MeshGen: Generating PBR Textured Mesh with Render-Enhanced Auto-Encoder and Generative Data Augmentation

Anonymous authors
Paper under double-blind review

Abstract

In this paper, we present MeshGen, an advanced image-to-3D pipeline designed to generate high-quality 3D objects with physically based rendering (PBR) textures. Existing methods struggle with issues such as poor auto-encoder performance, limited training datasets, misalignment between input images and 3D shapes, and inconsistent image-based PBR texturing. MeshGen addresses these limitations through several key innovations. First, we introduce a render-enhanced point-to-shape auto-encoder that compresses 3D shapes into a compact latent space, guided by perceptual loss. A 3D-native diffusion model is then established to directly learn the distribution of 3D shapes within this latent space. To mitigate data scarcity and image-shape misalignment, we propose geometric alignment augmentation and generative rendering augmentation, enhancing the diffusion model's controllability and generalization ability. Following shape generation, MeshGen applies a reference attention-based multi-view ControlNet for image-consistent appearance synthesis, complemented by a PBR decomposer to separate PBR channels. Extensive experiments demonstrate that MeshGen significantly enhances both shape and texture generation compared to previous methods.

1 Introduction

With the rapid advancement of diffusion-based image generation models, there has been significant progress in automatic 3D generation. In particular, methods utilizing score distillation sampling (Poole et al., 2023) have demonstrated breakthroughs by leveraging priors from text-to-image diffusion models. However, these optimization-based methods are relatively slow and face challenges such as mode collapse (Wang et al., 2023a;c) and the Janus problem (Armandpour et al., 2023; Seo et al., 2023) due to the lack of inherent 3D information. Subsequent strategies address these challenges by focusing on multi-view generation (Liu et al., 2023c; Long et al., 2023; Chen et al., 2024c; Voleti et al., 2024) and large reconstruction models (Zou et al., 2023; Tang et al., 2024a; Hong et al., 2023; Li et al., 2023a; Xu et al., 2024b; Liu et al., 2024; Wei et al., 2024). The former generates multi-view images for 3D reconstruction. The latter maps sparse view images to compact 3D representations using neural networks, such as triplane NeRF (Chan et al., 2021) or grid 3D Gaussians (Zou et al., 2023; Tang et al., 2024a). While these methods have improved the quality and speed of 3D generation, they typically use volumetric representations such as NeRF or Gaussian instead of 3D meshes, resulting in further loss of quality during conversion (Hong et al., 2023; Chen et al., 2024b; Tang et al., 2023). Moreover, these methods, which rely solely on render loss for supervision, are highly susceptible to inconsistencies across multiple synthesized views and often struggle to reconstruct objects with complex geometric structures (Sun et al., 2024).

Recently, 3D native diffusion methods have garnered significant attention as a promising paradigm towards mesh-oriented generation (Gupta et al., 2023; Wang et al., 2023b; Zhang et al., 2024a; Li et al., 2024b; Wu et al., 2024b; Hong et al., 2024; Chen et al., 2024a). By mapping 3D meshes into a compact latent space using 3D auto-encoders, these methods directly learn the distribution of 3D shapes instead of reconstructing from generated multi-views. Despite considerable progress has been made, several challenges remain unresolved. Firstly, the inherent limitations of current 3D auto-encoders preclude the integration of perceptual loss during training, leading to less detailed reconstructed meshes and consequently constrained expressiveness of the latent space. Moreover, existing 3D native diffusion methods typically generate simple and symmetric shapes, making it challenging to match the input images, and the scarcity and poor quality of public datasets further limit the generalization ability of current open-source models. In addition, existing image-guided texture generation methods struggle to produce appearances consistent with the original images and can only generate materials with light baked in, rather than the physically based rendering (PBR) materials required in practical applications.

In response to these challenges, we introduce MeshGen, a novel image-to-3D pipeline specially designed to generate PBR textured meshes that closely resemble the provided image in both geometry and appearance. Specifically, to enhance the expressiveness of the point-to-shape auto-encoder, we propose a triplane-based auto-encoder that incorporates perceptual render loss during training, thereby fully exploiting the memory efficiency of triplane compared to latent vector set representation. Next, based on the geometrical covariant property of the point-to-shape auto-encoder and the appearance-invariant nature of image-to-shape diffusion, we establish an image-to-shape diffusion model with geometric alignment and generative rendering augmentation to enhance image-shape consistency and generalization ability. For texture generation, we propose using a geometry-conditioned ControlNet with reference attention fine-tuning to generate multi-view images consistent with the input image in both appearance and lightning. We then employ a PBR decomposer to estimate the PBR components in the shaded image and a texture inpainter to fill in the invisible parts. As a result of these advancements, MeshGen can generate PBR textured 3D assets with consistent geometry and exceptional fidelity within 30 seconds.

To summarize, our contributions are:

- We propose the MeshGen auto-encoder, which substantially improves the expressiveness of the point-to-shape auto-encoder by incorporating both geometric and appearance supervision. It utilizes a coarse-to-fine optimization strategy guided by render-based perceptual loss, ensuring more accurate shape representation.
- We introduce a novel image-to-shape pipeline with our proposed geometric alignment augmentation and generative rendering augmentation, which largely enhance image-shape alignment and generalization capabilities.
- We design a reference attention-based image-conditioned mesh texturing pipeline. Coupled with our proposed PBR decomposer, our method is capable of generating relightable textures that closely align with the appearance of the input image.

2 Related Work

2.1 3D Generation

Early efforts in 3D generation focus on per-scene optimization methods based on CLIP similarity (Radford et al., 2021; Sanghi et al., 2021; Jain et al., 2022) and score distillation sampling (Poole et al., 2023). By utilizing powerful pre-trained image diffusion models, these methods soon excel in various 3D generation tasks (Wang et al., 2023c; Chen et al., 2023b; Lin et al., 2023; Tang et al., 2023; Chen et al., 2024b; Shi et al., 2023b; Li et al., 2023c; Wang & Shi, 2023; Sun et al., 2023; Chen et al., 2023c). Despite great success has been achieved, optimization-based methods still suffer from slow generation speed and low success rates. To overcome these challenges, researchers have explored multi-view generation (Liu et al., 2023b; Tang et al., 2024b; Lu et al., 2023; Liu et al., 2023c; Long et al., 2023; Wu et al., 2024a; Li et al., 2024a; Chen et al., 2024c; Voleti et al., 2024) and large reconstruction models (Szymanowicz et al., 2023; Liu et al., 2023e; Xu et al., 2023; 2024a; Hong et al., 2023; Li et al., 2023b;a; Tang et al., 2024a; Wang et al., 2024). InstantMesh (Xu et al., 2024b) adopts a two-stage optimization strategy that firstly trains a multi-view to triplane NeRF model, then uses this model as initialization for FlexiCubes (Shen et al., 2023), thus yielding direct textured mesh reconstruction from images. MeshLRM (Wei et al., 2024) follows a similar pipeline but uses differentiable marching cubes with deferred rendering for direct mesh output. MeshFormer (Liu et al., 2024) utilizes a hierarchical voxel structure for efficient large reconstruction model training. Although these methods have advanced 3D generation in speed and quality, the unsatisfying performance of multi-view generation and the growing demands for higher mesh quality have led researchers to focus increasingly on the development of native 3D generation methods (Liu et al., 2023a; Gupta et al., 2023; Chen et al., 2024a; Wang et al., 2023b; Ren et al., 2024). 3DTopia trains

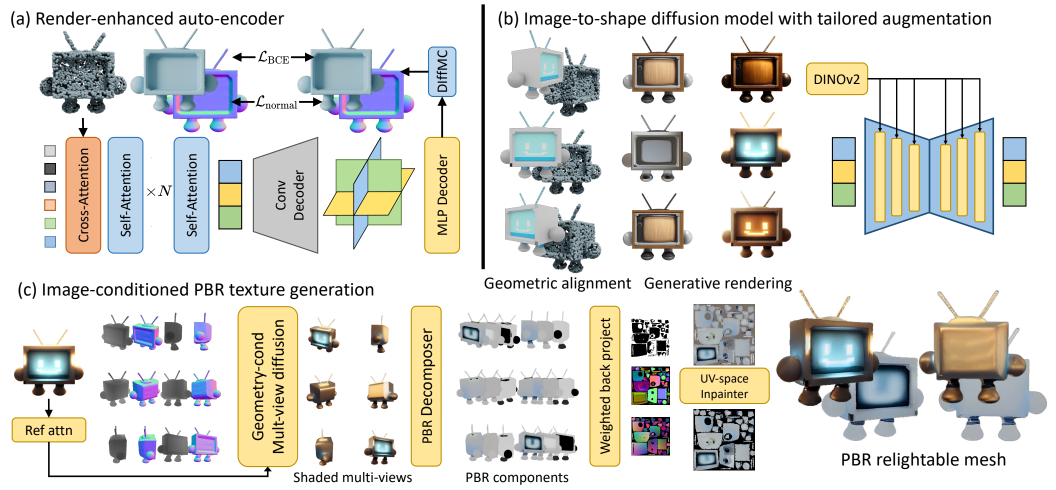

Figure 1: Overview of the proposed MeshGen. We first train a render-enhanced auto-encoder to compress 3D meshes to more compact latent space (Section. 3.1). We establish an image-to-shape diffusion model based on our tailored generative augmentations for improving image-shape alignment and generalization ability (Section. 3.2). The obtained mesh undergoes a reference attention-based multi-view synthesis and a PBR decomposer to obtain multi-view PBR channels. A UV-space inpainter is then exploited to fill the areas invisible in multi-view images (Section. 3.3).

a text-to-triplane NeRF diffusion model on pre-computed latents to achieve native text-to-3D generation. 3DShape2Vecset (Zhang et al., 2023a) and CLAY (Zhang et al., 2024a) exploit latent vector sets as representation, significantly enhancing the expressiveness of the latent space. CraftsMan (Li et al., 2024b) improves 3DShape2Vecset by incorporating point normal as input to the auto-encoder and proposes a normal enhancement process to generate finer details. Direct3D (Wu et al., 2024b) exploits triplane as latent representation for capturing the structural 3D information. Our method introduces a render-enhanced auto-encoder with geometric alignment and generative rendering augmentation during training, improving the performance of 3D native diffusion in image-to-3D tasks.

