

# Wonder3D++: Cross-domain Diffusion for High-fidelity 3D Generation from a Single Image

Anonymous submission

**Abstract**—In this work, we introduce Wonder3D++, a novel method for efficiently generating high-fidelity textured meshes from single-view images. Recent methods based on Score Distillation Sampling (SDS) have shown the potential to recover 3D geometry from 2D diffusion priors, but they typically suffer from time-consuming per-shape optimization and inconsistent geometry. In contrast, certain works directly produce 3D information via fast network inferences, but their results are often of low quality and lack geometric details. To holistically improve the quality, consistency, and efficiency of single-view reconstruction tasks, we propose a cross-domain diffusion model that generates multi-view normal maps and the corresponding color images. To ensure the consistency of generation, we employ a multi-view cross-domain attention mechanism that facilitates information exchange across views and modalities. Lastly, we introduce a cascaded 3D mesh extraction algorithm that drives high-quality surfaces from the multi-view 2D representations in only about 3 minute in a coarse-to-fine manner. Our extensive evaluations demonstrate that our method achieves high-quality reconstruction results, robust generalization, and good efficiency compared to prior works.

**Index Terms**—3D Generation, 3D Reconstruction, Diffusion model, Differentiable rendering

## I. INTRODUCTION

Reconstructing 3D geometry from a single image [13], [25], [36], [42], [43], [45], [48] stands as a fundamental task in computer graphics and 3D computer vision, benefiting a wide range of versatile applications such as novel view synthesis [7], [24], [39], 3D content creation [38], [52], and robotics grasping [29], [92]. However, this task is notably challenging since it is ill-posed and demands the ability to discern the 3D geometry of both visible and invisible parts. This ability requires extensive knowledge of the 3D world.

Recently, the field of 3D generation has experienced rapid and flourishing development with the introduction of diffusion models. A growing body of research [5], [34], [51], [70], [73], such as DreamField [23], DreamFusion [51], and Magic3D [34], resort to distilling prior knowledge of 2D image diffusion models or vision language models to create 3D models from text or images via Score Distillation Sampling (SDS) [51]. Despite achieving some degree of generalization performance, these methods still encounter challenges like multi-view inconsistencies and low-quality results.

To enhance the multi-view consistency, several methods emerged that directly generate multi-view 2D images, with representative works including SyncDreamer [38] and MV-Dream [59]. By improving information exchange between

views using attention mechanisms, existing methods generate multi-view images and then optimize 3D representations—such as Neus [71] and NeRF [46]—to reconstruct textured meshes. However, current multi-view generative methods face significant challenges. First, relying solely on color images often compromises the fidelity of the generated shapes, making it difficult to recover geometric details without incurring high computational costs. Additionally, most multi-view diffusion models are constrained to predefined camera types, which can introduce distortions when processing images captured in uncontrolled environments, further amplifying multi-view inconsistencies.

Recently, various techniques have been introduced that aim to generate 3D geometries via network inference, including point clouds [44], [48], [87], [93], meshes [16], [40], neural fields [1], [4], [8], [15], [17], [19], [21], [25], [26], [28], [47], [49], [72], [88]. Most of them attempt to train 3D generative models from scratch on datasets consisting of various 3D assets. To further enhance generalizability, recent approaches have focused on generating sparse view images using a multi-view diffusion model as a supplementary cue, followed by a feed-forward sparse-view reconstruction model to create 3D objects [62], [74], [81], [82]. However, despite some of these methods requiring over a hundred GPUs for training, they still face significant challenges in achieving high-quality geometry and textures. Additionally, since these methods rely on neural networks for direct forward propagation to generate 3D models, the resulting objects often lack precise alignment with the input images.

To address the challenges of fidelity, consistency, and generalizability in previous works, we propose a novel approach for single-view 3D reconstruction by generating multi-view consistent normal maps and their corresponding color images using a cross-domain diffusion model. The core idea is to extend diffusion frameworks to jointly model the distribution of two domains: normals and colors. Normal maps capture surface undulations and variations, encoding detailed geometric information that allows for high-fidelity 3D geometry extraction from 2D representations. By leveraging 2D representations, Wonder3D++ is built on the pre-trained Stable Diffusion model [53], utilizing strong priors that enable robust zero-shot generalization.

The following technical designs of Wonder3D++ make it a robust and efficient tool for creating 3D shapes from single images: 1) **Cross-domain switcher**: The domain switcher enables the diffusion model to generate either normal maps or

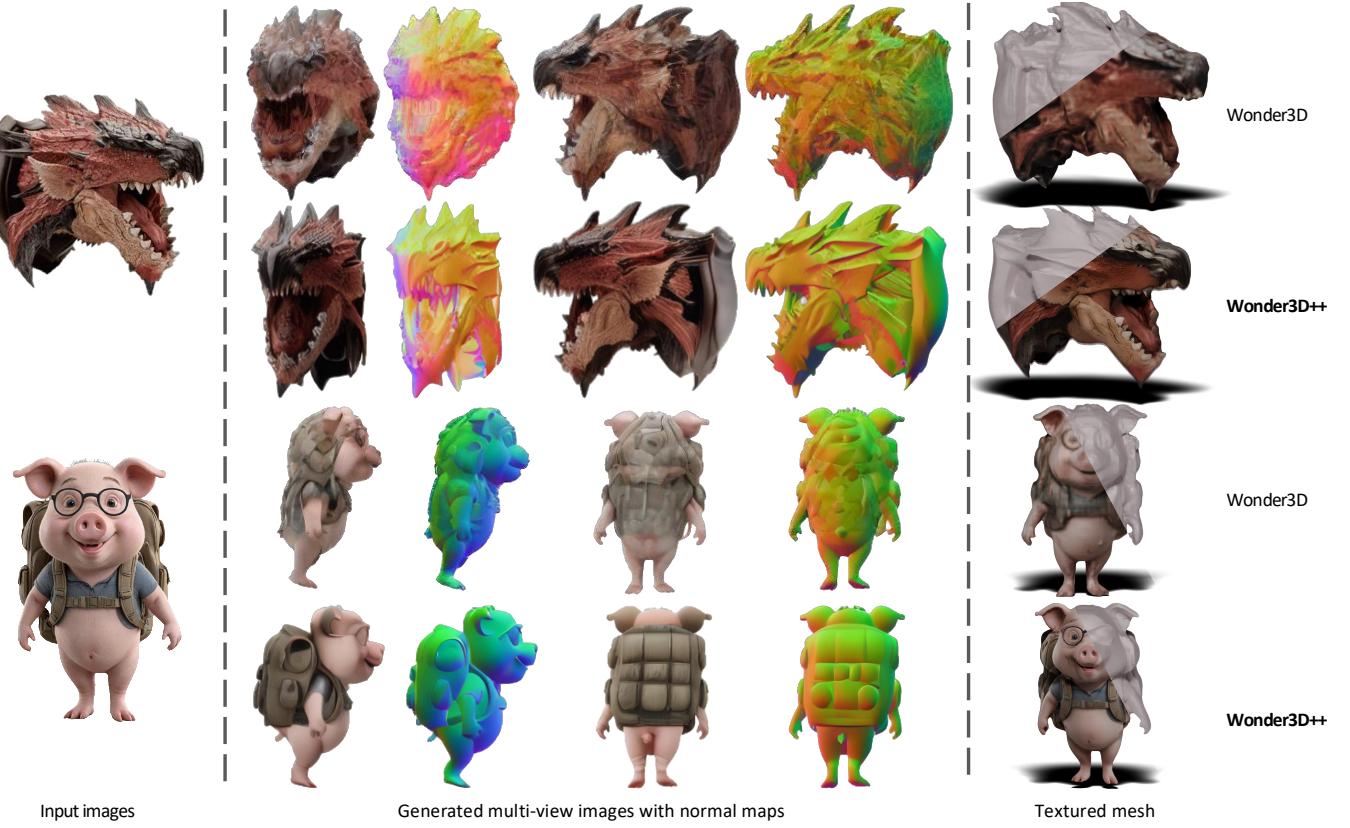


Fig. 1. *Wonder3D++* reconstructs highly-detailed textured meshes from a single-view image in only 3 minutes. *Wonder3D++* first generates consistent multi-view normal maps with corresponding color images via a cross-domain diffusion model and then leverages a cascaded 3D mesh extraction method to achieve fast and high-quality reconstruction.

color images with minimal modifications to the original model. 2) **Cross-domain attention**: We incorporate cross-domain attention mechanisms to facilitate information exchange between the two domains, improving consistency and quality. This enhances the model’s ability to recover high-fidelity geometry. 3) **Camera type switcher**: To handle input images from various sources—such as imaginative images generated by text-to-image models or everyday objects captured by smartphones—we introduce a camera type switcher. This allows the model to generate multi-view images using either perspective or orthogonal projection, enhancing its robustness. 4) **Cascaded 3D mesh extraction**: To address resolution limitations and improve efficiency, we propose a Cascaded 3D mesh extraction algorithm. This consists of a) geometric initialization, b) inconsistency-aware coarse reconstruction, and c) iterative mesh refinement via our cross-domain multi-view enhancement module. This process generates high-quality textured meshes while mitigating issues of multi-view inconsistency and limited resolution (see Figure 1).

We conduct extensive experiments on the Google Scanned Object dataset [14] and various 2D images with different styles. The experiments validate that *Wonder3D++* achieves a leading level of geometric details with high efficiency and strong generalization among current zero-shot single-view reconstruction methods.

**Differences with conference version.** *Wonder3D++* introduces several key improvements over *Wonder3D* [41], sig-

nificantly enhancing robustness and quality: 1) the addition of a **camera type switcher**, allowing the model to handle input images captured with various camera settings; 2) the implementation of **multi-domain pre-training**, ensuring a seamless transition from stable diffusion pre-training to the desired joint distribution for 3D data; and 3) replacing implicit neural SDF fields with explicit mesh representations for 3D textured mesh reconstruction, introducing a **cascaded 3D mesh extraction** algorithm that greatly improves both efficiency and effectiveness.

