

Kiss3DGen: Repurposing Image Diffusion Models for 3D Asset Generation

Jiantao Lin^{1*} Xin Yang^{1,2*} Meixi Chen^{1*} Yingjie Xu¹ Dongyu Yan¹
 Leyi Wu¹ Xinli Xu¹ Lie Xu³ Shunsi Zhang³ Ying-Cong Chen^{1,2†}
¹HKUST(GZ) ²HKUST ³Guangzhou Quwan Network Technology



Figure 1. **A 3D Harry Potter scene built with Kiss3DGen.** Our proposed framework, KISS3DGen, is a unified 3D generation framework that facilitates various 3D generation tasks, including text-to-3D, image-to-3D, 3D enhancement, editing and more. Specifically, most of the assets in the figure is generated from text (captioned with abbreviated **text prompts**) or image (marked by **dash lines**) conditions, while the main characters (*Hermione, Ron and Potter*) are created using a hybrid pipeline that combines image-to-3D and text-guided mesh editing. Please zoom in for details and refer to our main paper for a more introduction.

Abstract

Diffusion models have achieved great success in generating 2D images. However, the quality and generalizability of 3D content generation remain limited. State-of-the-art methods often require large-scale 3D assets for training, which are challenging to collect. In this work, we introduce **Kiss3DGen** (**K**eep **I**t **Simple and **S**traightforward in **3D G**eneration), an efficient framework for generating, editing, and enhancing 3D objects by repurposing a well-trained 2D image diffusion model for 3D generation. Specifically, we fine-tune a diffusion model to generate "3D Bundle Image", a tiled representation composed of multi-view images and their corresponding normal maps. The normal maps are then used to reconstruct a 3D mesh, and the multi-view images provide texture mapping, resulting in a complete**

3D model. This simple method effectively transforms the 3D generation problem into a 2D image generation task, maximizing the utilization of knowledge in pretrained diffusion models. Furthermore, we demonstrate that our Kiss3DGen model is compatible with various diffusion model techniques, enabling advanced features such as 3D editing, mesh and texture enhancement, etc. Through extensive experiments, we demonstrate the effectiveness of our approach, showcasing its ability to produce high-quality 3D models efficiently. Project page: <https://ltt-o.github.io/Kiss3dgen.github.io>.

1. Introduction

In recent years, generative models have significantly advanced the field of 3D object generation, transforming computer vision and graphics. The rapid progress of diffusion models led to remarkable improvements in syn-

* Equal contribution

† Corresponding author

This project is supported by Quwan Technology.

thesizing highly realistic 2D images. These developments have opened new possibilities for creating complex 3D objects, essential for applications such as virtual reality, gaming, and scientific simulations. However, extending these advancements from 2D to 3D remains challenging due to the inherent complexity of 3D geometry, the scarcity of high-quality datasets, and the high computational costs involved.

Existing 3D generation methods generally fall into two categories: optimization-based approaches and direct generation approaches. Optimization-based methods, such as DreamFusion [37], ProlificDreamer [52] and LucidDreamer [23], utilize pre-trained 2D diffusion models to create 3D content. While these methods have shown promising results, they are often time-consuming during inference time because of the intensive iterative optimization required to update the 3D representation.

In contrast, direct 3D generation aims to create 3D models directly, without extensive optimization. Typical approaches include InstantMesh [59], Unique3D [55], PRM [10], Clay [63], Craftsman [21], and Direct3D [56]. These methods are favored in practical 3D generation systems due to their fast generation speed. However, despite their potential, these techniques rely heavily on large-scale 3D datasets, which often suffer from limitations in quality and availability. For example, Objaverse-XL [7], the largest 3D dataset available, contains approximately 10 million samples, but around 70% of the data is compromised by missing textures, low resolution, or poor aesthetics. This lack of high-quality data significantly hampers the effectiveness of these direct 3D generation methods. In contrast, 2D datasets such as LAION-5B [43] contain billions of high-quality images, underscoring the disparity in data availability between 2D and 3D content.