2.2 Texture Generation

Initial efforts in mesh texturing have focused on only utilizing image diffusion models through iterative inpainting and optimization (Chen et al., 2023a; Jiang et al., 2024). TEXTure (Richardson et al., 2023) presents an iterative texturing method that employs a pre-trained depth-to-image diffusion model to progressively refine a 3D model's texture map from various views. TexFusion (Cao et al., 2023) enhances coherence by integrating texture information from multiple perspectives during the denoising stage. SyncMVD (Liu et al., 2023d) improves multi-view consistency by denoising in UV space and exploiting a self-attention reuse technique. FlashTex (Deng et al., 2024) proposes a light ControlNet for text-to-PBR generation. In addition to methods that use only image diffusion, various learning-based strategies initiate the training of generative texturing models using 3D textured mesh data (Nichol et al., 2022; Luo et al., 2023; Jun & Nichol, 2023; Li et al., 2022; Collins et al., 2022; Deitke et al., 2023; Chen et al., 2022; Yu et al., 2021b; Cheng et al., 2023). Texturify (Siddiqui et al., 2022) proposes a GAN-based pipeline with face convolution to colorize meshes without direct supervision. Point-UV (Yu et al., 2023) proposes a point diffusion to offer low-resolution global information and a UV diffusion for enhancing finer details. Paint3D (Zeng et al., 2023) proposes a coarse-to-fine strategy that firstly colorizes sparse views with depth-based inpainting and then improves texture quality within the UV space. Meta 3D TextureGen (Bensadoun et al., 2024) exploits a geometry-conditioned multi-view generator for text-to-texture generation.

3 Method

The overall pipeline of MeshGen is demonstrated in Fig. 1. We first train a render-enhanced auto-encoder to compress the 3D meshes into compact triplanes. A diffusion model is then established based on the proposed geometric alignment and generative rendering augmentation to enhance image-shape alignment and generalization ability. The decoded 3D mesh then undergoes our multi-view diffusion-based texturing pipeline for PBR material generation. The detailed MeshGen methodology is presented as follows.

3.1 RENDER-ENHANCED AUTO-ENCODER

Transformer-based point-to-shape auto-encoder. To compress the discrete 3D meshes into a continuous latent space, we adopted the same encoder as used in prior native 3D generation approaches (Zhang et al., 2023a; 2024a; Li et al., 2024b), namely the point-to-shape encoder. For a given 3D object, we first uniformly sample N_P points from its surface. Following previous methods, we encode the sampled point cloud using Fourier positional encoding (Rahaman et al., 2019). Subsequently, a set of learnable queries is introduced to extract information from the point cloud through cross-attention, followed by a series of self-attentions to enhance the obtained representation. The complete encoding process can be formulated as

$$\mathbf{z} = \texttt{SelfAttn}^n(\texttt{CrossAttn}(Q, \texttt{FourierPE}(P))), \tag{1}$$

where n refers to the number of self-attention layers, `SelfAttn`, `CrossAttn` and `FourierPE` represents self-, cross-attention, and Fourier positional encoding. Here $Q \in \mathbb{R}^{N_z \times d_z}$ and $P \in \mathbb{R}^{N_P \times 3}$ represent the learnable query set and the sampled point cloud respectively, N_z and d_z refer to the number of learnable queries and the dimension of the latent space. To incorporate render-based perceptual loss during auto-encoder training, we choose triplane as the latent representation (Wu et al., 2024b) instead of the latent vector set used in 3DShape2Vecset (Zhang et al., 2023a). This choice is motivated by the fact that when querying the occupancy, the latent vector set requires cross-attention with all latents, whereas the triplane only needs to pass through an MLP decoder, thus supporting surface extraction at a higher resolution. To obtain the occupancy of a specific point, a convolutional decoder is applied to upsample the encoded latent to a higher resolution to represent finer details. As analyzed in (Wang et al., 2023b; Wu et al., 2024b), we concatenate the three planes in the height dimension instead of the channel dimension to avoid artifacts caused by spatial misalignment. The occupancy of point $\mathbf{x}$ can be formulated as

$$\text{Occupancy}(\mathbf{x}) = \text{MLP}(\texttt{UpSample}(\mathbf{z}_{\text{tile}}), \mathbf{x}), \tag{2}$$

where `Upsample` denotes the convolution-based upsampling network, MLP refers to the occupancy decoder, $\mathbf{z}_{\text{tile}}$ represents the height-concatenated triplane. As suggested in Odena et al. (2016), we use interpolation with convolution instead of deconvolution.

Perceptual loss with ray-based regularization. Previous point-to-shape auto-encoder relies solely on occupancy loss, the absence of perceptual loss leads to poor performance when reconstructing high-frequency details. In response, we propose supervising the auto-encoder using the rendered normal map. During training, we query the occupancy of a 256^3 grid and extract iso-surface differentiably (Wei et al., 2023). To compute the render loss, we exploit nvdiffrast (Laine et al., 2020) to differentiably rasterize the normal map. However, we found in our early experiment that simply applying render loss alone will cause severe floaters in the final output mesh (see Fig. 8), which can also be observed in previous research (Wei et al., 2024). To address this issue, we propose a ray-based occupancy regularization that forces the occupancy in empty spaces to approach zero. As shown in the left part of Fig. 2, for each camera ray, we uniformly sample N_s points between the ray-bounding box intersection and the surface point, enforcing their occupancy to be near zero by minimizing the sum of their occupancy. To save GPU VRAM and accelerate training, we interpolate the occupancy of the samples from the values used for previous surface extraction, rather than querying the triplane.

Coarse-to-fine optimization. Due to the locality of differentiable marching cubes, the gradients of the render loss can only propagate to points near surface vertices. Therefore, a coarse-to-fine training process is required to ensure the effectiveness of the render loss. Specifically, during the coarse stage, we apply the standard binary cross-entropy (BCE) loss for the point-to-shape auto-encoder, along with a KL loss to regularize the latent space and a total variation loss (Yu et al., 2021a) for reducing the floaters, i.e.

$$\mathcal{L}_{\text{coarse}} = \mathcal{L}_{\text{BCE}} + \lambda_{\text{KL}}\mathcal{L}_{\text{KL}} + \lambda_{\text{TV}}\mathcal{L}_{\text{TV}}, \tag{3}$$

where $\mathcal{L}_{\text{BCE}}$, $\mathcal{L}_{\text{KL}}$ and $\mathcal{L}_{\text{TV}}$ denote the BCE loss, the KL loss and the total variation loss respectively, λ_{KL} and λ_{TV} refers to the loss weights. After the coarse stage training, the model is capable of

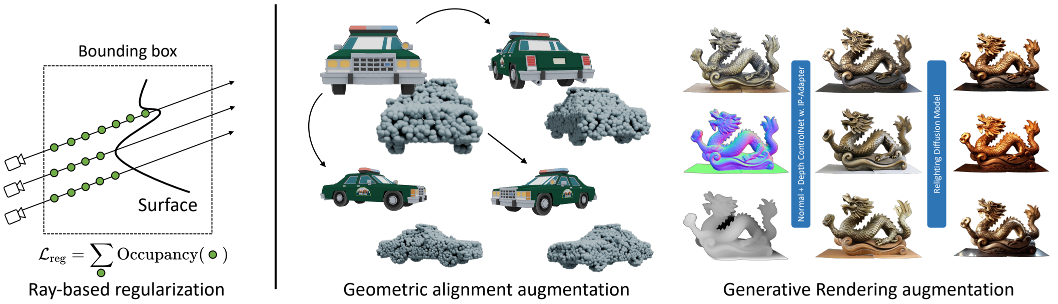
Bounding box
Surface
$\mathcal{L}_{\text{reg}} = \sum_{●} \text{Occupancy}(●)$
Ray-based regularization
Geometric alignment augmentation
Normal + Depth ControlNet w. IP-Adapter
Relighting Diffusion Model
Generative Rendering augmentation

Figure 2: Illustration of the proposed ray-based regularization and two data augmentations.

reconstructing a coarse mesh from the input point cloud. In the refinement stage, we exploit render loss with ray-based regularization to enhance the details of the reconstructed mesh:

$$\mathcal{L}_{\text{refine}} = \mathcal{L}_{\text{BCE}} + \lambda_{\text{KL}}\mathcal{L}_{\text{KL}} + \lambda_{\text{TV}}\mathcal{L}_{\text{TV}} + \lambda_{\text{MSE}}\mathcal{L}_{\text{normal}}^{\text{MSE}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{normal}}^{\text{LPIPS}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}, \tag{4}$$

where $\mathcal{L}_{\text{normal}}^{\text{MSE}}$ and $\mathcal{L}_{\text{normal}}^{\text{LPIPS}}$ denotes the MSE and LPIPS (Zhang et al., 2018) loss for rendered normal, λ_{MSE}, λ_{LPIPS}, and λ_{reg} refers to the corresponding loss weight. The concrete hyper-parameter settings are presented in appendix A.