## II. RELATED WORKS

### A. 2D Diffusion Models for 3D Generation

Recently, benefiting from the availability of large-scale high-quality data, there have been significant advancements in foundational models, such as 2D diffusion models [9], [20], [55] and large vision-language models (e.g., the CLIP model [53]). These models offer new opportunities for generating 3D assets by leveraging the strong priors provided by these pre-trained architectures. In comparison, directly trained native 3D generators often struggle to ensure both generalizability and quality, due to the relative scarcity of large-scale data in the three-dimensional domain. Therefore, pioneering works such as DreamFusion [51] and SJC [70] have proposed distilling a 2D text-to-image generation model to generate 3D shapes from textual descriptions, and subsequent works have adopted this per-shape optimization scheme, building upon the

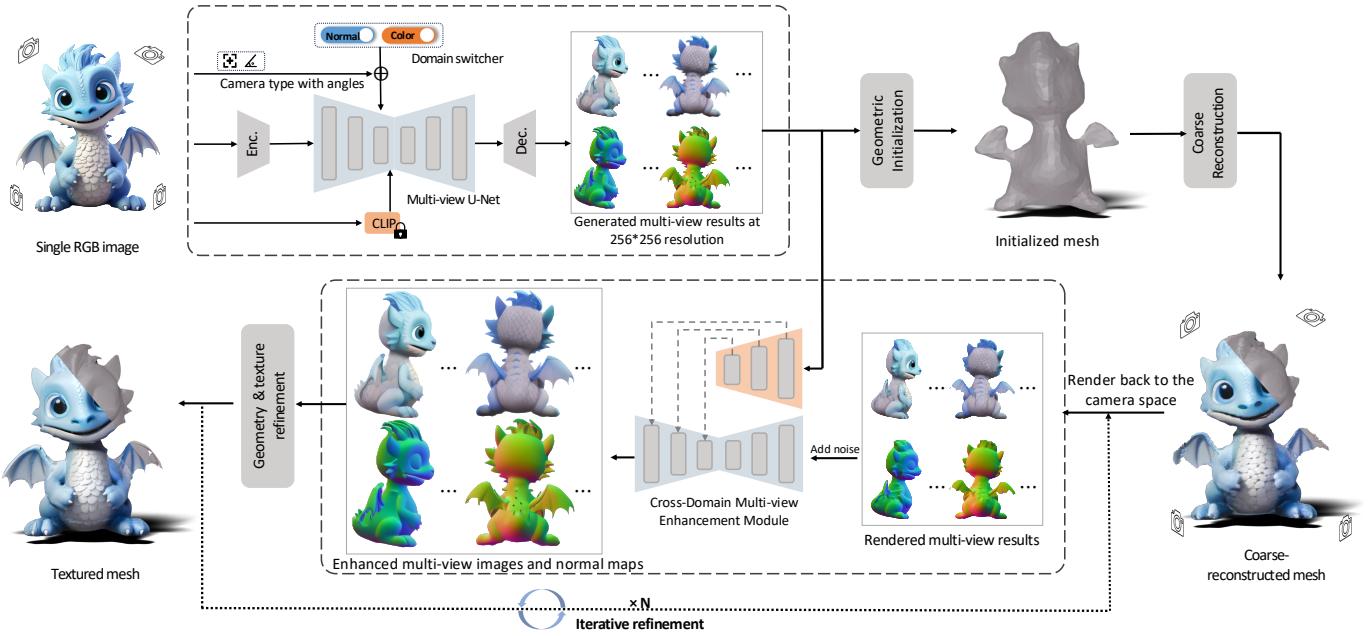


Fig. 2. Overview of *Wonder3D++*. Given a single image, *Wonder3D++* takes the input image, the image embedding produced by CLIP model [53], the camera parameters of multiple views, a camera type switcher and a domain switcher as conditioning to generate consistent multi-view normal maps and color images. Subsequently, *Wonder3D++* applies an innovative cascaded 3D mesh reconstruction algorithm, which utilizes a coarse-to-fine strategy consisting of geometric initialization, coarse mesh reconstruction, and iterative mesh refinement to produce a 3D mesh with high-quality geometry and high-fidelity textures.

success of these initial approaches. For the task of text-to-3D [2], [5], [6], [22], [34], [56], [57], [66], [73], [77], [85], [95] or image-to-3D synthesis [45], [52], [54], [58], [63], [80], these methods typically optimize a 3D representation (i.e., NeRF, mesh, or SDF), and then leverage differentiable rendering to deliver 2D images from various viewpoints. The images are then fed into the 2D diffusion models or CLIP model for calculating SDS [51] losses, which guide the 3D shape optimization. While this endows 3D generation methods with unprecedented generalization capabilities, most of these methods suffer from low efficiency and the “multi-face” problem (i.e., the Janus problem [51]) due to the lack of explicit 3D supervision. Moreover, per-shape optimization, which typically takes tens of minutes, significantly limits the practicality of this methodology.

### B. Multi-view Diffusion Models

To alleviate the “multi-face” problem, some efforts [3], [11], [18], [31], [37], [61], [64], [65], [67], [76], [79], [84], [86], [89], [94] are made to extend 2D diffusion models from single-view images to multi-view consistent images. However, most of these methods focus on image generation and are not designed for 3D reconstruction. The recent works Viewset Diffusion [61], SyncDreamer [38], and MVDream [59] share a similar idea to produce consistent multi-view color images via attention layers. However, unlike that normal maps explicitly encode geometric information, reconstruction from color images always suffers from texture ambiguity, and, thus, they either struggle to recover geometric details or require huge computational costs. For instance, SyncDreamer [38] requires dense views for 3D reconstruction but still suffers from low-quality geometry and blurring textures. MVDream [59] still

resorts to a time-consuming optimization using SDS loss for 3D reconstruction, and its multi-view distillation scheme requires 1.5 hours. Moreover, the inconsistency between the camera type of the input images and the predefined camera type leads to unnatural distortions in the generated meshes. In contrast, thanks to our proposed domain switcher, our method can simultaneously generate multi-view normal maps for geometric reconstruction and RGB images for coloring. Additionally, our method provides the flexibility to generate results in either orthogonal or perspective camera settings, adapting to different requirements as needed.

### C. 3D Generative Models

Instead of performing a time-consuming per-shape optimization guided by 2D diffusion models, some works attempt to directly train 3D diffusion models based on various 3D representations, like point clouds [44], [48], [87], [93], meshes [16], [40], neural fields [1], [4], [8], [15], [17], [19], [25], [26], [28], [47], [49], [72], [88]. However, due to the limited size of publicly available 3D assets datasets, most of the works have only been validated on limited categories of shapes or lack of sufficient generalization properties, and how to scale up on large datasets is still an open problem. In order to effectively improve the generalization performance of the model, a portion of recent work [33], [62], [74], [81], [82] seeks to first utilize the multi-view diffusion model to infer multi-view images and then use sparse view reconstruction models to obtain meshes in a single forward pass. Such sparse view reconstruction models are usually based on transformer architectures to predict Nerf [81], Gaussian Splatting [62], [82] or triplane [74] representations and employ differentiable rendering to compute the loss function, which hinders the

efficiency and stability of the training process. As a result, the training process typically requires tens or hundreds of GPUs and it is difficult to obtain geometries and textures with sufficient details. On the contrary, our method first adopts 2D representations, enabling it to leverage 2D diffusion models [55], whose pre-trained priors significantly enhance zero-shot generalization ability. Building on this, we then employ a cascaded approach to efficiently reconstruct 3D meshes, improving both geometric and texture quality while avoiding the high computational costs associated with training a feedforward reconstruction model.

### III. PROBLEM FORMULATION

#### A. Diffusion Models

Diffusion models [20], [60] are first proposed to gradually recover images from a specifically designed degradation process, where a forward Markov chain and a Reverse Markov chain are adopted. Given a sample  $z_0$  drawn from the data distribution  $p(z)$ , the forward process of denoising diffusion models yields a sequence of noised data  $\{z_t \mid t \in (0, T)\}$  with  $z_t = M_t z_0 + \sigma_t \epsilon$ , where  $\epsilon$  is random noise drawn from distribution  $\mathcal{N}(0, 1)$ , and  $M_t, \sigma_t$  are fixed sequence of the noise schedule. The forward process will be iteratively applied to the target image  $z$  until the image becomes complete Gaussian noise at the end. On the contrary, the reverse chain iteratively denoises the corrupted image, recovering  $z_{t-1}$  from  $z_t$  by step-by-step predicting the added random noise  $\epsilon$ .

#### B. The Distribution of 3D Assets

Unlike prior works that adopt 3D representations like point clouds, tri-planes, or neural radiance fields, we propose that the distribution of 3D assets denoted as  $p_a(\mathbf{z})$ , can be modeled as a joint distribution of its corresponding 2D multi-view normal maps and color images. Specifically, given a set of cameras  $\{\pi_1, \pi_2, \dots, \pi_K\}$  and a conditional input image  $y$ , we have

$$p_a(\mathbf{z}) = p_{nc}(N^{1:K}, X^{1:K} | y), \quad (1)$$

where  $p_{nc}$  is the distribution of the normal maps  $N^{1:K}$  and color images  $X^{1:K}$  observed from a 3D asset conditioned on an image  $y$ . For simplicity, we omit the symbol  $y$  for this equation in the following discussions. Therefore, our goal is to learn a model  $f$  that synthesizes multiple normal maps and color images of a set of camera poses denoted as

$$(N^{1:K}, X^{1:K}) = f(y, \pi_{1:K}). \quad (2)$$

Finally, we can formulate this cross-domain joint distribution as a Markov chain within the diffusion scheme:

$$p(N_T^{(1:K)}, X_T^{(1:K)}) \prod_t p_{nc}(N_t^{(1:K)}, X_t^{(1:K)} | N_t^{(1:K)}, X_t^{(1:K)}), \quad (3)$$

where  $p(N_T^{(1:K)}, X_T^{(1:K)})$  are random Gaussian noises. Our key problem is to characterize the joint distribution  $p_{nc}$ , so that we can sample from this Markov chain to generate normal maps and images.

#### C. Mesh Reconstruction via Differentiable Rendering

Given the synthesized multi-view normal maps  $N^{1:K}$ , we extract the 3D mesh using differentiable rendering, ensuring accurate geometry reconstruction through gradient-based optimization. Specifically, starting with an initial triangle mesh  $M$ , composed of vertices  $V$  and faces  $F$ , the multi-view normal maps  $N^{1:K}$  and masks  $M^{1:K}$  are rendered differentiably using the nvdiffraast library [30]. The reconstruction loss is then computed, and the gradient is back-propagated to optimize the vertices  $V$ , refining the mesh geometry based on the multi-view input.

