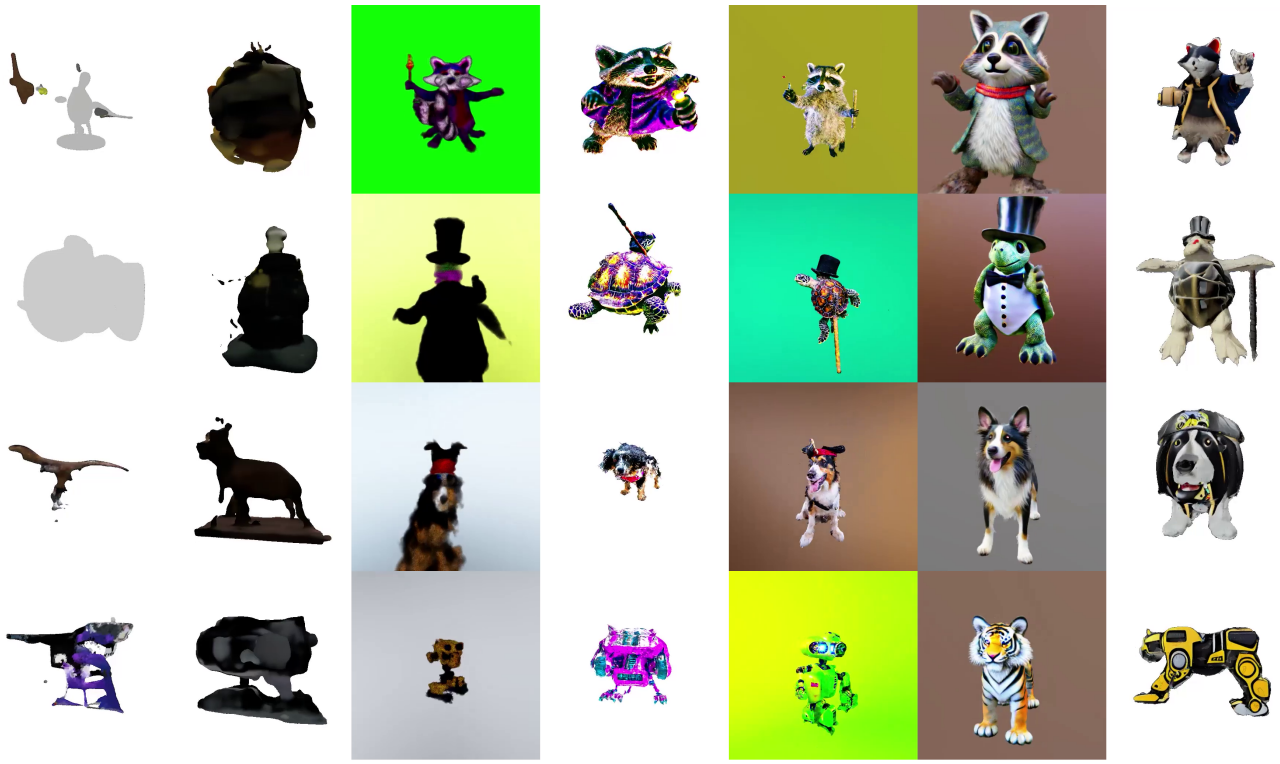


Figure 4. Qualitative comparisons. Direct 3D diffusion systems are not well generalized to the complex prompts. SDS-based methods except MVDream are slow and suffered from multi-face and over-saturation problems. MVDream can generate realistic geometry and appearance with fine details but has limited diversity (Fig. 5). In contrast, our system can generate realistic 3D models efficiently. Input prompts: 1) a zoomed out DSLR photo of a wizard raccoon casting a spell, 2) a DSLR photo of a turtle standing on its hind legs, wearing a top hat and holding a cane, 3) a DSLR photo of a pirate collie dog, high resolution, and 4) a DSLR photo of a robot tiger.

As shown in figure 1, the architecture of the multi-view normal-conditioned diffusion model is similar to the text-to-normal model, except that we extend the first convolution layer by increasing the number of channels to satisfy the normal latent condition input. Specifically, we initialize the extra trainable parameters in the first layer to zero before training. The normal condition plays a pivotal role in shape information and guides the model to generate both text- and shape-aligned texture images. We further apply super-resolution, i.e., Real-ESRGAN [41] on the generated texture maps to increase more appearance details, resulting in a 4 $\times$ resolution upscale from 256×256 to 1024×1024.