3.2 Image-to-shape diffusion model with generative data augmentation

As shown in Fig. 4 and Fig. 5, compared to large reconstruction models, existing native 3D generation models almost always tend to generate symmetrical shapes that lack detail, leading to image-shape misalignment. We believe this phenomenon arises because the training of existing models predominantly relies on the Objaverse dataset (Deitke et al., 2022), and a considerable portion of the objects in Objaverse have symmetrical geometry and lack realistic textures. Therefore, the diffusion models trained on it tend to replicate simple geometries and struggle to generalize to images with complex textures or lighting. To optimize a diffusion model with strong generalization capabilities on limited data, we identify two key differences between the proposed pipeline and the previous NeRF-based native 3D generation pipeline (typical methods include Rodin (Wang et al., 2023b), 3DTopia (Hong et al., 2024), etc): **(1) Geometrical covariant auto-encoder.** Previous NeRF-based native 3D generation methods utilize a per-object optimized neural radiance field as the latent representation. This approach requires pre-computing and storing numerous latent vectors and lacks geometric covariance, as it necessitates re-optimizing the radiance field to obtain triplane latent variables after applying a transformation to the object. In contrast, our point-to-shape auto-encoder takes point clouds as input and does not require per-object optimization, it naturally achieves geometric covariance for transformations such as rotations by directly manipulating the point cloud. **(2) Appearance invariant image-to-shape modeling.** Previous methods learn to generate textured meshes from images, resulting in the entanglement of input images and the textures of the output meshes. In contrast, our diffusion model is specifically designed to map images to shapes, ensuring that the same shapes produce consistent renderings, regardless of variations in textures or lighting conditions. Based on both insights, we propose two data augmentations that are critical for training the image-to-shape model.

Geometric alignment augmentation. To enhance image-shape correspondence during training, we propose utilizing the geometric covariance property of our point-to-shape auto-encoder to ensure that different views of the same object correspond to different latents. Specifically, for each object in the dataset, we select one view from multi-view images as the condition and rotate the point cloud's azimuth to align the object's orientation with the selected image as the target (see the middle part of Fig. 2 for a simple demonstration). The aligned image-shape pairs are then used as training data for the diffusion model. Our experiments reveal that geometric alignment not only expands the training dataset but also significantly improves the alignment between generated shapes and images. We present the corresponding ablation study in Fig. 7.

Generative rendering augmentation. To enhance the generalization ability of the image-to-shape diffusion, we propose leveraging the appearance-invariant property by utilizing generative rendering to synthesize images with realistic textures and rich lighting based on the geometry of the object. Concretely, for each rendered image in the dataset, we utilize the corresponding normal map and depth map as control signals to synthesize realistic renderings with ControlNet (Zhang et al., 2023b). To ensure that the augmented images do not deviate significantly from the originals, we inject the original image using an IP-adapter (Ye et al., 2023). We then use IC-light (Zhang et al., 2024b) to generate renderings under various lighting conditions and directions. Experiments show that generative rendering augmentation is highly beneficial for helping diffusion models understand lighting effects and generalize to real-world images (see Fig. 7 for the corresponding ablation study).

Image-to-3D diffusion UNet. We adopt a UNet similar to Stable Diffusion (Rombach et al., 2022) as the image-to-shape diffusion network. Following Rodin (Wang et al., 2023b), we concatenate triplanes along the height dimension as input to the UNet to avoid spatial mismatches. Interactions between different planes are handled via self-attention layers. To incorporate image information during diffusion, we encode the input image using DINOv2 (Oquab et al., 2024) and inject the extracted features into the denoising process through cross-attention. Following SD3 (Esser et al., 2024), we adopt rectified flow Liu et al. (2022) with lognorm timestep sampling as the training schedule. For more details on the training and inference of our diffusion UNet, please refer to appendix A.

3.3 Texture Generation

3.3.1 Geometry-conditioned multi-view generation with reference attention

Previous image-guided texturing pipelines (Richardson et al., 2023; Zeng et al., 2023; Perla et al., 2024) adopt IP-adapter (Ye et al., 2023) or personalization techniques (Ruiz et al., 2022; Gal et al., 2022) to inject the image to pre-trained diffusion models. These methods are unable to generate textures consistent with the original image and are highly prone to the multi-face problem. To generate a Janus-free, image-consistent texture, we propose a geometry-conditioned ControlNet with reference attention to produce multi-view shaded images that align with the input in both appearance and lighting. Unlike the IP-adaptor, which maps images to prompts, reference attention (Zhang, 2023) integrates the keys and values from self-attention layers corresponding to the reference image into the denoising process, thus enhancing the consistency between the generated and original images. Our texturing model is based on Zero123++ (Shi et al., 2023a), which inherently uses scaled reference attention for generating multi-view images. To add geometry control, we trained a ControlNet (Zhang et al., 2023b) on top of the base model, enabling it to generate corresponding multi-view images from multi-view normal and depth maps. Specifically, our model mirrors the original ControlNet architecture but takes a six-channel image (3 for normal and 3 for depth) as input, i.e.

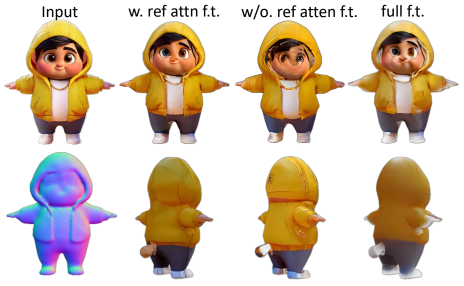

Figure 3: The effectiveness of the proposed reference attention fine-tuning.

$$\mathbf{I}_i^{MV} = f_\theta(\mathbf{I}_{i-1}^{MV}, \mathbf{I}^{\text{front}}, i, h_\phi(\mathbf{I}^{\text{front}}, i | N^{MV}, D^{MV})), \tag{5}$$

where $\mathbf{I}_i^{MV}$ represents the multi-view images at denoising step i, $\mathbf{I}^{\text{front}}$, N^{MV} and D^{MV} refer to the input image, multi-view normal map and normalized depth map. However, applying the geometry-conditioned ControlNet directly to generated shapes yields unsatisfying results, especially when the input image and the multi-view normal and depth maps are not geometrically consistent, as shown in the middle part of Fig. 3. We attribute this degradation to the gap between training and inference. To mitigate this issue, we propose fine-tuning the reference attention layers to ensure that the generated results focus more on the semantic information of the reference image, rather than being overly sensitive to minor discrepancies. Specifically, we randomly apply slight translations to the condition images and perform rotations and scaling on the mesh to perturb the rendered depth and normals, thereby simulating situations of imperfect geometric consistency. We freeze the entire model except for the projection matrices of the reference attention layers and fine-tune it on the augmented dataset. As shown in Fig. 3, this lightweight fine-tuning effectively compensates for performance loss due to imperfect matching without compromising the model's generative capability. In contrast, full fine-tuned models produce overly smooth textures, and the original model suffers significantly from inconsistency.

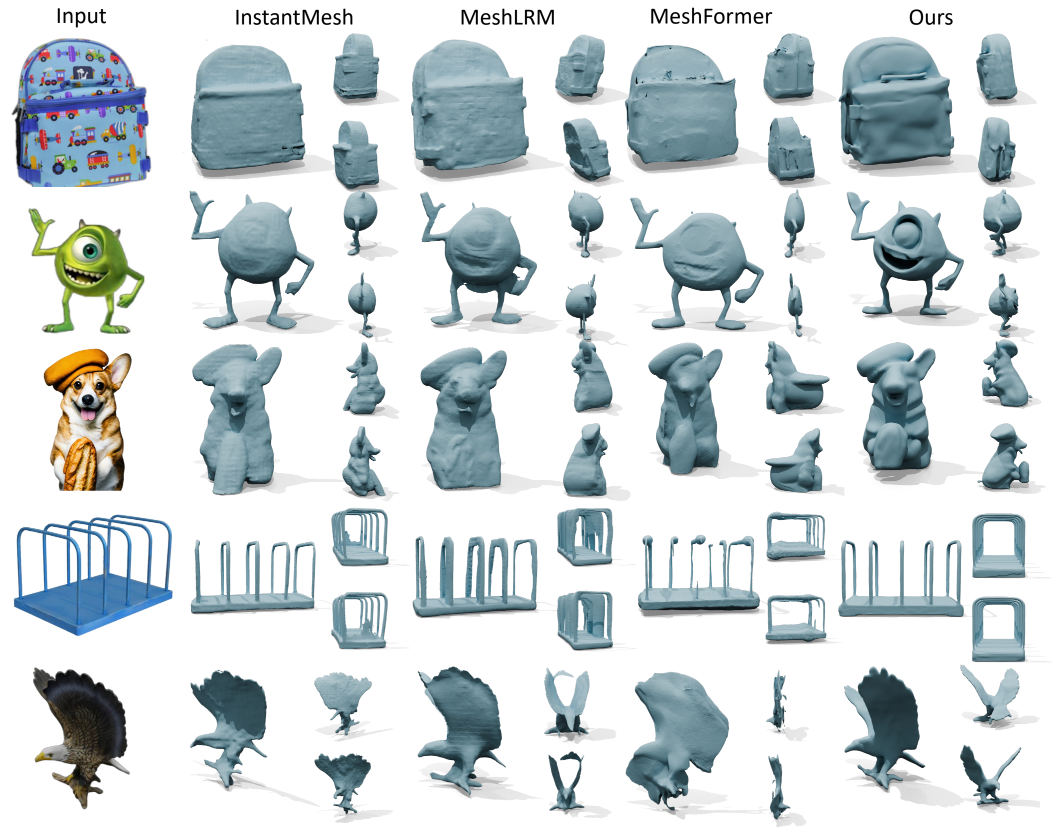

Figure 4: Qualitative comparison with state-of-the-art large reconstruction models, including InstantMesh (Xu et al., 2024b), MeshLRM (Wei et al., 2024) and MeshFormer (Liu et al., 2024).