Recent research has indicated that 2D diffusion models inherently possess powerful 3D priors that can be utilized for 3D understanding and generation [12, 17]. However, these methods focus on producing 2.5D representations like depth or normal maps, which provide a partial view of the 3D structure and insufficient for full 3D generation. This raises an important question: Can we leverage the expressive priors learned by 2D diffusion models to create complete 3D representations?

In this work, we introduce Kiss3DGen, a simple yet effective framework for 3D asset generation. Our approach involves fine-tuning a powerful diffusion transformer model (DiT), such as Flux [1], for 3D generation tasks. Specifically, given a set of 3D objects for training, we first render each object into four distinct views along with their corresponding normal maps, forming a comprehensive representation we call the “3D Bundle Image”, which captures both geometry and texture. Note that 3D Bundle Image is essentially an image, making

it highly compatible with the existing knowledge of pre-trained diffusion models, thus ensuring easy training and strong generalization capabilities.

Concretely, we use GPT-4V(ision) to generate descriptive captions for these 3D Bundle Images based on their RGB portion. These caption-image pairs are then used to fine-tune the Flux Model with Low-Rank Adaptation (LoRA) [15], resulting in our core model: Kiss3DGen-Base. During testing, given a user-provided caption, Kiss3DGen generates a 3D Bundle Image, which is subsequently optimized using existing mesh reconstruction approaches (e.g., NeuS [29], MeshFormer [25], ISOMER [55]) produce a final 3D mesh. Thanks to the generative power of the Flux model, Kiss3DGen is capable of generating 3D content beyond the training distribution.

Since Kiss3DGen is fundamentally a diffusion model, it is naturally compatible with various diffusion-based techniques. In this work, we demonstrate this by incorporating ControlNet [62] to extend Kiss3DGen’s capabilities, which is termed Kiss3DGen-ControlNet. Given a 3D mesh, Kiss3DGen first renders a 3D Bundle Image, which serves as a condition for ControlNet to perform tasks such as mesh enhancement or editing tasks. Leveraging ControlNet, Kiss3DGen efficiently handles Image-to-3D generation (e.g., enhancing a coarse mesh), 3D editing (e.g., via ControlNet-Canny), and quality enhancement via ControlNet-Tile, which rebuilds low-quality 3D meshes to higher-quality.

Our contributions can be summarized as follows:

- We propose Kiss3DGen, a simple yet effective approach that retargets 2D diffusion models for 3D asset generation tasks.
- We show that Kiss3DGen can be seamlessly integrated with ControlNet, enabling diverse functionalities such as text-to-3D generation, image-to-3D generation, 3D editing, and 3D asset enhancement.
- Extensive experiments demonstrate that the Kiss3DGen model achieves state-of-the-art performance across various tasks.

2. Related Works

2.1. 3D Generation

Distillation-based 3D generation. The rapid advancement of large-scale generative models, particularly the remarkable success of 2D diffusion models [40–42], has driven significant progress in 3D reconstruction. Pioneering methods like DreamFusion [37] and SJC [51] try to distill a 3D representation like NeRF [34] or Gaussian Splatting [18] from a 2D image diffusion by a Score Distillation Sampling (SDS) loss and its variants. Follow-up distillation-based methods [4, 23, 24, 32, 39, 52] attempt to improve the quality and efficiency. While these meth-

ods offer versatility and general applicability across diverse object categories, they frequently encounter challenges with convergence due to noisy and inconsistent gradient signals. This instability often results in incomplete reconstructions or artifacts, such as the “multi-faced Janus problem”. Additionally, these methods generally demand extensive optimization time, limiting their practicality in real-world applications where speed and efficiency are essential.