After obtaining the high-resolution RGB images, the final stage is to project these images to the shape geometry and generate a global texture. We perform UV parameterization and the Poisson blending algorithm [38] to alleviate multi-view inconsistency.

Iterative updating. In most cases, a single run of the pipeline is enough to generate high-quality results. However, since we generate 4-view information at once, there may be some areas unobserved in the generated RGB images (such as the top area of the object), and a texture refinement is required. To address this issue, we could iteratively update the generated images by using popular inpainting [21] pipelines in diffusion models to refine the generated textures. By computing a visibility mask at a new camera viewpoint, the invisible areas could be generated given a certain noise strength. During experiments, we found that only 1 or 2 iterations are enough to inpaint the unseen areas.

4. Implementation Details

In the following, we describe the aspects relevant to our system implementation details: dataset preparation in Sec. 4.1 and training setups in Sec. 4.2.

4.1. Dataset Preparation

We use the Objaverse [7] dataset for 2.5D training data generation, which is a large-scale 3D object dataset containing 800K high-quality models. We use the captions provided by cap3d [23] as text prompts. We filter the dataset by sorting the CLIP scores and selecting the top 500K objects with high text-image consistency. Each object is firstly normal-

ized at the center, and we render the scene from 32 viewpoints uniformly distributed in azimuth angles.

Besides, we also adopt a large-scale 2D image-text dataset to improve the generation diversity. Specifically, we use the COYO-700M dataset [2], which also contains metadata like resolution and CLIP scores [29]. We filter the dataset with both width and height greater than 512, aesthetic scores [31] greater than 5, and watermark scores lower than 0.5, which results in a 65M-size subset. Though the filtered dataset is reduced to 1/10 of the original size, it is still larger than the 3D dataset. Actually, we do not use the whole filtered dataset during training.

Please check the supplementary for more details.

4.2. Training Setup

As introduced above, we train the model with both 2.5D rendered images and natural images, with a probability of 80% to select the former. This makes the instances seen in each batch nearly equal for two kinds of data. We use the Stable Diffusion v2.1 base model as our backbone model and fine-tune the latent UNet only for another 50K steps with 1000 warmup steps. Similar to Zero123 [16], we use an image sample size of 256×256 for better and faster training convergence. The learning rate is set to $1e-5$. We drop the text prompt conditioning with a probability of 15% and apply a noise offset of 0.05. The full training procedure is conducted on 32 NVIDIA A100 80G GPUs (800K steps for the text-to-normal model and 18K steps for the normal-conditioned RGB model, which takes around 80 and 20 hours separately). The batch size is set to 45 on each GPU which leads to a total batch size of 1440.

5. Experiments

In the following, we represent the experiment results of our approach and evaluate the design of our system, including qualitative comparisons against state-of-the-art techniques and quantitative evaluations of model performances.

5.1. Text-to-3D contents generation

Given a random input text prompt, the proposed system is able to generate a high-fidelity 3D triangle mesh. Fig. 3 shows a gallery of our generation results. Generated multi-view normal and RGB images are also presented beside the 3D mesh. Our multi-view normal diffusion model is able to generate high-quality normal maps with expressive geometry details, and the normal-conditioned RGB diffusion model also generates detailed textures aligned with input normal maps, which validates the effectiveness of our cross-view attention design. All prompts used are unseen during training, which proves the generalization ability.

Figure 5. Comparison of sample diversity. Multiple samples are generated from the same prompt with different seeds. Our method is able to generate various samples while MVDream generates extremely similar results due to the SDS's mode-seeking nature.

5.2. Qualitative and Quantitative Evaluation

Qualitative evaluation. In this section we compare our method with SDS-based methods including DreamFusion [28], Fantasia3D [6], and MVDream [33]. We also compare with the direct 3D generation methods including Point-E [26] and Shap-E [12]. The text prompts are provided from DreamFusion, which were unseen during the fine-tuning for MVDream and ours. Fig. 4 illustrates qualitative comparisons of the renderings. It is clearly found that Point-E and Shap-E fail to generate reasonable text-aligned results. These direct 3D-based generation methods were trained on the relatively small 3D dataset compared to large-scale 2D text-image datasets, leading to poor generalization ability. Besides, DreamFusion and Fantasia3D suffer from the multi-face problem, while the results from the latter contain more details because of the supervision on geometry only. The rest two methods are 3D-aware so are able to produce reasonable 3D topology. MVDream generally achieves better visual quality, while our results are more consistent with the text prompts and take much less time to generate (35 mins v.s. 10s).