3.3.2 Multi-view PBR decomposition

Previous methods that rely on pre-trained image diffusion models generate images with inherent shading effects, leading to lighting-baked-in textures. In order to generate relightable PBR textures, we propose a diffusion-based multi-view PBR decomposer, aiming to decompose the shaded multi-view image to corresponding intrinsic channels with multi-view information. Specifically, inspired by Zeng et al. (2024), our PBR decomposer employs an InstructPix2Pix (Brooks et al., 2023)-based architecture, concatenating the shaded image latent and the noisy latent along the channel dimension to output the desired PBR components, i.e.

$$\mathbf{I}_i^{MV}(y) = g_\phi(\mathbf{I}_{i-1}^{MV}(y); \mathbf{I}^{MV}, \mathbf{I}^{front}, i, \tau(y)), \tag{6}$$

where g_ϕ represents the denoising UNet, τ denotes the CLIP (Radford et al., 2021) text encoder, $y \in \{$"metallic","roughness","albedo"$\}$ denotes the component prompt for PBR texture, $\mathbf{I}_i^{MV}(y)$ refers to the denoised y component at timestep i. After generating multi-view PBR components, we use a view-weighted approach to fuse the multi-view textures in UV space, i.e. $\text{UV} = \sum_i \text{Softmax}_i(\text{BP}(\mathbf{I}^{(i)}), \text{BP}(w^{(i)}))$, where BP refers to back-projecting the rendered image to UV space, $\mathbf{I}^{(i)}$ denote the target image for the i-th view and $w^{(i)}$ represents the pixel-wise weight calculated as the cosine of the viewing angle to the point.

3.3.3 UV-space Texture Inpainting

For meshes with complicated topology, the generated views are not adequate to cover the entire surface of the mesh. We propose a UV-space texture inpainter to fill the invisible part of the multi-views. Specifically, due to the significant gap between casual images and texture maps, we first train a LoRA on the texture maps in Objaverse with “A UV space [y] texture map of [*]” as textual prompt, where the [*] represents the original caption of the corresponding 3D object generated using Cap3D (Luo et al., 2023) and [y] represent the PBR component prompt. Subsequently, we merge LoRA into the original UNet and train an inpainting ControlNet on top of it. To let the

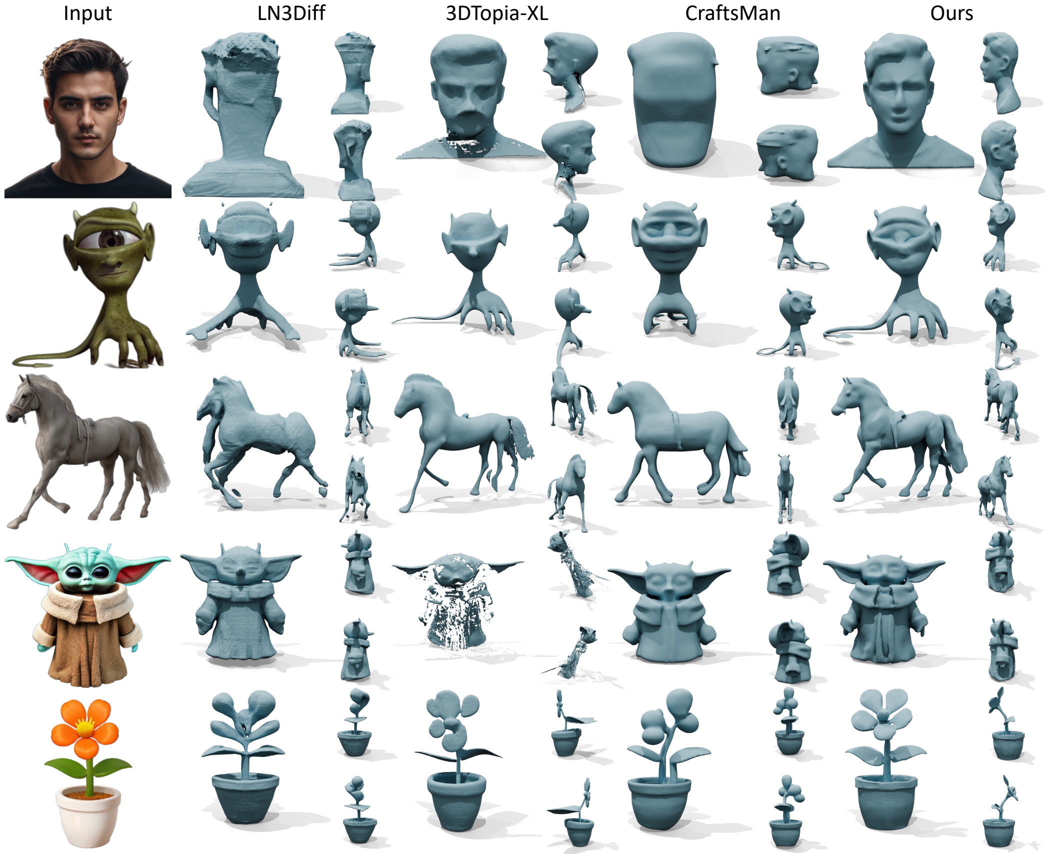

Figure 5: Qualitative comparison with state-of-the-art native 3D diffusion models, including CraftsMan (Li et al., 2024b), LN3Diff (Lan et al., 2024), and 3DTopia-XL (Chen et al., 2024a).

2D image perceive information from the original mesh, our control signal includes not only the masked image but also the normal and position maps in UV space. For the mask setup during training, we simulate the inference process by back-projecting the visible mask from the fixed views of our multi-view diffusion into UV space to obtain the invisible mask. To enhance robustness, we randomly erode the visible masks from multi-views. We present more details about the texturing pipeline in appendix A.3.

4 Experiments

4.1 Mesh Generation

In our experiments, we compare our method with state-of-the-art image-to-3D methods from the following two categories.

Large reconstruction models. (Hong et al., 2023; Li et al., 2023a;b; Wang et al., 2024; Tang et al., 2024a) exploit a neural network to map sparse views into 3D representations. We compare the proposed methods with recent state-of-the-art large reconstruction models, including InstantMesh (Xu et al., 2024b), MeshLRM (Wei et al., 2024) and MeshFormer (Liu et al., 2024). To be clear, the results of MeshLRM and MeshFormer are obtained through their official demo, since their source code is not publicly available.

Native 3D generation models. (Liu et al., 2023a; Zhang et al., 2023a; 2024a) compress discrete 3D meshes into a continuous and compact latent space using a 3D auto-encoder, followed by a diffusion model trained on this latent space to achieve 3D generation. We compared our method with recent

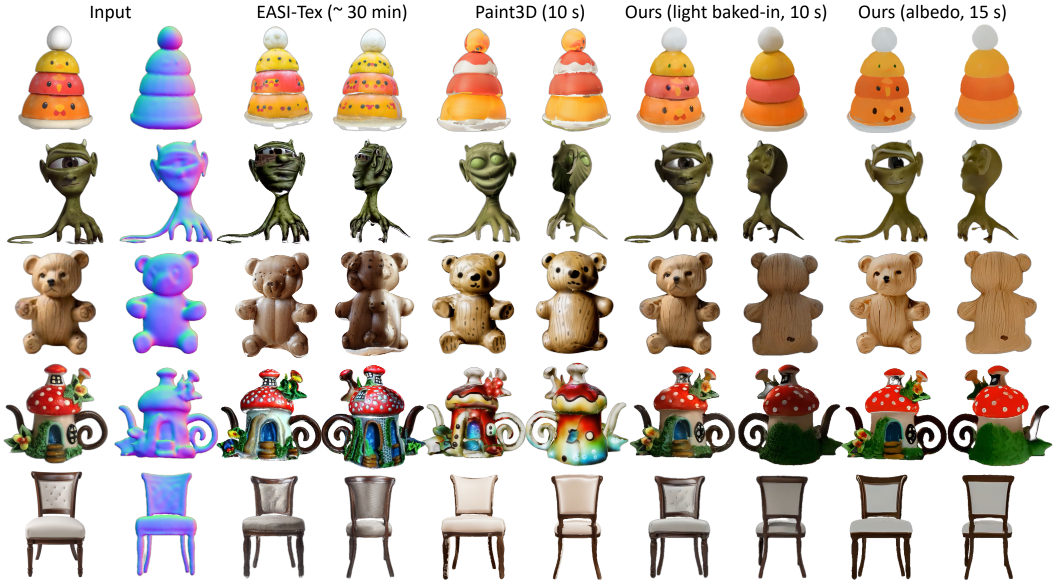

Figure 6: Qualitative comparison with image-guided mesh texturing pipelines, including EASI-Tex (Perla et al., 2024) and Paint3D (Zeng et al., 2023).

state-of-the-art open-source models, including LN3Diff (Lan et al., 2024), 3DTopia-XL (Chen et al., 2024a), and CraftsMan (Li et al., 2024b).

We present a qualitative comparison between our method and large reconstruction models in Fig. 4. Our approach significantly outperforms others in generating high-quality geometry. Specifically, our method excels at producing complex structures from images, such as backpack handles, the mouth of the alien, and the file sorter, where large reconstruction models struggle. Additionally, large reconstruction models often suffer from poor multi-view generation outcomes, resulting in lower quality when reconstructing details that require multi-view consistency, such as the gap between an eagle's legs. In Fig. 5, we provide a qualitative comparison with other 3D native generation methods, where our approach significantly outperforms the others. We observe that previous methods often produce symmetrical and overly simplistic geometric structures, resulting in noticeable misalignment with the input images. Our method, leveraging the proposed two data augmentations, effectively addresses this issue. This image-shape alignment enhances the controllability of shape generation and simplifies subsequent texturing. For 3DTopia-XL, its explicit representation results in a lower compression rate compared to the point-to-shape auto-encoder. Consequently, despite higher training costs, MeshGen still outperforms 3DTopia-XL by a large margin.