To directly optimize the vertices of the mesh and continuously adjust the topology of the mesh and the edge lengths of the mesh, we adopt the remeshing strategy proposed by [50], and apply it after each optimization iteration. This process involves edge-splitting, edge-collapsing, edge-flipping, and smoothing to adaptively adjust local edge lengths and correct defects caused by sparse or inconsistent vertex gradients.

The reasons we choose an explicit mesh representation over implicit methods like Neus [71] are twofold: 1) The **computational complexity** of optimizing with the explicit method scales with the number of mesh vertices and the square of the resolution of the multi-view normal maps, whereas in implicit neural field methods, it scales with the cube of the image resolution, making our approach more efficient. 2) Directly optimizing mesh vertices and faces allows our method to **integrate seamlessly** with the cascaded mesh extraction process, eliminating the computational overhead and detail loss that typically occur during conversions between mesh and implicit SDF representations.

### IV. METHOD

As per our problem formulation in Section III-B, we propose a multi-view cross-domain diffusion scheme, which operates on two distinct domains to generate multi-view consistent normal maps and color images. The overview of our method is presented in Figure 2. First, our method adopts a **cross-domain multi-view diffusion** scheme to generate multi-view normal maps and color images and enforces consistency across different views using multi-view attention. Second, our proposed domain switcher allows the diffusion model to operate on more than one domain. A **cross-domain attention** is proposed to propagate information between the normal domain and color image domain ensuring geometric and visual coherence between the two domains (see Section IV-A). Furthermore, to achieve high-quality geometry and appearance from multi-view 2D normal and color images while minimizing blurring from sparse views and multi-view inconsistencies, we introduce a **cascaded 3D mesh extraction algorithm**. This algorithm reconstructs the 3D mesh in a coarse-to-fine manner, iteratively refining it with our cross-domain multi-view enhancement module (see Section IV-B).

#### A. Multi-view Cross-domain Diffusion Model

Our model is built upon the pre-trained stable diffusion models [53] to leverage its strong priors. Given that current 2D diffusion models [36], [53] are designed for a single

domain, the main challenge lies in how to effectively extend stable diffusion models to support multi-domain operations. To achieve this goal, we explore several possible designs.

1) *Naive Solutions for Cross-domain Generation*: A straightforward solution is to add four more channels to the output of the UNet module representing the extra domain. Therefore, the diffusion model can simultaneously output normals and color image domains. However, we notice that such a design suffers from low convergence speed and poor generalization. This is because the channel expansion may perturb the pre-trained weights of stable diffusion models and therefore cause catastrophic model forgetting.

Revisiting Eq. 1, it is possible to factor the joint distribution into two conditional distributions:

$$q_a(\mathbf{z}) = q_n(N^{1:K}) \cdot q_c(X^{1:K} | N^{1:K}). \quad (4)$$

This equation suggests an alternative solution where we could initially train a diffusion model to generate normal maps and then train another diffusion model to produce color images, conditioning on the generated normal maps (or vice versa).

However, implementing this two-stage framework introduces certain complexities and significantly increases computational costs. More importantly, as discussed in Section III-B, multi-view color images and normal maps share a joint distribution in 3D data, where each domain influences the other. Jointly learning these domains facilitates cross-domain corroboration, enabling each domain to benefit from the other. In contrast, a sequential learning strategy trains the two stages separately, resulting in a larger domain gap and performance degradation. This occurs because the second-stage model is trained on ideal GT images but must infer from the imperfect images predicted by the first-stage model. For a detailed discussion, please refer to Section V-F.

2) *Domain Switcher*: To overcome these difficulties mentioned above and model the joint distribution of the multi-view color images and normal maps, we design a cross-domain diffusion scheme via a **domain switcher**, denoted as  $s_d$ . The switcher  $s_d$  is a one-dimensional vector serving as labels for distinct domains, and we further feed the switcher into the diffusion model as an extra input. Therefore, the formulation of Eq. 2 can be extended as:

$$N^{1:K}, X^{1:K} = f(y, \pi_{1:K}, s_d = n), f(y, \pi_{1:K}, s_d = c). \quad (5)$$

The domain switcher is first encoded via positional encoding [46] and subsequently concatenated with the time embedding. This combined representation is then injected into the UNet of the stable diffusion models. Interestingly, experiments show that this subtle modification does not significantly alter the pre-trained priors. As a result, it allows for fast convergence and robust generalization, without requiring substantial changes to the stable diffusion models.

3) *Cross-domain Attention*: Using the proposed domain switcher, the diffusion model can generate two different domains. However, it is important to note that for a single view, there is no guarantee that the generated color image and the normal map will be geometrically consistent if the color images and normal maps are sampled independently from the distribution. To address this issue and ensure consistency

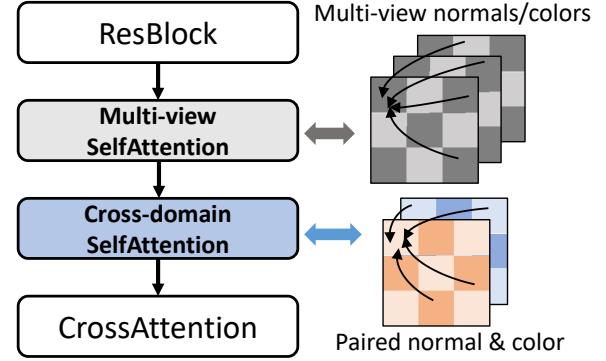


Fig. 3. The illustration of the structure of the multi-view cross-domain transformer block, where the multi-view attention layer and cross-domain attention layer facilitate information exchange across different views and domains, respectively.

between the generated normal maps and color images, we introduce a cross-domain attention mechanism to facilitate the exchange of information between the two domains. This mechanism aims to ensure that the generated outputs align well in terms of geometry and appearance.

Given a latent embedding  $Z$ , the original self-attention mechanism [68] can be defined as:

$$Attn(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \cdot V \quad (6)$$

where the query, key, and value are projected by learnable matrices as follows:

$$Q = W^Q Z, \quad K = W^K Z, \quad V = W^V Z \quad (7)$$

Our cross-domain attention layer maintains a similar structure as the original self-attention layer and is integrated seamlessly before the cross-attention layer in each transformer block of the UNet, as depicted in Figure 3. In the cross-domain attention layer, given the  $Z_n$  and  $Z_x$  represent the latent feature of the normal domain and the color domain respectively,  $[ \cdot ]$  signifies concatenation operation, the key and value are derived as follows:

$$K = W^K [Z_n, Z_x], \quad V = W^V [Z_n, Z_x] \quad (8)$$

We then apply attention operation specifically as:

$$Z_n = Attn(Z_n, K, V), \quad Z_x = Attn(Z_x, K, V) \quad (9)$$

This design ensures that the generations of color images and normal maps are closely correlated, thus promoting geometric consistency between the two domains.

4) *Multi-view Cross-domain Transformer Block*: The prior 2D diffusion models [36], [53] generate each image separately, so that the resulting images are not geometrically and visually consistent across different views. To enhance consistency among different views, similar to prior works such as SyncDreamer [38] and MVdream [59], we utilize attention mechanism to facilitate information propagation across different views, implicitly encoding multi-view dependencies (as illustrated in Figure 3). This is achieved by extending the original self-attention layers to be global-aware, allowing connections to other views within the attention layers. Keys

and values from different views are connected to each other to facilitate the exchange of information. By sharing information across different views and domains within the attention layers, the model perceives cross-domain multi-view correlation and is capable of generating consistent multi-view color images and normal maps.

5) *Camera Type Switcher*: The input images typically come from diverse sources, such as real photographs, artist-created figures, or images generated by 2D AIGC tools, resulting in significant variation in their projection types. For instance, camera-captured images usually follow perspective projection, while synthetic images are often better suited to orthogonal projection. Ignoring these underlying camera projections makes it challenging to generate multi-view consistent and visually plausible images, leading to distortions in the reconstructed mesh. To handle varying camera settings in input images, we introduce a one-dimensional camera type switcher,  $s_c$ , allowing users to specify the projection type, such as orthogonal or perspective. For perspective projection, we adopt the standard 35mm focal length. The camera switcher is integrated into the model similarly to the domain switcher, enabling the generation of multi-view images consistent with the specified camera settings.

6) *Multi-Stage Training Scheme*: To enable a seamless transition from a pre-trained stable diffusion model to the target joint distribution of 3D objects, we propose a multi-stage training strategy consisting of three stages.

a) *Multi-Domain Pre-Training Stage*: Building on the stable diffusion model, we initially remove the domain switcher and cross-domain attention layers, modifying only the self-attention layers into a multi-view design. Here, the model is fine-tuned for multi-view generation from a single view within the same domain, predicting multi-view colors, normals, or masks from a single view of color, normal, or mask, respectively. This stage equips the model with robust multi-view prior knowledge across various domains.

b) *Mixed-Domain Fine-Tuning Stage*: In this stage, we introduce the domain switcher, fine-tuning the model to generate either multi-view colors or normals from a single-view color image, guided by the domain switcher. This enables the model to produce multi-view outputs in either the color or normal domain from a single-view color image.

c) *Cross-Domain Alignment Fine-Tuning Stage*: In the final stage, we incorporate cross-domain attention layers while freezing all previously trained parameters. The model is then fine-tuned to jointly predict both multi-view color images and normal maps from a single-view color image. The cross-domain attention layers facilitate information exchange across color and normal domains, enhancing the consistency between generated normal maps and color images.

This multi-stage training scheme enhances training stability and leverages multi-task learning principles, enabling our multi-view cross-domain diffusion model to generate high-quality multi-view color images and normal maps with excellent generalizability.

## B. Cascaded 3D Mesh Extraction

To efficiently extract explicit 3D geometry from 2D normal maps and color images, we use the mesh-based differentiable rendering algorithm described in Section III-C, optimizing directly on the mesh vertices  $V$  guided by the generated 2D representations. Our cascaded 3D mesh extraction process is divided into three stages: 1) *Geometric Initialization*: Initialize a rough mesh with approximate topology for optimization. 2) *Inconsistency-Aware Coarse Reconstruction*: Reconstruct a coarse shape and texture using the generated multi-view color images and normals. 3) *Iterative Refinement*: Iteratively refine the geometry and texture through a cross-domain enhancement module.