Sample diversity. Here, we compare the diversity of generated samples with MVDream. In this experiment, We generate 10 samples with the same prompt but different seeds. Fig. 5 presents the experiment results. Although both multi-view diffusion models are regularized by large-

Settings \ Metrics	FID(↓)	IS (↑)	CLIP (↑)
Groundtruth normal renderings	–	9.17	0.279
(T2N) w/o 2D joint training	43.61	8.94	0.270
(T2N) fewer 3D training data	37.29	9.44	0.294
(T2N) proposed	36.08	9.39	0.289
Groundtruth RGB renderings	–	11.31	0.261
(N2I) proposed	35.40	11.25	0.257

Table 1. We evaluate the proposed two multi-view diffusion models by computing. FID [11] (lower is better), IS [30] (higher is better), and CLIP scores [29] (higher is better) are used to measure the performance of different model variants.

scale image-caption datasets to prevent overfitting on the 3D dataset, the results from MVDream still collapse to a single type because of the mode-seeking nature of SDS. On the contrary, our method can still keep the content diversity of the pre-trained diffusion model because the construction of 3D models is independent of the diffusion process, which would faithfully follow the random denoising process.

Quantitative evaluation. In the following, we quantitatively evaluate image generation quality and the text-image consistency of the proposed two novel multi-view diffusion models. Table 2 demonstrates the evaluation results. Specifically, Frechet Inception Distance (FID) [11] and Inception Score (IS) [30] are adopted to measure the generation image quality and CLIP score cosine similarity [29] is calculated to measure the text-image consistency. We randomly select 2000 subjects as well as their multi-view RGB and normal renderings in the Objaverse [7] dataset as our evaluation database. FID and IS are calculated independently of viewpoints while the CLIP similarity is selected as the max value across all 4-view scores.

In general, we find that the proposed model achieves similar or even better results compared to the groundtruth renderings, which proves the high image quality and image-text consistency. We also evaluate the training strategies used in multi-view normal diffusion training, including using 2D large-scale dataset joint training, using higher consistency but fewer 3D subjects for training. It is clearly shown that the performance drastically drops when training without a 2D wild dataset injection. We believe that this is because fine-tuning purely multi-view normal data, would lead to a catastrophic forgetting of the original learned distribution and leads to poor learning ability. Training using fewer but higher text-consistent data leads to better IS and CLIP similarities, but worse FID. In practice, we found this model has lower generalization ability and diversity compared to the model that used more 3D data.

We also compare to previous SOTA methods quantitatively in Table 2. We randomly selected 50 prompts from Dreamfusion, not seen during our method and MVDream's fine-tuning, as the evaluation set. We adopt IS, CLIP scores and FID (Objaverse rendering and COCO validation set) to evaluate rendering results. Running time is also preseneted. Our method outperforms direct 3D diffusion methods significantly across all metrics and is on par with state-of-the-art SDS-based methods. Our method achieves slightly better CLIP scores and FID but worse IS compared to MVDream, and consumes significantly less time for generation.

Please check the supplementary for more evaluations.

Methods \ Metrics	IS (↑)	CLIP (↑)	FID (↓ objv.)	FID (↓ COCO)	Run Time
Point-E	7.265	0.220	104.105	164.765	∼ 20 s
Shap-E	7.412	0.236	103.557	163.105	∼ 4 s
Dreamfusion	7.724	0.245	125.873	150.285	∼ 50 m
Fantasia3d	8.311	0.207	132.941	150.255	∼ 115 m
ProlificDreamer	9.457	0.269	121.577	124.185	∼ 5 h
MVDream	8.180	0.262	117.715	133.089	∼ 35 m
Ours	8.111	0.267	82.324	126.014	∼ 10 s

Table 2. Quantitative comparisons with previous methods.

6. Limitations and Future Work

Limited view numbers. Due to the small view number, areas such as top, bottom and concavity cannot be fully observed, and thus their geometry or appearance cannot be well reconstructed. Apart from the iterative update scheme, the multi-view diffusion can be extended to more views.

Texture quality. For the appearance, we finetune a multi-view normal-conditioned diffusion model for efficiency. However, the ability of generating realistic images is degraded due to the texture quality of the 3D training samples and their rendering quality. Apart from further enhancing the training samples, we can also apply the state-of-the-art texture generation systems [4] for non-time-sensitive tasks.

Please check the supplementary for more discussions.