4.2 Texture Generation

As there exist no algorithms specifically designed for image-consistent texturing, we compared our method with state-of-the-art image-guided approaches, including EASI-Tex (Perla et al., 2024) and Paint3D (Zeng et al., 2023). Fig. 6 demonstrates the texturing results on meshes generated by our image-to-shape diffusion model. Our method significantly outperforms previous approaches in quality and texture-image consistency, even when the shape and image do not perfectly align. In contrast, despite using image inversion which requires additional per-input optimization, EASI-TEX still struggles to maintain consistency with the original image and takes dozens of times longer than our approach. Paint3D, which uses simple back projection and inpainting, exhibits noticeable seams in the generated textures and is prone to the Janus problem. We further showcase the PBR materials generated by our method in Fig. 13, highlighting its remarkable capability in handling objects with complicated appearances under different lighting conditions.

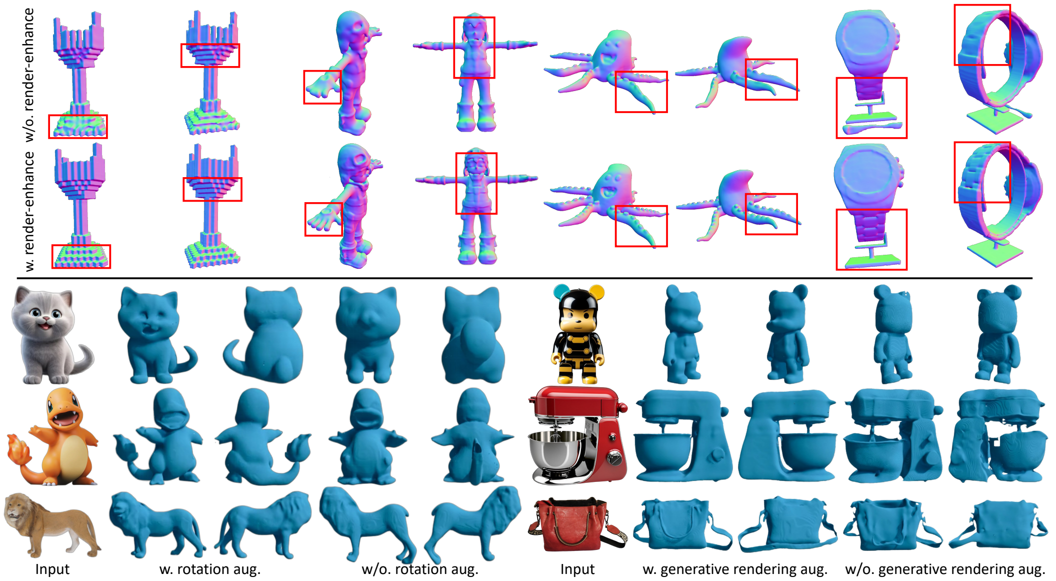

Figure 7: Ablation study results on render-enhanced auto-encoder, geometric alignment augmentation, and generative rendering augmentation.

4.3 Ablations

Render-enhanced auto-encoder. To evaluate the significance of incorporating render loss in the point-to-shape auto-encoder, we trained a variant without this component and compared it to our render-enhanced version, as illustrated in the upper part of Fig. 7. The render-enhanced auto-encoder exhibits markedly superior performance, particularly in capturing high-frequency details such as the suckers on tentacles and the gaps in watch bands.

Geomtric alignment augmentation. To validate the impact of geometric alignment augmentation, we trained a smaller diffusion UNet consisting of 4 layers without this augmentation for 300 epochs as an ablation study. The comparison is presented in the lower left part of Fig. 7. Evidently, the diffusion model trained without geometric alignment augmentation tends to generate symmetric objects, whereas our model produces shapes that align well with the input images, significantly enhancing the model's controllability.

Generative rendering augmentation. To assess the impact of generative rendering augmentation on image-to-shape diffusion training, we trained a smaller model without this augmentation, as depicted in the lower right part of Fig. 7. The model trained without generative rendering augmentation exhibits poor performance in handling lighting effects in images and struggles to infer the geometric structure based on lighting cues, such as determining the shape of the doll's head and the shape of the blender. These findings suggest that generative rendering augmentation significantly enhances the model's ability to understand lighting effects and interpret real-world images.

More ablations regarding to auto-encoder, image-to-shape diffusion model, and texture generation model are presented in appendix B.1.

5 Conclusion

In this paper, we propose MeshGen, a novel pipeline for generating delicate PBR textured mesh given a single image. MeshGen encodes 3D meshes to compact latent space with a render-enhanced auto-encoder. Based on our in-depth analysis of point-to-shape auto-encoder and image-to-shape diffusion, we propose to train the diffusion model with geometric alignment and generative rendering augmentation to address the issues of image-shape misalignment and poor generalization ability. Besides, to generate PBR texture consistent with the image, we establish a reference attention-based multi-view generator followed by a PBR decomposer to obtain PBR components and a UV-space inpainter to fill the invisible part. Extensive experiments have demonstrated the effectiveness of our method. We hope our work will aid in a deeper understanding of native 3D diffusion and provide support for future related research.

REFERENCES

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.

Raphael Bensadoun, Yanir Kleiman, Idan Azuri, Omri Harosh, Andrea Vedaldi, Natalia Neverova, and Oran Gafni. Meta 3d texturegen: Fast and consistent texture generation for 3d objects. *arXiv preprint arXiv: 2407.02430*, 2024.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023.

Tianshi Cao, Karsten Kreis, Sanja Fidler, Nicholas Sharp, and Kangxue Yin. Texfusion: Synthesizing 3d textures with text-guided image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4169–4181, 2023.

Eric Chan, Connor Z. Lin, Matthew Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J. Guibas, Jonathan Tremblay, S. Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3d generative adversarial networks. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16102–16112, 2021. URL `https://api.semanticscholar.org/CorpusID:245144673`.

Dave Zhenyu Chen, Yawar Siddiqui, Hsin-Ying Lee, Sergey Tulyakov, and Matthias Nießner. Text2tex: Text-driven texture synthesis via diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 18558–18568, October 2023a.

Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2023b.

Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. *CoRR*, abs/2311.14521, 2023c. doi: 10.48550/ARXIV.2311.14521. URL `https://doi.org/10.48550/arXiv.2311.14521`.

Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, Liang Pan, Dahua Lin, and Ziwei Liu. 3dtopia-xl: High-quality 3d pbr asset generation via primitive diffusion. *arXiv preprint arXiv:2409.12957*, 2024a.

Zhiqin Chen, Kangxue Yin, and Sanja Fidler. Auv-net: Learning aligned uv maps for texture transfer and synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1465–1474, 2022.

Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *CVPR*, 2024b. URL `https://arxiv.org/abs/2309.16585`.

Zilong Chen, Yikai Wang, Feng Wang, Zhengyi Wang, and Huaping Liu. V3d: Video diffusion models are effective 3d generators. *arXiv preprint arXiv:2403.06738*, 2024c.

An-Chieh Cheng, Xueting Li, Sifei Liu, and Xiaolong Wang. Tuvf: Learning generalizable texture uv radiance fields. *arXiv preprint arXiv:2305.03040*, 2023.

Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21126–21136, 2022.

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. *arXiv preprint arXiv:2212.08051*, 2022.

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*, pp. 13142–13153, 2023.

Kangle Deng, Timothy Omernick, Alexander Weiss, Deva Ramanan, Jun-Yan Zhu, Tinghui Zhou, and Maneesh Agrawala. Flashtex: Fast relightable mesh texturing with lightcontrolnet. *arXiv preprint arXiv: 2402.13251*, 2024.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. *arXiv preprint arXiv: 2403.03206*, 2024.

Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H. Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion, 2022. URL https://arxiv.org/abs/2208.01618.

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.

Fangzhou Hong, Jiaxiang Tang, Ziang Cao, Min Shi, Tong Wu, Zhaoxi Chen, Tengfei Wang, Liang Pan, Dahua Lin, and Ziwei Liu. 3dtopia: Large text-to-3d generation model with hybrid diffusion priors. *arXiv preprint arXiv:2403.02234*, 2024.

Yicong Hong, Kai Zhang, Jiuxiang 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. *CoRR*, abs/2311.04400, 2023. doi: 10.48550/ARXIV.2311.04400. URL https://doi.org/10.48550/arXiv.2311.04400.

Ajay Jain, Ben Mildenhall, Jonathan T. Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 857–866. IEEE, 2022. doi: 10.1109/CVPR52688.2022.00094. URL https://doi.org/10.1109/CVPR52688.2022.00094.

DaDong Jiang, Xianghui Yang, Zibo Zhao, Sheng Zhang, Jiaao Yu, Zeqiang Lai, Shaoxiong Yang, Chunchao Guo, Xiaobo Zhou, and Zhihui Ke. Flexitex: Enhancing texture generation with visual guidance. *arXiv preprint arXiv: 2409.12431*, 2024.

Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023.

Samuli Laine, Janne Hellsten, Tero Karras, Yeongho Seol, Jaakko Lehtinen, and Timo Aila. Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics*, 39(6), 2020.

Yushi Lan, Fangzhou Hong, Shuai Yang, Shangchen Zhou, Xuyi Meng, Bo Dai, Xingang Pan, and Chen Change Loy. Ln3diff: Scalable latent neural fields diffusion for speedy 3d generation. In *ECCV*, 2024.

Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *CoRR*, abs/2311.06214, 2023a. doi: 10.48550/ARXIV.2311.06214. URL https://doi.org/10.48550/arXiv.2311.06214.

Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *CoRR*, abs/2311.06214, 2023b. doi: 10.48550/ARXIV.2311.06214. URL https://doi.org/10.48550/arXiv.2311.06214.

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*, 2024a.

Weiyu Li, Rui Chen, Xuelin Chen, and Ping Tan. Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. *arxiv:2310.02596*, 2023c.

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*, 2024b.

Yuchen Li, Ujjwal Upadhyay, Habib Slim, Ahmed Abdelreheem, Arpit Prajapati, Suhail Pothigara, Peter Wonka, and Mohamed Elhoseiny. 3d compat: Composition of materials on parts of 3d things. In *European Conference on Computer Vision*, pp. 110–127. Springer, 2022.