1) *Geometric Initialization*: Explicit mesh optimization methods typically require a mesh with approximate topology for initialization. Following [78], we first use normal map integration from the front and back views to obtain two corresponding depth maps. These depth maps are then projected into 3D space to create two point cloud parts with normal, which are combined into an initialized mesh using the Poisson reconstruction [27] method. Unlike the commonly used sphere initialization, this approach not only speeds up optimization but also enables our method to capture relatively complex shapes by providing an approximate topological structure.

However, this method does not always yield initializations with the right topology. As shown in Figure 4, when the target object's geometry is concave rather than convex, this strategy leads to failed initialization. This issue arises because the depth recovered from a monocular normal map is scale ambiguous and often shifted relative to the true depth. Consequently, the mesh parts may be incorrectly combined, with significant overlap, due to inaccurate depth information, resulting in initializations with completely wrong topology. To address this, we introduce a simple yet effective concave topology-checking strategy. Specifically, for each depth map (front and back views), we calculate the average depth of the entire map ( $d_2$ ) and of a central sub-region ( $d_1$ ). If the front of the object is concave,  $d_1$  should be greater than  $d_2$  (i.e.,  $d_1 > d_2$ ) for the front view. Similarly, if the back of the object is concave,  $d_2$  will be greater than  $d_1$  for the back view. If either condition is met, the object is likely concave, and we apply sphere initialization instead. This strategy effectively enhances the robustness of the initialization process.

2) *Coarse Reconstruction*: To extract explicit 3D geometry from 2D normal maps and color images, we optimize an explicit mesh to amalgamate all 2D generated data. Nonetheless, adopting existing differentiable rendering-based optimization schemes proves unviable. These methods were tailored for real-captured images and necessitate dense input views. In contrast, our generated views are relatively sparse, and the generated normal maps and color images may exhibit subtle inaccurate predictions of some pixels. Regrettably, these errors accumulate during the geometry optimization, leading to distorted geometries, outliers, and incompleteness. To overcome the challenges above, we propose a novel geometric-aware optimization scheme.

a) *Geometric Carving*: With the initialized mesh, we leverage differentiable rendering technique to optimize 3D

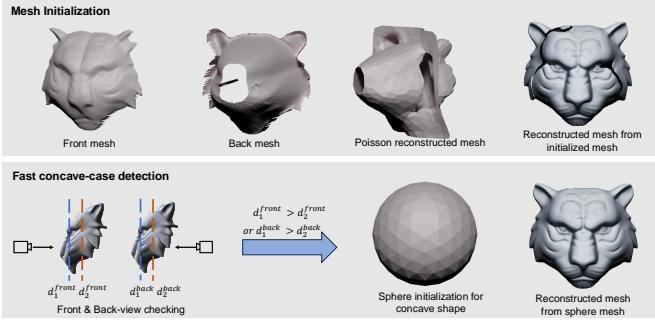


Fig. 4. The illustration of our geometric initialization strategy. We employ the Poisson reconstruction [27] method for geometric initialization using normal maps from front and back views, while estimated depth maps are used to detect and correct potential initialization errors.

shapes using the generated multi-view normals. Specifically, we render the normal maps  $N_r^{1:K}$  and masks  $M_r^{1:K}$  via a differentiable renderer, and then minimize the differences between the rendered ones and generated ones. The objective loss function is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{normal} + \mathcal{L}_{mask} + \mathcal{L}_{geo} + \mathcal{R}_{Laplace} \quad (10)$$

where  $\mathcal{L}_{normal}$  denotes the  $L_2$  loss between the rendered normals and the generated normals,  $\mathcal{L}_{mask}$  denotes  $L_2$  loss between the rendered masks and the masks segmented from the generated normals,  $\mathcal{L}_{geo}$  denotes the geometry-aware normal loss which we will discuss later, and  $\mathcal{R}_{Laplace}$  denotes a Laplace regularization term that calculates the position deviation of a vertex with its neighbors to encourage the smoothness of the mesh surface.

Through the initialization and geometric carving process mentioned above, we obtain a mesh with an approximately correct shape. However, due to the presence of several heuristic operations during the remeshing step and the challenge of ensuring perfect multi-view consistency when generating normal maps, the optimized mesh still lacks fine geometric details and smoothness.

b) *Geometry-aware Normal Loss*: Thanks to the differentiable rendering technique, we can easily extract the surface normals  $\hat{\mathbf{g}}$  of the optimized mesh by leveraging gradients of the surface with respect to its spatial coordinates. We maximize the similarity of the normal of the mesh  $\hat{\mathbf{g}}$  and our generated normal  $\mathbf{g}$  to provide 3D geometric supervision. For a 3D point  $p$  on the optimized mesh, it will be visible by multiple viewpoints but its corresponding normals of the multiple viewpoints  $\hat{\mathbf{g}}_p^{1:K}$  may have subtle variance, thus leading to trivial inaccuracies. To tolerate trivial inaccuracies of the generated normals from different views, we introduce a geometry-aware normal loss.

$$\mathcal{L}_{geo} = \sum_{p \in \mathcal{P}} w_p^i \cdot |\hat{\mathbf{g}}_p - \mathbf{g}_p^i| \quad (11)$$

where

$$w_p^i = \frac{\bar{w}_p^i}{\sum_{i=1}^K \bar{w}_p^i} \quad (12)$$

$$\bar{w}_p^i = \begin{cases} 0, & \cos(\mathbf{v}_k, \mathbf{g}_k) > \epsilon \\ m_p^i \cdot |\cos(\mathbf{v}_p^i, \hat{\mathbf{g}}_p^i)|, & \cos(\mathbf{v}_p^i, \mathbf{g}_p^i) \leq \epsilon \end{cases} \quad (13)$$

Here  $|\cdot|$  denotes an absolute function,  $m_p^i$  is the visible mask and  $\epsilon$  is a negative threshold closing to zero. For a 3D point  $p$  located in the optimized mesh, we measure the cosine value of the angle between its current normal  $\mathbf{g}_k$  and its corresponding ray's viewing direction  $\mathbf{v}_k$  across the generated multiple viewpoints.

The design rationale behind this approach lies in the orientation of normals, which are deliberately set to face outward, while the viewing direction is inward-facing. This configuration ensures that the angle between the normal vector and the viewing ray remains not less than  $90^\circ$ . A deviation from this criterion would imply inaccuracies in the generated normals. Furthermore, rather than treating normals from different views equally, we introduce a weighting mechanism. We assign higher weights to normals that its corresponding viewing ray direction maintains larger angle with the current mesh normal. This prioritization enhances the accuracy of our geometric supervision process.

c) *UV-based Texture Generation*: The prior works store texture information in mesh vertices, so that the texture quality is limited by the density of mesh vertices. In contrast, we adopt UV map to store texture information to maintain high-quality texture without information loss. Specifically, a simple yet efficient two-stage strategy is used to create a high-quality UV texture map with enhanced integrity and coherence from multi-view color images generated by the cross-domain diffusion model while mitigating the texture detail loss due to optimization.

In the first stage, we derive the vertex colors of the mesh from multi-view color images. For a vertex  $v$  on the mesh, it may be visible by several neighboring viewpoints, each corresponding to a multiple pixel value  $x^i$  in camera spaces with slight variance. The vertex color  $c$  is computed by fusing these multiple pixel values as  $c = \sum w_p^i x^i$ . Rather than treating the color information of all viewpoints equally, such a vertex texturing process takes into account the imperfect consistency of the pixel level of multi-view images by an adaptive weighting strategy. Due to the limited viewpoints and self-occlusion of the object, some vertices remain unobserved by any view (i.e., invisible vertices), resulting in texture gaps. To address this, starting from the well-colored vertices, we gradually propagate color information to adjacent uncolored vertices. During each propagation step, the color attribute of each uncolored vertex is updated with the average color of its neighboring vertices, yielding a continuous and complete texture across the meshing.

In the second stage, we unwrap the color of the vertices into UV space, yielding an initial UV texture map. Next, we project the multi-view images onto the UV space to create a UV texture, which is then blended with the initial texture to enhance and refine its details. Finally, dilation operations are operated to fill in small uncolored regions. This strategy preserves the overall texture coherence achieved through vertex colorization, while also enhancing finer details

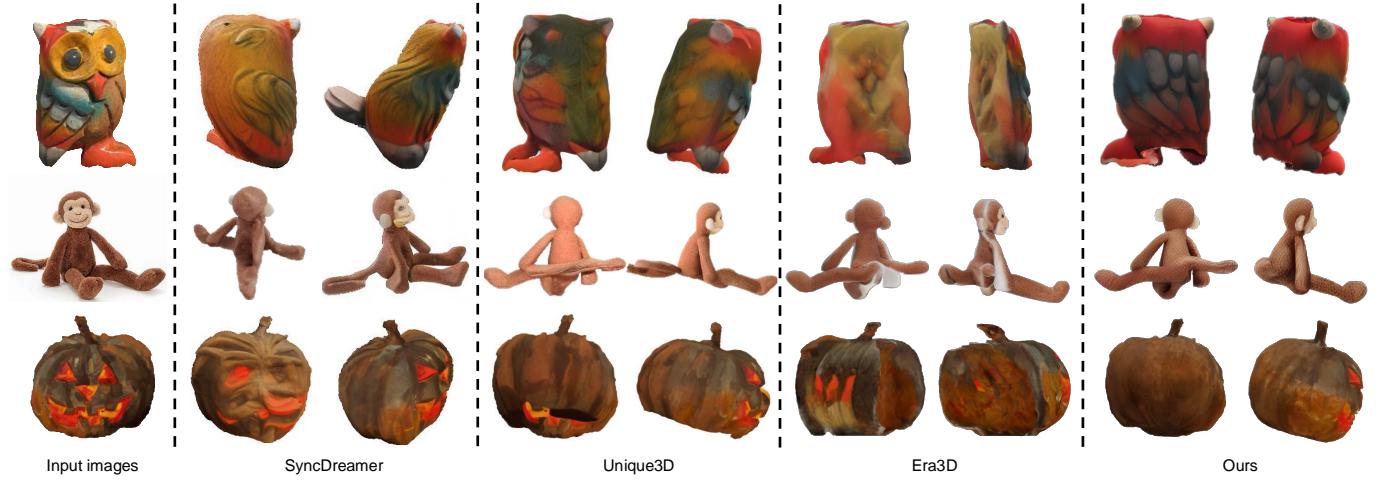


Fig. 5. The qualitative comparisons with baseline models on synthesized multi-view color images.