7. Conclusion

We propose to perform fast text-to-3D generation by fine-tuning a multi-view 2.5D diffusion from pre-trained RGB diffusion models. To learn multi-view consistency, the model is fine-tuned on multi-view normal map renderings, with cross-view attention as the structural guarantee. After the simultaneous generation of multi-view normal maps, 3D models are obtained by deforming meshes by differentiable rasterization. Finally, appearance is generated by multi-view normal-conditioned RGB diffusion. Our generation pipeline produces diverse and high-quality 3D models in 10 seconds, and demonstrates strong generalization to complex content and generates fine details. Extensive experiments are conducted to show that our method enables fast generation of realistic, complex, and diverse models.

8. Acknowledgments

This work is supported by National Natural Science Foundation of China under Grants 62001213 and Hong Kong RGC GRF 16206722.

References

[1] 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. 4

[2] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 7

[3] 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. 1, 2

[4] 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. 8

[5] Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. *arXiv preprint arXiv:2304.06714*, 2023. 2

[6] 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. 1, 7

[7] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 1, 2, 6, 8

[8] Sara Fridovich-Keil, Giacomo Meanti, Frederik Rahbæk Warburg, Benjamin Recht, and Angjoo Kanazawa. K-planes: Explicit radiance fields in space, time, and appearance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12479–12488, 2023. 4

[9] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2020. 4

[10] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv preprint arXiv:2303.05371*, 2023. 2

[11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 8

[12] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 1, 2, 7

[13] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *International journal of computer vision*, 38:199–218, 2000. 4

[14] 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. 5

[15] 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*, pages 300–309, 2023. 1

[16] 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. *arXiv preprint arXiv:2303.11328*, 2023. 2, 7

[17] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2

[18] Zhen Liu, Yao Feng, Michael J Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3d mesh modeling. *arXiv preprint arXiv:2303.08133*, 2023. 2

[19] 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. 2

[20] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4):163–169, 1987. 4

[21] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 6

[22] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2837–2845, 2021. 1, 2

[23] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *arXiv preprint arXiv:2306.07279*, 2023. 6

[24] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 4

[25] 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. 4

[26] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 1, 2, 7

[27] Werner Palfinger. Continuous remeshing for inverse rendering. *Computer Animation and Virtual Worlds*, 33(5):e2101, 2022. 5

[28] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 1, 2, 7

[29] 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. 7, 8

[30] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016. 8

[31] Christoph Schuhmann. Improved aesthetic predictor. https://github.com/christophschuhmann/improved-aesthetic-predictor, 2022. 7

[32] 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. 1

[33] 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, 7

[34] Jaehyeok Shim, Changwoo Kang, and Kyungdon Joo. Diffusion-based signed distance fields for 3d shape generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20887–20897, 2023. 1, 2

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

[36] Shitao Tang, Fuyang Zhang, Jiacheng Chen, Peng Wang, and Yasutaka Furukawa. Mvdiffusion: Enabling holistic multi-view image generation with correspondence-aware diffusion. *arXiv preprint arXiv:2307.01097*, 2023. 2

[37] Hung-Yu Tseng, Qinbo Li, Changil Kim, Suhib Alsisan, Jia-Bin Huang, and Johannes Kopf. Consistent view synthesis with pose-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16773–16783, 2023. 2

[38] Michael Waechter, Nils Moehrle, and Michael Goesele. Let there be color! — Large-scale texturing of 3D reconstructions. In *Proceedings of the European Conference on Computer Vision*. Springer, 2014. 6

[39] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12619–12629, 2023. 1, 2

[40] 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. 2

[41] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *International Conference on Computer Vision Workshops (ICCVW)*, 2021. 6

[42] 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. 1, 2

[43] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 4

[44] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5525–5534, 2019. 4

[45] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. *arXiv preprint arXiv:2210.06978*, 2022. 2

[46] Jingyang Zhang, Shiwei Li, Zixin Luo, Tian Fang, and Yao Yao. Vis-mvsnet: Visibility-aware multi-view stereo network. *International Journal of Computer Vision*, 131(1): 199–214, 2023. 4

[47] Jingyang Zhang, Shiwei Li, Yuanxun Lu, Tian Fang, David McKinnon, Yanghai Tsin, Long Quan, and Yao Yao. Jointnet: Extending text-to-image diffusion for dense distribution modeling. *International Conference on Learning Representations (ICLR)*, 2024. 2

[48] Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5826–5835, 2021. 2