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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. *arXiv preprint arXiv:2311.07885*, 2023a.

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

Minghua Liu, Chong Zeng, Xinyue Wei, Ruoxi Shi, Linghao Chen, Chao Xu, Mengqi Zhang, Zhaoning Wang, Xiaoshuai Zhang, Isabella Liu, Hongzhi Wu, and Hao Su. Meshformer: High-quality mesh generation with 3d-guided reconstruction model. *arXiv preprint arXiv:2408.10198*, 2024.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

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*, 2023c.

Yuxin Liu, Minshan Xie, Hanyuan Liu, and Tien-Tsin Wong. Text-guided texturing by synchronized multi-view diffusion. *arXiv preprint arXiv:2311.12891*, 2023d.

Zexiang Liu, Yangguang Li, Youtian Lin, Xin Yu, Sida Peng, Yan-Pei Cao, Xiaojuan Qi, Xiaoshui Huang, Ding Liang, and Wanli Ouyang. Unidream: Unifying diffusion priors for relightable text-to-3d generation, 2023e.

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.

Yuanxun Lu, Jingyang Zhang, Shiwei Li, Tian Fang, David McKinnon, Yanghai Tsin, Long Quan, Xun Cao, and Yao Yao. Direct2.5: Diverse text-to-3d generation via multi-view 2.5d diffusion. *ArXiv*, abs/2311.15980, 2023. URL `https://api.semanticscholar.org/CorpusID:265456029`.

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

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.

Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016. doi: 10.23915/distill.00003. URL `http://distill.pub/2016/deconv-checkerboard`.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a68SUt6zFt.

Sai Raj Kishore Perla, Yizhi Wang, Ali Mahdavi-Amiri, and Hao Zhang. Easi-tex: Edge-aware mesh texturing from single image. *ACM Transactions on Graphics (Proceedings of SIGGRAPH)*, 43(4), 2024. doi: 10.1145/3658222. URL https://github.com/sairajk/easi-tex.

Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/pdf?id=FjNys5c7VyY.

Lingteng Qiu, Guanying Chen, Xiaodong Gu, Qi Zuo, Mutian Xu, Yushuang Wu, Weihao Yuan, Zilong Dong, Liefeng Bo, and Xiaoguang Han. Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9914–9925, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL http://proceedings.mlr.press/v139/radford21a.html.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pp. 5301–5310. PMLR, 2019.

Xuanchi Ren, Jiahui Huang, Xiaohui Zeng, Ken Museth, Sanja Fidler, and Francis Williams. Xcube: Large-scale 3d generative modeling using sparse voxel hierarchies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Elad Richardson, Gal Metzer, Yuval Alaluf, Raja Giryes, and Daniel Cohen-Or. Texture: Text-guided texturing of 3d shapes. In Erik Brunvand, Alla Sheffer, and Michael Wimmer (eds.), *ACM SIGGRAPH 2023 Conference Proceedings, SIGGRAPH 2023, Los Angeles, CA, USA, August 6-10, 2023*, pp. 54:1–54:11. ACM, 2023. doi: 10.1145/3588432.3591503. URL https://doi.org/10.1145/3588432.3591503.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pp. 10674–10685. IEEE, 2022. doi: 10.1109/CVPR52688.2022.01042. URL https://doi.org/10.1109/CVPR52688.2022.01042.

Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. 2022.

Aditya Sanghi, Hang Chu, Joseph G Lambourne, Ye Wang, Chin-Yi Cheng, and Marco Fumero. Clip-forge: Towards zero-shot text-to-shape generation. *arXiv preprint arXiv:2110.02624*, 2021.

Junyoung Seo, Wooseok Jang, Min-Seop Kwak, Jaehoon Ko, Hyeonsu 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.

Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4), jul 2023. ISSN 0730-0301. doi: 10.1145/3592430. URL https://doi.org/10.1145/3592430.

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

Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv:2308.16512*, 2023b.

Yawar Siddiqui, Justus Thies, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Texturify: Generating textures on 3d shape surfaces. In *European Conference on Computer Vision*, pp. 72–88. Springer, 2022.

Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. *arXiv preprint arXiv:2310.16818*, 2023.

Wenqiang Sun, Zhengyi Wang, Shuo Chen, Yikai Wang, Zilong Chen, Jun Zhu, and Jun Zhang. Freeplane: Unlocking free lunch in triplane-based sparse-view reconstruction models. *arXiv preprint arXiv:2406.00750*, 2024.

Stanislaw Szymanowicz, Christian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *arXiv*, 2023.

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

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*, 2024a.

Shitao Tang, Jiacheng Chen, Dilin Wang, Chengzhou Tang, Fuyang Zhang, Yuchen Fan, Vikas Chandra, Yasutaka Furukawa, and Rakesh Ranjan. Mvdiffusion++: A dense high-resolution multi-view diffusion model for single or sparse-view 3d object reconstruction. *arXiv preprint arXiv:2402.12712*, 2024b.

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.

Peihao Wang, Dejia Xu, Zhiwen Fan, Dilin Wang, Sreyas Mohan, Forrest Iandola, Rakesh Ranjan, Yilei Li, Qiang Liu, Zhangyang Wang, and Vikas Chandra. Taming mode collapse in score distillation for text-to-3d generation. *arXiv preprint: 2401.00909*, 2023a.

Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023.

Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, and Baining Guo. RODIN: A generative model for sculpting 3d digital avatars using diffusion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pp. 4563–4573. IEEE, 2023b. doi: 10.1109/CVPR52729.2023.00443. URL `https://doi.org/10.1109/CVPR52729.2023.00443`.

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*, 2023c.

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.

Xinyue Wei, Fanbo Xiang, Sai Bi, Anpei Chen, Kalyan Sunkavalli, Zexiang Xu, and Hao Su. Neumanifold: Neural watertight manifold reconstruction with efficient and high-quality rendering support. *arXiv preprint arXiv:2305.17134*, 2023.

Xinyue Wei, Kai Zhang, Sai Bi, Hao Tan, Fujun Luan, Valentin Deschaintre, Kalyan Sunkavalli, Hao Su, and Zexiang Xu. Meshlrm: Large reconstruction model for high-quality mesh. *arXiv preprint arXiv: 2404.12385*, 2024.

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, 2024a.

Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Jingxi Xu, Philip Torr, Xun Cao, and Yao Yao. Direct3d: Scalable image-to-3d generation via 3d latent diffusion transformer. *arXiv preprint arXiv: 2405.14832*, 2024b.

Dejia Xu, Ye Yuan, Morteza Mardani, Sifei Liu, Jiaming Song, Zhangyang Wang, and Arash Vahdat. Agg: Amortized generative 3d gaussians for single image to 3d. *arXiv preprint arXiv: 2401.04099*, 2024a.

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*, 2024b.

Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Peng Wang, Jiahao Li, Zifan Shi, Kalyan Sunkavalli, Gordon Wetzstein, Zexiang Xu, and Kai Zhang. Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model. *ArXiv*, abs/2311.09217, 2023. URL `https://api.semanticscholar.org/CorpusID:265213192`.

Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. 2023.

Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. *arXiv preprint arXiv:2112.05131*, 2021a.

Rui Yu, Yue Dong, Pieter Peers, and Xin Tong. Learning texture generators for 3d shape collections from internet photo sets. In *British Machine Vision Conference*, 2021b.

Xin Yu, Peng Dai, Wenbo Li, Lan Ma, Zhengzhe Liu, and Xiaojuan Qi. Texture generation on 3d meshes with point-uv diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4206–4216, 2023.

Xianfang Zeng, Xin Chen, Zhongqi Qi, Wen Liu, Zibo Zhao, Zhibin Wang, BIN FU, Yong Liu, and Gang Yu. Paint3d: Paint anything 3d with lighting-less texture diffusion models, 2023.

Zheng Zeng, Valentin Deschaintre, Iliyan Georgiev, Yannick Hold-Geoffroy, Yiwei Hu, Fujun Luan, Ling-Qi Yan, and Miloš Hašan. Rgb¡-¿x: Image decomposition and synthesis using material- and lighting-aware diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, SIGGRAPH ’24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400705250. doi: 10.1145/3641519.3657445. URL `https://doi.org/10.1145/3641519.3657445`.

Biao Zhang, Jiapeng Tang, Matthias Nießner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Trans. Graph.*, 42(4), jul 2023a. ISSN 0730-0301. doi: 10.1145/3592442. URL `https://doi.org/10.1145/3592442`.

Longwen Zhang, Ziyu Wang, Qixuan Zhang, Qiwei Qiu, Anqi Pang, Haoran Jiang, Wei Yang, Lan Xu, and Jingyi Yu. Clay: A controllable large-scale generative model for creating high-quality 3d assets. *ACM Transactions on Graphics (TOG)*, 43(4):1–20, 2024a.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023b.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Ic-light github page, 2024b.

Lyumin Zhang. Reference-only control. `https://github.com/Mikubill/sd-webui-controlnet/discussions/1236`, 2023.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 586–595. Computer Vision Foundation / IEEE Computer Society, 2018. doi: 10.1109/CVPR.2018.00068. URL `http://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html`.

Xin-Yang Zheng, Yang Liu, Peng-Shuai Wang, and Xin Tong. Sdf-stylegan: Implicit sdf-based stylegan for 3d shape generation. In *Comput. Graph. Forum (SGP)*, 2022.

Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. *arXiv preprint arXiv:2312.09147*, 2023.

A Implementation details

A.1 Auto-encoder

We present our hyper-parameter setting in training auto-encoder in Tab. A.1.

Table 1: Concrete hyper-parameter setting of our render-enhanced auto-encoder.