TABLE I  
QUANTITATIVE COMPARISON WITH BASELINE METHODS ON SINGLE VIEW RECONSTRUCTION TASK. WE REPORT CHAMFER DISTANCE AND VOLUME IOU ON THE GSO [14] DATASET ON BOTH ORTHOGONAL AND PERSPECTIVE CAMERA SETTINGS.

Method	Chamfer Dist. $\downarrow$	Volume IoU $\uparrow$
Realfusion [45]	0.0819	0.2741
Magic123 [52]	0.0516	0.4528
One-2-3-45 [35]	0.0629	0.4086
Point-E [48]	0.0426	0.2875
Shap-E [25]	0.0436	0.3584
Zero123 [36]	0.0339	0.5035
SyncDreamer [38]	0.0261	0.5421
InstantMesh [81]	0.0224	0.5353
CRM [74]	0.0220	0.5412
Unique3D [74]	0.0238	0.5134
Era3D [32]	0.0239	0.5340
Wonder3D [41]	0.0222	0.5521
Ours(Perspective Setting)	0.0206	0.6326
Ours(Orthogonal Setting)	<b>0.0193</b>	<b>0.6402</b>

in UV space, resulting in a more accurate and visually cohesive texture across the mesh surface.

3) *Iterative Refinement*: With the proposed operations above, a textured mesh is recovered from the generated multi-view color images and normal maps. Due to the mesh still containing imperfections, we name it coarse mesh and would like to further improve it. We leverage a cross-domain multi-view enhancement module to provide high-resolution 2D guidance for optimization. Specifically, as shown in Fig. 2, we render RGB images and normal maps from the coarse mesh, and subsequently the rendered images and normals are enhanced via the cross-domain multi-view enhancement module to produce high-quality images and normals with a  $512 \times 512$  resolution. The enhanced high-resolution images and normals are then used to improve the coarse mesh via differentiable rendering.

a) *Cross-domain Multi-view Enhancement Module*: The module uses the ControlNet [90] architecture to fully leverage prior information from stable diffusion. We treat the ControlNet as a trainable module while keeping the pre-trained stable diffusion weights fixed. To maintain consistency between the enhanced normal maps and color images, we integrate

TABLE II  
THE QUANTITATIVE COMPARISON IN NOVEL VIEW SYNTHESIS. WE REPORT PSNR, SSIM [75], LPIPS [91] ON THE GSO [14] DATASET.

Method	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$
Realfusion [45]	15.26	0.722	0.283
Zero123 [36]	18.93	0.779	0.166
SyncDreamer [38]	20.05	0.798	0.146
SV3D [69]	21.26	0.880	0.080
Unique3D [78]	21.71	0.913	0.083
Era3D [32]	22.49	0.916	0.069
Wonder3D [41]	22.92	0.919	0.063
Ours-(Perspective Setting)	23.23	0.929	0.054
Ours-(Orthogonal Setting)	<b>23.95</b>	<b>0.933</b>	<b>0.051</b>

cross-domain attention into the ControlNet. Low-quality RGB images and normal maps are input into the enhancement module, producing corresponding high-quality outputs with 2x resolution upsampling and enhancement. Additionally, the original input image is processed through an IP-Adapter [83] to incorporate textual information.

b) *Iterative High-resolution Refinement*: We first render normal maps and color images of the currently optimized mesh, feeding these into the cross-domain multi-view enhancement module to produce enhanced outputs. Unlike in the training stage, the low-resolution color images and normal maps generated in Sec. IV-A are used as conditioning inputs for the ControlNet branch, while Gaussian noise is added to the rendered outputs, which are then fed into the stable diffusion branch for denoising. This process, known as DDIM inversion, establishes a robust starting point for the iterative denoising process, enabling the refinement of surface imperfections and blurring on the existing mesh. With these enhanced multi-view results, we iteratively refine geometry and texture through the proposed reconstruction scheme (see Sec. IV-B2). Ultimately, this approach produces a high-quality textured mesh with intricate geometric and visual details.

## V. EXPERIMENTS

### A. Implementation Details

We train our model on a high-quality single object subset of the Objaverse dataset [10], which comprises approximately 100k objects following a cleanup process. To support the

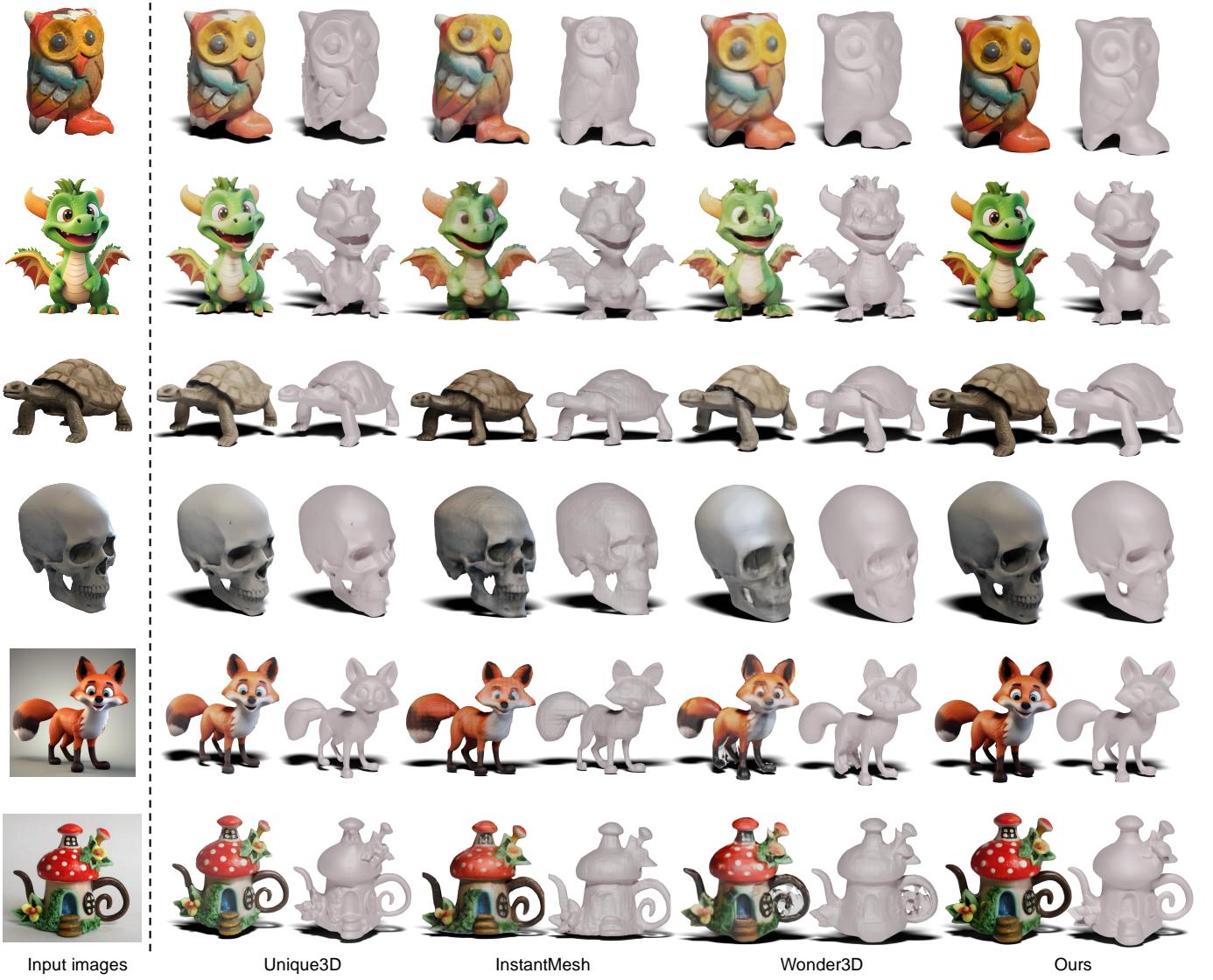


Fig. 6. The qualitative comparisons with baseline methods on in-the-wild images in terms of the reconstructed textured meshes.

model in learning consistent representations across diverse camera settings, we create the rendered augmented multi-view dataset. We first normalized each object to be centered and on a unit scale. Then we render normal maps and color images from six views, including the front, back, left, right, front-right, and front-left views, using Blenderproc [12] under both orthogonal and perspective camera settings.

We fine-tune our multi-view cross-domain model starting from the Stable Diffusion Image Variations Model, which has previously been fine-tuned with image conditions. During fine-tuning, we use a reduced image size of  $256 \times 256$  and a total batch size of 512 for training. The fine-tuning process involves training the model for 80,000 steps. This entire training procedure typically requires approximately 6 days on a cluster of 8 Huawei Kunpeng 910B GPUs.

For training the cross-domain enhancement module, we use rendered multi-view images at  $512 \times 512$  resolution as targets, enabling the model to learn robust multi-view cross-domain enhancement capabilities. The module was trained for 80,000

steps with a batch size of 24 on 8 Huawei Kunpeng 910B GPUs, completing in approximately two days.