Symbol	Meaning	Value
N_P	Number of points sampled from a mesh	65536
N_z	Number of learnable queries	3072
n	Number of self-attention layers	10
d_z	Dimension of the latent space	16
N_s	Number of samples for calculating ray-based regularization loss	128
λ_{KL}	Loss weight for KL loss	10^{-6}
λ_{TV}	Loss weight for TV loss	5×10^{-3}
λ_{MSE}	Loss weight for normal MSE loss	1.0
λ_{LPIPS}	Loss weight for normal LPIPS loss	2.0
λ_{reg}	Loss weight for ray-based regularization loss	0.5

We first train our auto-encoder for 150 epochs in the coarse stage with a batch size of 192. The model obtained after the coarse stage can reconstruct the rough shape of the original mesh but lacks details. We then train the auto-encoder for another 50 epochs with the proposed render loss and a batch size of 16.

A.2 Image-to-shape diffusion model

A.2.1 Data augmentation

In generative rendering data augmentation, to enhance the similarity between the generated images and the original image, in addition to using the normal depth ControlNet and IP-adapter, we set the initial noise to the latent of the original image with maximum noise added. For relighting diffusion, we used IC-light (Zhang et al., 2024b). Specifically, during data augmentation, we randomly select one lighting direction from the pre-defined light initial latent in IC-light (i.e., uniformly select from left, right, top, and bottom), and choose one lighting condition from a set of predefined light prompts.

A.2.2 Image-to-shape diffusion model

Our diffusion UNet takes in the noised triplane latent and exploits 8 ResNet blocks with spatial self-attention as the encoder and a symmetric architecture as the decoder. We exploit DINOv2-G (Oquab et al., 2024) to encode the input image and inject the extracted feature to the diffusion UNet using cross-attention. For the diffusion schedule, we follow SD3 to use the simple yet effective rectified flow Liu et al. (2022) with timesteps sampled from a standard logit-normal distribution. We train the image-to-shape diffusion model with our proposed augmentations on a filtered subset of GObjaverse (Qiu et al., 2024), which consists of about 120k high-quality multiview-mesh pairs. To handle input images with different elevations, since the meshes in Objaverse are aligned in the gravity axis, we force the diffusion to generate meshes with absolute elevation equal to zero. We experimentally found that this conditioning method works better than generating meshes with a rotation in elevation, as suggested in Chen et al. (2024c). We train the diffusion UNet of 16 NVIDIA A800 GPUs using bf16 precision with an effective batch size equal to 1536. The whole training lasts for about 18 days.

A.3 PBR texture generation

Data preparation. To train the geometry-conditioned ControlNet and the multi-view PBR decomposer, we rendered multi-view images and corresponding multi-view normals, depth, albedo, roughness, and metallic maps of a subset of Objaverse containing PBR materials using Blender.

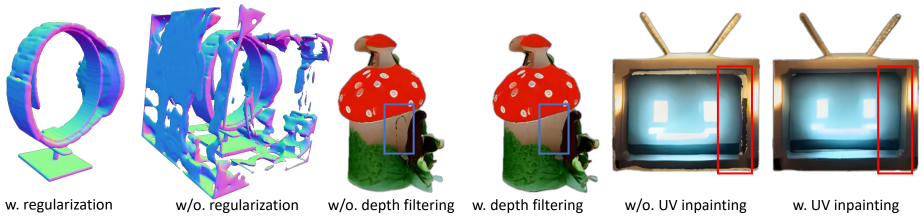

Figure 8: Ablations on ray-based regularization, depth filtering, and UV space inpainting.

This constitutes a dataset comprising 35k multi-view images. For UV space inpainting, we calculate the multi-view visible masks and back-project them into UV space to determine the invisible part of the texture map and render the UV space position and normal map. To further enhance robustness, we randomly erode the visible mask in pixel and UV space.

Geometry-conditioned ControlNet training. To ensure the model perceives precise depth and positional information, we did not transform depth to normalized disparity as done in the original depth ControlNet (Zhang et al., 2023b); instead, we performed a unified multi-view normalization based on camera distance and object bounding box. Specifically, the depth map is processed as $D_{\text{normalized}} = \frac{D - \text{bias}}{\text{scale}}$, where bias equals to camera distance minus the length of the diagonal of the bounding box (i.e. the minimal possible depth value) and the scale equals to the length of the diagonal of the bounding box.

Multi-view target back-projection. As detailed in the main text, the obtained multi-view PBR components are merged in UV space using back-projection with softmax. We apply a softmax operation with a temperature of 0.1 to ensure consistent textures. However, images generated by ControlNet sometimes extend beyond object boundaries, causing some pixels to be back-projected onto surfaces behind them, leading to artifacts. To address this, we propose a simple depth filtering technique. For each view, we identify locations in the depth map where sudden changes occur and exclude these pixels during back-projection. Our experiments demonstrate that this approach effectively reduces artifacts, and the color values of the corresponding surface points can be supplemented by other views, as shown in the middle of Fig. 8.

UV space inpainting. Our UV space inpainter is a multi-channel ControlNet trained on top of the LoRA fine-tuned diffusion model. The input to our inpainting model is a 9-channel image: the first three channels represent the normal map in UV space, the middle three channels represent the position map, and the last three channels contain the masked texture map, with pixel values set to -1 in regions that requires inpainting. During inference, we follow ControlNet inpainting (Zhang et al., 2023b), applying masking in the latent space to maintain consistency in areas that do not require inpainting.

B More experiments

B.1 More ablations

The effectiveness of ray-based regularization. We show in the left part of Fig. 8 an example obtained using an auto-encoder trained without ray-based regularization. Without ray-based regularization, the training of the auto-encoder quickly became unstable, resulting in severe floaters in the reconstructed mesh.

Quatitative ablation study on render-enhanced auto-encoder. To better assess the importance of incorporating render loss in our render-enhanced auto-encoder, we propose several variants and demonstrate the corresponding accuracy and volumeIoU on a validation set of Objaverse consisting of 2048 objects in Tab. B.1. Here, “base” represents the case with only BCE loss, while “w/. 3D GAN loss” represents incorporating the 3D patch-based GAN loss proposed in Zheng et al. (2022).

As shown in Tab. B.1, removing either the MSE loss or the LPIPS loss leads to a certain performance drop. Moreover, compared to the 3D patch-based GAN loss, the proposed render-based perceptual loss is more beneficial for auto-encoder training.

UV-space texture inpainting. In the right part of Fig. 8, we compare the mesh obtained without UV inpainting. The figure clearly shows that without UV inpainting, colors may be missing from re-

Table 2: Quantitative ablation study on the proposed render-enhanced auto-encoder.

Setting	Accuracy↑	VolumeIoU↑
Ours	**96.987**	**91.045**
w/o. $\mathcal{L}_{\text{normal}}^{\text{MSE}}$	95.972	89.977
w/o. $\mathcal{L}_{\text{normal}}^{\text{LPIPS}}$	96.021	90.044
w/. 3D patch GAN loss	96.224	90.149
base	94.745	87.164

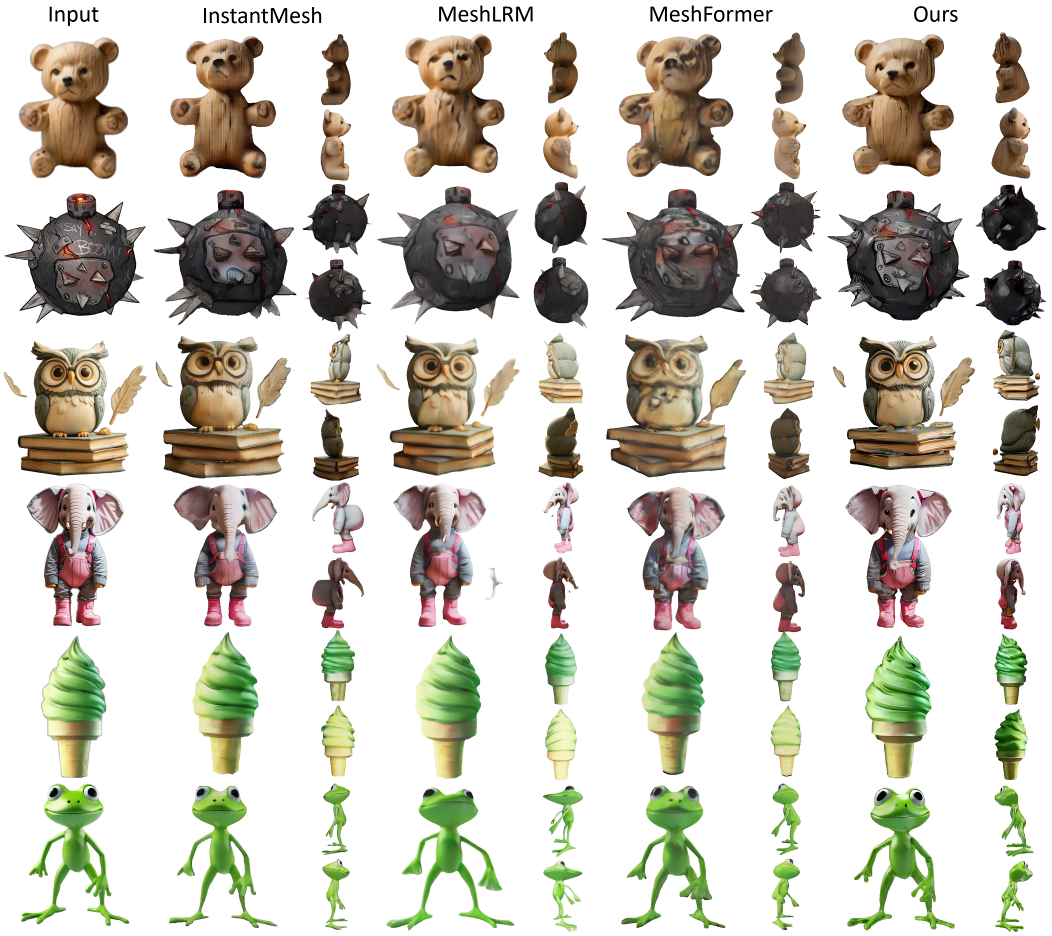

Figure 9: Qualitative comparison on textured meshes with state-of-the-art large reconstruction models, including InstantMesh (Xu et al., 2024b), MeshLRM (Wei et al., 2024) and MeshFormer (Liu et al., 2024).

gions not visible from the fixed viewpoints generated by multi-view diffusion. UV space inpainting effectively fills these regions with appropriate colors, enhancing both the visual quality and realism of the model.