### B. Baselines

We adopt a comprehensive set of baseline methods across three key categories: SDS-based methods, multi-view-based methods, and feed-forward methods. For SDS-based methods, RealFusion [45] and Magic123 [52] leverage Score Distillation Sampling (SDS) loss for guiding single-view 3D reconstruction from an input image. In the MV-based category, Zero123 [36] generates novel views from arbitrary perspectives, which can be further combined with SDS loss [51] to perform 3D reconstruction. Additionally, SyncDreamer [38], Era3D [32], and Wonder3D [41] produce multi-view consistent images, using the Neus [71] representation to reconstruct 3D geometry. In contrast, Unique3D [78] directly optimizes mesh vertices for a reconstruction approach. For feed-forward methods, Point-E [48] and Shap-E [25] are 3D generative models trained on a large internal OpenAI 3D dataset, both of which

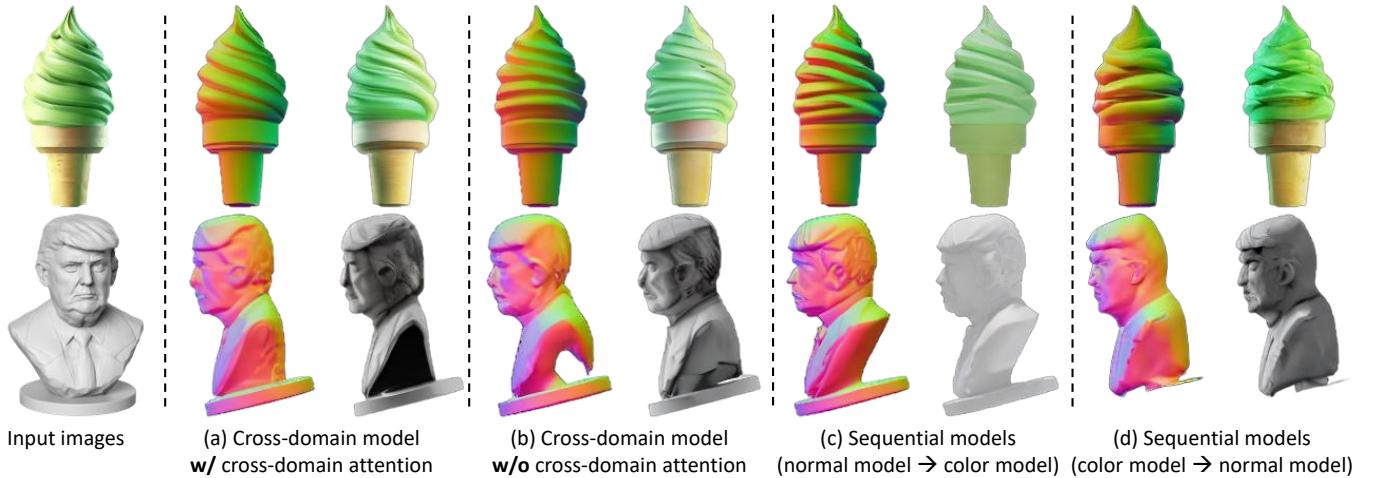


Fig. 7. Ablation studies on different cross-domain diffusion schemes. Jointly predicting multi-view normal maps and color images provides a more integrated perception of information across domains. Additionally, the proposed cross-domain attention mechanism enhances the consistency between the generated normal maps and color images, ensuring improved alignment and quality in the outputs.

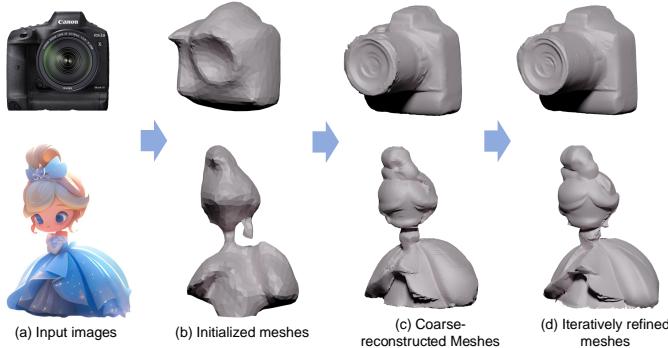


Fig. 8. Visualization of our cascaded 3D mesh extraction method shows how the geometry, from general topology to fine surface details, is captured in a coarse-to-fine manner, with subtle geometric errors corrected iteratively.

are able to convert a single-view image into a point cloud or an implicit representation. One-2-3-45 [35], CRM [74] and InstantMesh [81] first generate multi-view results and then leverage a sparse-view feed-forward reconstruction model to deliver a textured mesh using the representation of triplane or implicit SDF representation.

### C. Evaluation Protocol

**Evaluation Datasets.** Following prior research [36], [38], we adopt the Google Scanned Object dataset [14] for our evaluation, which includes a wide variety of common everyday objects. Our evaluation dataset aligns with that of SyncDreamer [38], comprising 30 objects spanning from everyday items to animals. For each object in the evaluation set, we render an image with a size of 256×256, which serves as the input. Additionally, we include in-the-wild images with diverse styles, collected from the internet or generated using text-to-image models, in our evaluation.

**Metrics.** To evaluate the quality of the single-view reconstructions, we adopt two metrics: Chamfer Distances (CD) and Volume IoU between ground-truth shapes and reconstructed shapes on both orthogonal and perspective settings of our

method. Since different methods adopt various canonical systems, we first align the generated shapes to the ground-truth shapes before calculating the two metrics. Moreover, we adopt the metrics PSNR, SSIM [75] and LPIPS [91] for evaluating the generated color images.

### D. Single View Reconstruction

We compare the quality of the reconstructions with SOTA methods, the quantitative results are summarized in Table I, and the qualitative comparisons are presented in Fig. 6. Rather than modeling multi-view images and normal maps distribution jointly, Unique3D [78] sequentially generate multi-view colors and normal maps. This exacerbates the geometric inconsistency since the stage 2 model has a huge domain gap in the inputs during training and in the inference period (see Section V-F for a more detailed discussion). As a result, vertices are updated by unstable and incorrect gradients during the optimization process, destabilizing the process and ultimately leading to a reconstructed object with incorrect shapes. Leveraging the generalization performance of the multi-view diffusion model and sparse-view reconstruction model, InstantMesh [81] produces results with reasonable overall geometry. However, due to the limited resolution of the triplane representation, the generated geometry and textures lack sufficient detail and exhibit artifacts aligned perpendicular to the coordinate axes. Wonder3D [41] takes relatively low-resolution multi-view images and normal maps to reconstruct 3D objects using implicit SDF representation without an enhancement process, it also struggles to obtain fine geometric and texture details.

Thanks to our multi-stage training scheme and cascaded 3D mesh extraction, our method produces textured meshes with a well-defined overall shape and superior geometric and textural quality compared to baseline approaches. Furthermore, geometric quantization comparisons on the GSO dataset [14] highlight the generalization ability and effectiveness of our method (see Table I).

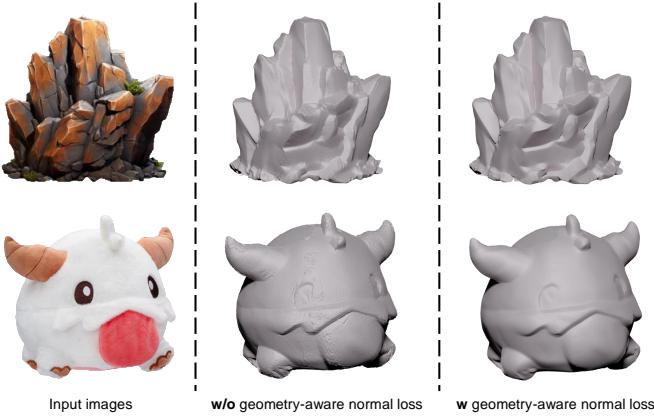


Fig. 9. Geometric visualization of the effect of our geometry-aware normal loss. Our geometry-aware normal loss effectively mitigates inaccuracies in the generated normal maps, enhancing the surface details of the meshes.

### E. Novel View Synthesis

We evaluate the quality of novel view synthesis for different multi-view-based methods. The quantitative results are presented in Table II, and the qualitative results can be found in Figure 5. SyncDreamer [36] introduces a volume attention scheme to enhance the consistency of multi-view images, their model is sensitive to the elevation degrees of the input images and tends to produce unreasonable results. Unique3D [78] stitching together the multiple views into a single image to generate multi-view results. This method severely limits the number of viewpoints it can produce (up to four). Moreover, generating images and normal maps in a sequential manner limits the generalizability and cross-view consistency of the model. Era3D [32] proposes row-wise attention to ensure multi-view consistency and estimates the input camera focal length to normalize the output results into an orthogonal coordinate system. Although it generates results with a higher resolution, it is difficult to stably output reasonable multi-view results for in-the-wild images.

In contrast, by learning information from both the color and normal domains simultaneously and exchanging information across domains via our cross-domain attention layers, our cross-domain multi-view diffusion model effectively learns the joint distribution of multi-view images and normal maps through our proposed multi-stage training scheme. This allows it to stably generate high-fidelity, multi-view consistent results.

### F. Discussions

In this section, we perform a series of experiments to validate the effectiveness of the designs in our method.

**Cross-Domain Diffusion.** To validate the effectiveness of our proposed cross-domain diffusion scheme, we study the following settings: (a) cross-domain model with cross-domain attention; (b) cross-domain model without cross-domain attention; (c) sequential models normal-to-rgb: first train a multi-view color diffusion model then train a multi-view normal diffusion model conditioned on the previously generated color images; (d) sequential models rgb-to-normal: first train a multi-view normal diffusion model then train a multi-view

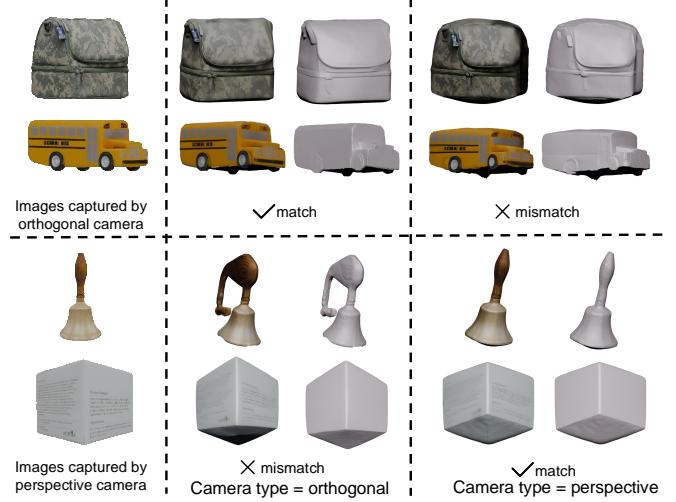


Fig. 10. Ablation study on our camera type switcher. The camera type switcher enables the generation process across various projection types, thereby enhancing the robustness of our method.

color diffusion model conditioned on the previously generated normal images.

As shown in (a) and (b) of Figure 7, it's evident that the cross-domain attentions significantly enhance the consistency between color images and normals, particularly in terms of the detailed geometries of objects like the ice-cream and the sculpture. For the sequential model normal-to-rgb, conditioning on the separately generated normal maps, the generated color images exhibit color aberrations in comparison to the input image, as shown in (c) of Figure 7. Conversely, for the sequential model rgb-to-normal, conditioning on the separately generated color images, the normal maps give unreasonable geometry, as illustrated in (d) of Figure 7. The result further substantiates our hypothesis in Sec. IV-A, confirming that the domain gap in stage 2 does contribute to performance degradation. These experiments also indicate that jointly predicting normal maps and color images via the cross-domain attention mechanism enables a more integrated perception of information across domains, effectively enhancing the model's ability to capture consistent details and address domain-specific discrepancies.