B.2 More results

Compare with large reconstruction models with texture. To comprehensively compare our approach with large reconstruction models, we compare the final generated textured mesh in Fig. 9. It is evident from the figure that our method not only exceeds the previous best large reconstruction models in geometry but also produces clearer and more consistent textures.

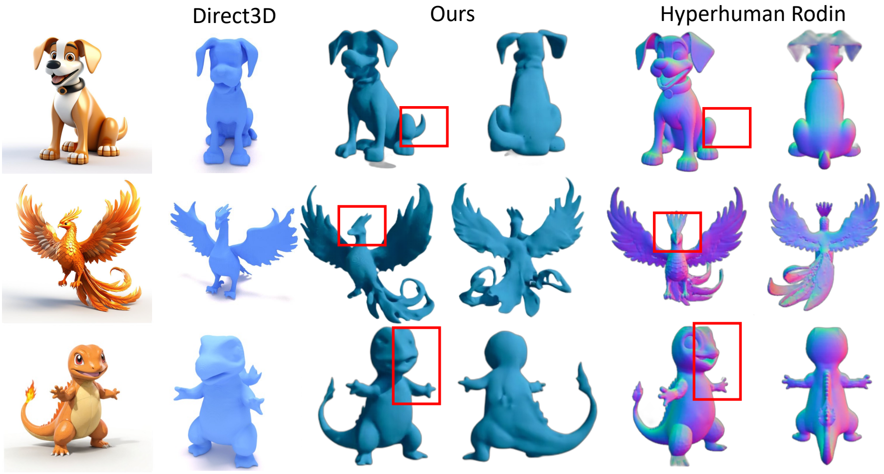

Figure 10: Comparison with non-open-source commercial products, including Direct3D and Hyperhuman Rodin.

Compare with commercial products. In Fig. 10, we compare our method with existing non-open-source commercial products. The results for Direct3D are sourced from their paper, while those for HyperHuman Rodin are generated on their official website without the "symmetric" tags. Although our method is currently limited by lacking high-quality data and computational resources, resulting in slightly lower mesh quality compared to commercial products, our proposed augmentation allows for better alignment with the images. We believe that with increased computational power and more high-quality data, our method can match the mesh quality of commercial products while preserving image-shape alignment.

Real-world images. To validate the performance of our method on real-world objects, we present a set of textured meshes generated from casual captures in Fig. 11. As shown in Fig. 11, our method is capable of generating reasonable shapes and consistent textures when processing real objects, demonstrating the generalization ability of our pipeline.

PBR decompositions. In Fig. 13, we present the intrinsic channels estimated using our proposed multi-view PBR decomposer. The results show that our PBR decomposer can accurately infer the PBR materials of objects by leveraging multi-view information and can still generate multi-view consistent results under complex lighting conditions.

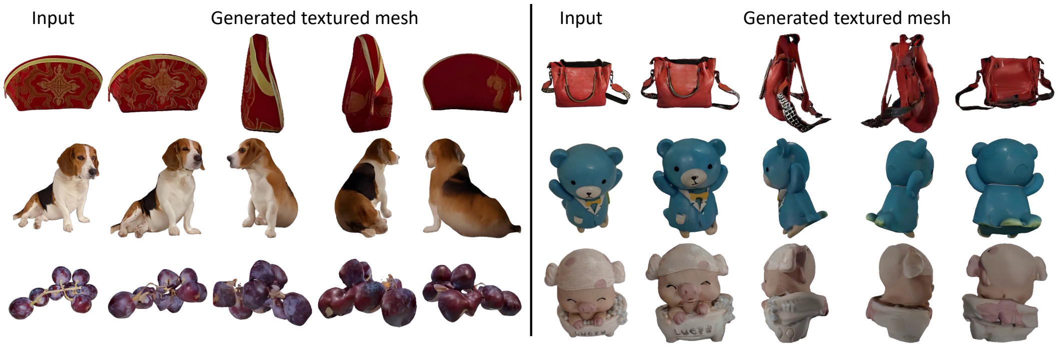

Figure 11: Performance of MeshGen on real-world captures.

C Limitations

Although our method has made some progress in native image-to-3D generation, there are still limitations in the following three areas.

1. Due to the limited resolution of multi-view diffusion generation and the constraints of the auto-encoder used, our texture model struggles to accurately reproduce high-frequency de-

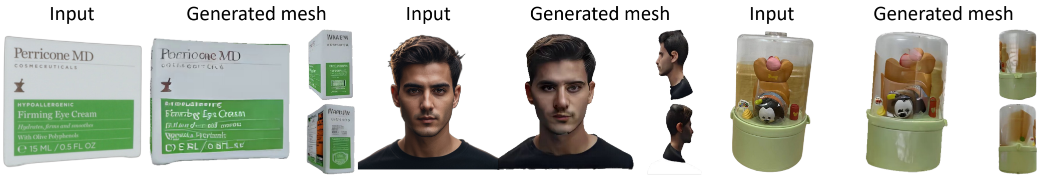

Figure 12: Some typical failure cases of MeshGen.

tails, such as the text on the box in the left part of Fig. 12. We believe that using more advanced network architectures could achieve higher-resolution multi-view generation.

2. Our texture model finds it challenging to accurately capture textures and lighting effects from input images when dealing with objects with complex high-frequency information and lighting conditions, as shown by the face in the center of Fig. 12.
3. Our geometry generation model currently cannot effectively handle transparent objects, as illustrated by the object on the right in Fig. 12.

Addressing these limitations will be the focus of our future research.

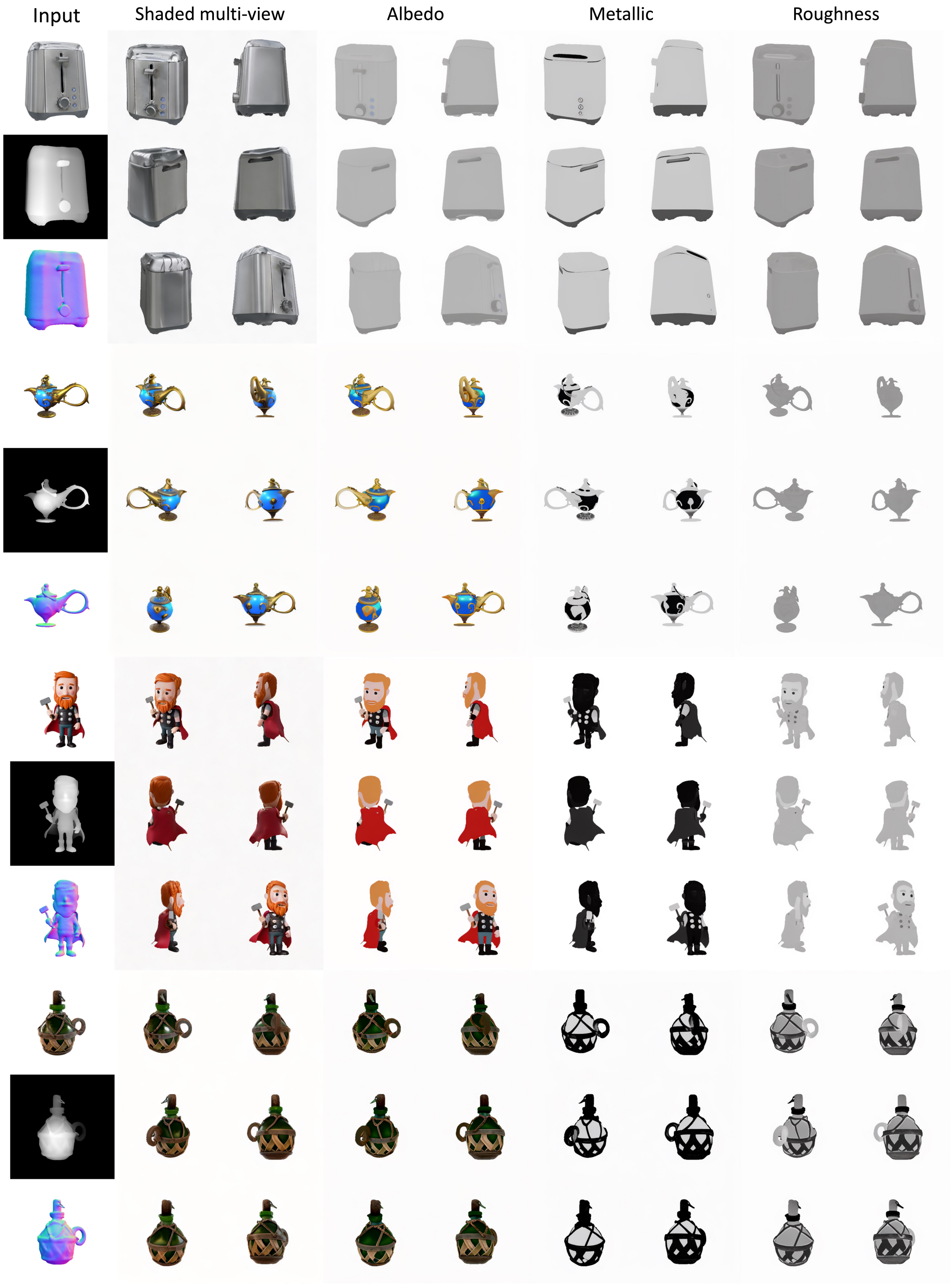

Figure 13: Intrinsic channels estimated using our multi-view PBR decomposer. The proposed PBR decomposer can handle images with complicated material under different lighting conditions.