**Cascaded 3D Mesh Reconstruction.** As visualized in Figure 8, we present the results of each stage of our proposed cascaded pipeline to demonstrate the effectiveness. (1) In the geometric initialization stage, we check whether the target mesh has a concave overall geometry to decide between using a sphere mesh or a Poisson-reconstructed mesh as the starting point for optimization. This ensures that the initial mesh attains an approximate but correct geometric topology, providing a robust foundation for efficient optimization and enabling our method to handle meshes with complex topologies (see (b) in Figure 8). (2) The inconsistency-aware coarse reconstruction method derives coarse textured meshes that capture preliminary geometric details and model multi-view inaccuracies and blurry explicit on the mesh (see Figure 8). (3) Finally, within the iterative refinement stage, enhanced multi-view outputs are employed to further refine the mesh's geometry and texture. As a result, the mesh surface gains additional

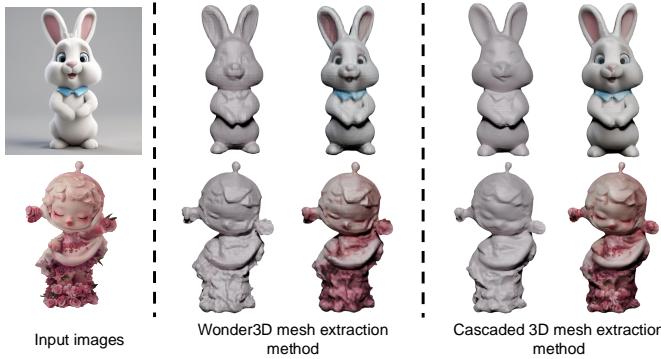


Fig. 11. The qualitative comparisons of reconstruction method between the geometry fusion algorithm proposed by Wonder3D [41] and cascaded 3D mesh extraction method. Our approach has significant advantages in both geometric and textural details

detail, effectively correcting geometric errors introduced by multi-view inaccuracy (see (d) in Figure 8).

**Discussion with Wonder3D.** We also include a comparison about the reconstruction strategy with wonder3D. As shown in Figure 11, our approach produces smoother, more detailed meshes with enhanced geometric and texture fidelity. This improvement is attributed to: 1) the cascading structure, which supports a coarse-to-fine 3D object extraction, avoid detail loss that often occurs during transformations between mesh and SDF representation; and 2) our cross-domain multi-view enhancement module, which iteratively refines both geometry and texture, achieving higher resolution while correcting viewpoint inconsistencies, resulting in superior overall quality.

**Geometry-aware Normal Loss.** To validate the effectiveness of our proposed geometry-aware normal loss, we conduct an experiment to reconstruct geometry with or without such loss. As shown in Figure 9, the reconstructed geometry with geometry-aware loss presents more geometric details and avoids surface irregularities caused by inconsistent gradients. This is because the proposed loss could further alleviate the inaccuracies among the generated multi-view normal maps, enhancing the geometric detail of the mesh surface.

**Camera Type Switcher.** We perform experiments to validate the effectiveness of the camera type switcher, which allows our model to process input images captured by both perspective and orthogonal cameras. As shown in Figure 10, when input images from either orthogonal or perspective cameras are processed with an incorrectly configured camera type switcher, the reconstruction results exhibit significant distortions and degraded texture quality. These experiments confirm that our proposed camera type switcher enables the model to achieve robust 3D generation across different imaging systems.

**Multi-stage Training Strategy.** To discuss the efficiency of our multi-stage training strategy, we conduct an experiment that omits the multi-domain pre-training phase and directly fine-tunes the diffusion model on the cross-domain multi-view generation task. The qualitative results in Figure 12 clearly demonstrate that incorporating multi-domain pre-training significantly improves both the generalization of structural consistency and the fidelity of generated multi-view outputs. Notably, we incorporate multi-view masks as part of the multi-domain

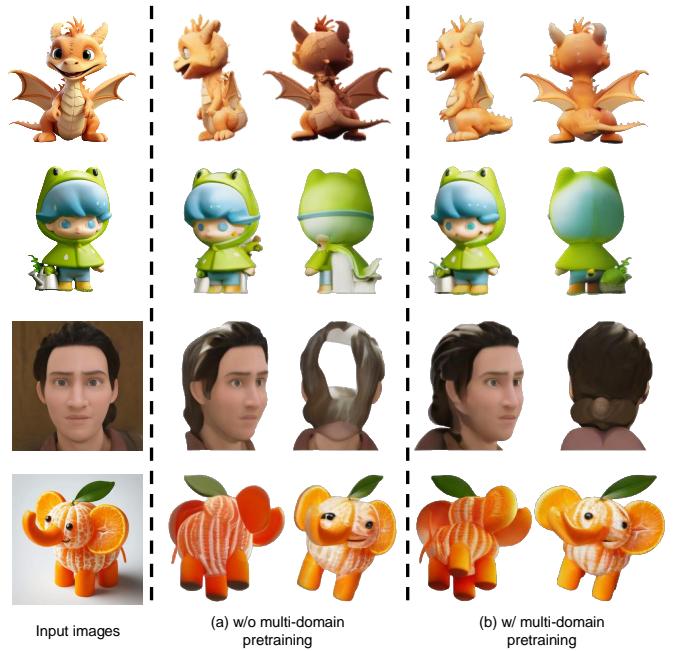


Fig. 12. Ablation study of the multi-stage training strategy. The results demonstrate that our multi-domain pretraining significantly improves both the consistency of the overall shape and the fidelity of the generated results.

pre-training, where input and output views are randomly replaced with masks. Although predicting multi-view masks is not a direct objective in the cross-domain generation task, these masks provide essential shape information about the 3D objects. This also confirms that our multi-stage training strategy effectively bridges the pre-trained image generation task and the cross-domain multi-view generation task in a step-by-step manner.

## VI. CONCLUSIONS AND LIMITATIONS

In this paper, we present *Wonder3D++*, an innovative framework for efficiently generating high-fidelity textured meshes from single-view images. Given an input image, *Wonder3D++* leverages a cross-domain multi-view diffusion module to generate multi-view images and normal maps. These results are then utilized in a novel cascaded 3D mesh extraction algorithm, consisting of geometric initialization, inconsistency-aware coarse reconstruction, and an iterative refinement stage to extract textured mesh. Experimental results demonstrate that our method achieves robust generalization and produces textured meshes with superior geometric and textural quality.

While *Wonder3D++* has shown promising performance in reconstructing 3D geometry from single-view images, it encounters challenges when handling objects with highly complex geometries and severe self-occlusion. This limitation arises from the restricted number of viewpoints and the inherent constraints of the mesh-based optimization method. We believe these challenges can be alleviated by increasing the number of viewpoints and adopting a more robust mesh-based reconstruction strategy.

## REFERENCES

- [1] Titas Auciukevičius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J Mitra, and Paul Guerrero. Renderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In *CVPR*, 2023.
- [2] Mohammadreza Armandpour, Huangjie Zheng, Ali Sadeghian, Amir Sadeghian, and Mingyuan Zhou. Re-imagine the negative prompt algorithm: Transform 2d diffusion into 3d, alleviate janus problem and beyond. *arXiv preprint arXiv:2304.04968*, 2023.
- [3] Eric R Chan, Koki Nagano, Matthew A Chan, Alexander W Bergman, Jeong Joon Park, Axel Levy, Miika Aittala, Shalini De Mello, Tero Karras, and Gordon Wetzstein. Generative novel view synthesis with 3d-aware diffusion models. In *ICCV*, 2023.
- [4] 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. In *ICCV*, 2023.
- [5] 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.
- [6] Yiwen Chen, Chi Zhang, Xiaofeng Yang, Zhongang Cai, Gang Yu, Lei Yang, and Guosheng Lin. It3d: Improved text-to-3d generation with explicit view synthesis. *arXiv preprint arXiv:2308.11473*, 2023.
- [7] Kai Cheng, Xiaoxiao Long, Kaizhi Yang, Yao Yao, Wei Yin, Yuexin Ma, Wenping Wang, and Xuejin Chen. Gaussianpro: 3d gaussian splatting with progressive propagation. *arXiv preprint arXiv:2402.14650*, 2024.
- [8] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *CVPR*, 2023.
- [9] Florinel-Alin Croitoru, Vlad Hondu, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *T-PAMI*, 2023.
- [10] 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 *CVPR*, 2023.
- [11] Congye Deng, Chiyu Jiang, Charles R Qi, Xinchen Yan, Yin Zhou, Leonidas Guibas, Dragomir Anguelov, et al. Nerd: Single-view nerf synthesis with language-guided diffusion as general image priors. In *CVPR*, 2023.
- [12] Maximilian Denninger, Dominik Winkelbauer, Martin Sundermeyer, Wout Boerdijk, Markus Knauer, Klaus H. Strobl, Matthias Humt, and Rudolph Triebel. Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901, 2023.
- [13] Zhiyang Dou, Qingxuan Wu, Cheng Lin, Zeyu Cao, Qiangqiang Wu, Weilin Wan, Taku Komura, and Wenping Wang. Tore: Token reduction for efficient human mesh recovery with transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15143–15155, 2023.
- [14] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *ICRA*, 2022.
- [15] Ziya Erkoç, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Hyperdiffusion: Generating implicit neural fields with weight-space diffusion. *arXiv preprint arXiv:2303.17015*, 2023.
- [16] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *NeurIPS*, 2022.
- [17] Jiatao Gu, Qingzhe Gao, Shuangfei Zhai, Baoquan Chen, Lingjie Liu, and Josh Susskind. Learning controllable 3d diffusion models from single-view images. *arXiv preprint arXiv:2304.06700*, 2023.
- [18] Jiatao Gu, Alex Trevithick, Kai-En Lin, Joshua M Susskind, Christian Theobalt, Lingjie Liu, and Ravi Ramamoorthi. Nerfdiff: Single-image view synthesis with nerf-guided distillation from 3d-aware diffusion. In *ICML*, 2023.
- [19] 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.
- [20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.
- [21] Yicong Hong, Kai Zhang, Juxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023.
- [22] Yukun Huang, Jianan Wang, Yukai Shi, Xianbiao Qi, Zheng-Jun Zha, and Lei Zhang. Dreamtime: An improved optimization strategy for text-to-3d content creation. *arXiv preprint arXiv:2306.12422*, 2023.
- [23] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876, 2022.
- [24] Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. *arXiv preprint arXiv:2311.17977*, 2023.
- [25] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023.
- [26] Animesh Karnewar, Niloy J Mitra, Andrea Vedaldi, and David Novotny. Holofusion: Towards photo-realistic 3d generative modeling. In *ICCV*, 2023.
- [27] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006.
- [28] Seung Wook Kim, Bradley Brown, Kangxue Yin, Karsten Kreis, Katja Schwarz, Daqing Li, Robin Rombach, Antonio Torralba, and Sanja Fidler. Neuralfield-ldm: Scene generation with hierarchical latent diffusion models. In *CVPR*, 2023.
- [29] Kilian Kleeberger, Richard Bormann, Werner Kraus, and Marco F Huber. A survey on learning-based robotic grasping. *Current Robotics Reports*, 1:239–249, 2020.
- [30] 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.
- [31] Jiabao Lei, Jiapeng Tang, and Kui Jia. Generative scene synthesis via incremental view inpainting using rgbd diffusion models. In *CVPR*, 2022.
- [32] Peng Li, Yuan Liu, Xiaoxiao Long, Feihu Zhang, Cheng Lin, Mengfei Li, Xingqun Qi, Shanghang Zhang, Wenhan Luo, Ping Tan, et al. Era3d: High-resolution multiview diffusion using efficient row-wise attention. *arXiv preprint arXiv:2405.11616*, 2024.
- [33] Weiyu Li, Jiarui Liu, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman: High-fidelity mesh generation with 3d native generation and interactive geometry refiner. *arXiv preprint arXiv:2405.14979*, 2024.
- [34] 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 *CVPR*, 2023.
- [35] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *arXiv preprint arXiv:2306.16928*, 2023.
- [36] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *ICCV*, 2023.
- [37] Xinhang Liu, Shiu-hong Kao, Jiaben Chen, Yu-Wing Tai, and Chi-Keung Tang. Deceptive-nerf: Enhancing nerf reconstruction using pseudo-observations from diffusion models. *arXiv preprint arXiv:2305.15171*, 2023.
- [38] 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.
- [39] Yuan Liu, Sida Peng, Lingjie Liu, Qianqian Wang, Peng Wang, Christian Theobalt, Xiaowei Zhou, and Wenping Wang. Neural rays for occlusion-aware image-based rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7824–7833, 2022.
- [40] Zhen Liu, Yao Feng, Michael J Black, Derek Nowrouzezahrai, Liam Paull, and Weiyang Liu. Meshdiffusion: Score-based generative 3d mesh modeling. In *ICLR*, 2023.
- [41] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9970–9980, 2024.
- [42] Xiaoxiao Long, Cheng Lin, Lingjie Liu, Wei Li, Christian Theobalt, Ruigang Yang, and Wenping Wang. Adaptive surface normal constraint for depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12849–12858, October 2021.

- [43] Xiaoxiao Long, Yuhang Zheng, Yupeng Zheng, Beiwen Tian, Cheng Lin, Lingjie Liu, Hao Zhao, Guyue Zhou, and Wenping Wang. Adaptive surface normal constraint for geometric estimation from monocular images. *arXiv preprint arXiv:2402.05869*, 2024.
- [44] 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.
- [45] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Realfusion: 360deg reconstruction of any object from a single image. In *CVPR*, 2023.
- [46] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020.
- [47] Norman Müller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kortschieder, and Matthias Nießner. Diffirf: Rendering-guided 3d radiance field diffusion. In *CVPR*, 2023.
- [48] 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.
- [49] Evangelos NTavelis, Aliaksandr Siarohin, Kyle Olszewski, Chaoyang Wang, Luc Van Gool, and Sergey Tulyakov. Autodecoding latent 3d diffusion models. *arXiv preprint arXiv:2307.05445*, 2023.
- [50] Werner Palfinger. Continuous remeshing for inverse rendering. *Computer Animation and Virtual Worlds*, 33(5):e2101, 2022.
- [51] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023.
- [52] Guocheng Qian, Jinjie Mai, Abdullah Hamdi, Jian Ren, Aliaksandr Siarohin, Bing Li, Hsin-Ying Lee, Ivan Skorokhodov, Peter Wonka, Sergey Tulyakov, et al. Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. *arXiv preprint arXiv:2306.17843*, 2023.
- [53] 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 *ICML*, 2021.
- [54] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. *arXiv preprint arXiv:2303.13508*, 2023.
- [55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.
- [56] Hoigi Seo, Hayeon Kim, Gwanghyun Kim, and Se Young Chun. Ditto-nerf: Diffusion-based iterative text to omni-directional 3d model. *arXiv preprint arXiv:2304.02827*, 2023.
- [57] Junyoung Seo, Wooseok Jang, Min-Seop Kwak, Jaehoon Ko, Hyeyoung Kim, Junho Kim, Jin-Hwa Kim, Jiyoung Lee, and Seungryong Kim. Let 2d diffusion model know 3d-consistency for robust text-to-3d generation. *arXiv preprint arXiv:2303.07937*, 2023.
- [58] QiuHong Shen, Xingyi Yang, and XinChao Wang. Anything-3d: Towards single-view anything reconstruction in the wild. *arXiv preprint arXiv:2304.10261*, 2023.
- [59] 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.
- [60] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015.
- [61] Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Viewset diffusion(0-) image-conditioned 3d generative models from 2d data. *arXiv preprint arXiv:2306.07881*, 2023.
- [62] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. *arXiv preprint arXiv:2402.05054*, 2024.
- [63] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *ICCV*, 2023.
- [64] 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.
- [65] Ayush Tewari, Tianwei Yin, George Cazenavette, Semon Reznikov, Joshua B Tenenbaum, Frédéric Durand, William T Freeman, and Vincent Sitzmann. Diffusion with forward models: Solving stochastic inverse problems without direct supervision. *arXiv preprint arXiv:2306.11719*, 2023.
- [66] Christina Tsaligoglou, Fabian Manhardt, Alessio Tonioni, Michael Niemeyer, and Federico Tombari. Textmesh: Generation of realistic 3d meshes from text prompts. *arXiv preprint arXiv:2304.12439*, 2023.
- [67] Hung-Yu Tseng, Qinbo Li, Changil Kim, Suhib Alsisan, Jia-Bin Huang, and Johannes Kopf. Consistent view synthesis with pose-guided diffusion models. In *CVPR*, 2023.
- [68] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- [69] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. *arXiv preprint arXiv:2403.12008*, 2024.
- [70] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In *CVPR*, 2023.
- [71] 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. In *NeurIPS*, 2021.
- [72] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. In *CVPR*, 2023.
- [73] 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.
- [74] Zhengyi Wang, Yikai Wang, Yifei Chen, Chendong Xiang, Shuo Chen, Dajiang Yu, Chongxuan Li, Hang Su, and Jun Zhu. Crm: Single image to 3d textured mesh with convolutional reconstruction model. *arXiv preprint arXiv:2403.05034*, 2024.
- [75] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *TIP*, 2004.
- [76] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022.
- [77] Jinbo Wu, Xiaoobo Gao, Xing Liu, Zhengyang Shen, Chen Zhao, Haocheng Feng, Jingtuo Liu, and Errui Ding. Hd-fusion: Detailed text-to-3d generation leveraging multiple noise estimation. *arXiv preprint arXiv:2307.16183*, 2023.
- [78] Kailu Wu, Fangfu Liu, Zhihan Cai, Runjie Yan, Hanyang Wang, Yating Hu, Yueqi Duan, and Kaisheng Ma. Unique3d: High-quality and efficient 3d mesh generation from a single image. *arXiv preprint arXiv:2405.20343*, 2024.
- [79] Jianfeng Xiang, Jiaolong Yang, Binbin Huang, and Xin Tong. 3d-aware image generation using 2d diffusion models. *arXiv preprint arXiv:2303.17905*, 2023.
- [80] Dejia Xu, Yifan Jiang, Peihao Wang, Zhiwen Fan, Yi Wang, and Zhangyang Wang. Neurallift-360: Lifting an in-the-wild 2d photo to a 3d object with 360 views. *arXiv e-prints*, pages arXiv–2211, 2022.
- [81] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.
- [82] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv preprint arXiv:2403.14621*, 2024.
- [83] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023.
- [84] Paul Yoo, Jiaxian Guo, Yutaka Matsuo, and Shixiang Shane Gu. Dreamsparse: Escaping from plato’s cave with 2d frozen diffusion model given sparse views. *CoRR*, 2023.
- [85] Chaohui Yu, Qiang Zhou, Jingliang Li, Zhe Zhang, Zhibin Wang, and Fan Wang. Points-to-3d: Bridging the gap between sparse points and shape-controllable text-to-3d generation. *arXiv preprint arXiv:2307.13908*, 2023.
- [86] Jason J. Yu, Fereshteh Forghani, Konstantinos G. Derpanis, and Marcus A. Brubaker. Long-term photometric consistent novel view synthesis with diffusion models. In *ICCV*, 2023.
- [87] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *NeurIPS*, 2022.

- [88] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. In *SIGGRAPH*, 2023.
- [89] Jingbo Zhang, Xiaoyu Li, Ziyu Wan, Can Wang, and Jing Liao. Text2nerf: Text-driven 3d scene generation with neural radiance fields. *arXiv preprint arXiv:2305.11588*, 2023.
- [90] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.
- [91] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.
- [92] Yuhang Zheng, Xiangyu Chen, Yupeng Zheng, Songen Gu, Runyi Yang, Bu Jin, Pengfei Li, Chengliang Zhong, Zengmao Wang, Lina Liu, et al. Gaussiangrasper: 3d language gaussian splatting for open-vocabulary robotic grasping. *arXiv preprint arXiv:2403.09637*, 2024.
- [93] 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.
- [94] Zhizhuo Zhou and Shubham Tulsiani. Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In *CVPR*, 2023.
- [95] Joseph Zhu and Peiyi Zhuang. Hifa: High-fidelity text-to-3d with advanced diffusion guidance. *arXiv preprint arXiv:2305.18766*, 2023.