Multi-view generation. Multi-view generation aims to generate multiple viewpoints of an object with a given image or textual prompt. Early efforts in multi-view generation models, such as MVDream [47], have successfully adapted pretrained text-to-image diffusion models [41] to generate object-centric multi-view images. Concurrent studies [26, 33, 46, 49] explore image-conditioned multi-view generation, achieving impressive quality in multi-view outputs. However, they primarily focus on multi-view RGB generation and often overlook the challenges of 3D reconstruction. While some studies [31, 55] explore the separate generation of color and normal maps, another line of research achieves joint generation of both modalities with model context switcher [20, 29]. The generation of normal images profoundly enhances the accuracy and quality of 3D shape formation. However, these approaches significantly modify the network architecture or the input-output patterns of the pretrained models and discard textural conditions, thus reducing their effectiveness for general tasks like 3D refinement and editing. To our knowledge, no existing model unifies color and geometry generation with text conditions, a capability we believe is crucial for extending models to various 3D generation tasks, which we have achieved (see Sec. 3).

Feedforward 3D generation. Beyond multi-view image generation, feedforward 3D generation refers to generating 3D representations of objects. Notably, several models within the Large Reconstruction Model (LRM) series, including MeshLRM [53], LGM [48], Instant3D [19], InstantMesh [59], and GS-LRM [61], use a single-image-to-multi-view generation approach to produce fixed-pose multi-view images, followed by a robust sparse-view reconstruction model to generate the final 3D assets. Distinct from the LRM series, another category of models incorporates diffusion models, such as Craftsman [21] and Direct3D [56], which typically follow a two-stage process: first, training a 3D variational autoencoder (VAE) to encode 3D structural information, then applying a latent diffusion model to generate 3D assets conditioned on input text or images. While these models yield high-quality results, their generalizability and robustness are limited due to constrained 3D training datasets. In contrast, some studies leverage the strong priors of 2D diffusion

models. For example, methods such as ATT3D [30] and LATTE3D [58] distill 2D diffusion model priors to construct feed-forward text-to-3D generation models, though the quality of 3D assets generated in these approaches remains suboptimal. Similarly, models such as PI3D [28] employ 2D diffusion priors to generate triplane representations; however, this approach modifies the original training data structure of the stable diffusion model, disrupting its intrinsic priors and significantly limiting its applicability in open-domain generation. Kiss3DGen can seamlessly integrate with these models in multiple ways. On the one hand, these models can generate (coarse) meshes that contribute to enhancing the stability of Kiss3DGen’s reconstruction phase. On the other hand, Kiss3DGen can refine and further edit these meshes with prior knowledge inherited from diffusion models. This largely improves their level of detail and overall visual quality, as well as the capacities to generate open-domain 3d assets.

2.2. 3D Enhancement and Editing

3D enhancement and editing means to repair and refine initial low-quality objects, or add, delete and stylize objects using text or user interaction. Early works like EditNeRF [27] and CLIP-NeRF [50] achieves simple part removal and colorization by feeding different codes into pretrained conditional NeRF. For more fine-grained enhancement and editing, methods [11, 16, 35, 36, 44] combine Instructpix2pix [2] and SDS to add precise text editing instructions. However, these approaches are often time-consuming due to their optimization-based frameworks, and their implicit 3D representations are not well-suited for mesh enhancement. DreamEditor [64] and Focal Dreamer [22] propose to use mesh-based neural field and DMTet [45] respectively for direct mesh optimization. Progressive3D [5] and Focal Dreamer [22] achieve convenient user interaction by spatial masking and 3D composition. Additionally, MVEdit [3] proposes a text-to-3D diffusion model for 3D initialization followed by a refining process guided by a 2D diffusion prior. Coin3D [8] uses 3D volume adapter and coarse proxy to aid score distillation process and achieves stronger 3D control. Despite these advancements, such methods primarily focus on sculpting the 3D representation in a view-independent manner, often resulting in suboptimal global coherence in both geometry and texture. In contrast, our proposed Kiss3DGen naturally offers 3D enhancement and editing, achieving high-quality results with a streamlined pipeline.

3. Proposed Method

In this section, we present an in-depth explanation of Kiss3DGen, which repurposes a powerful diffusion transformer model (DiT) for 3D generation tasks. In

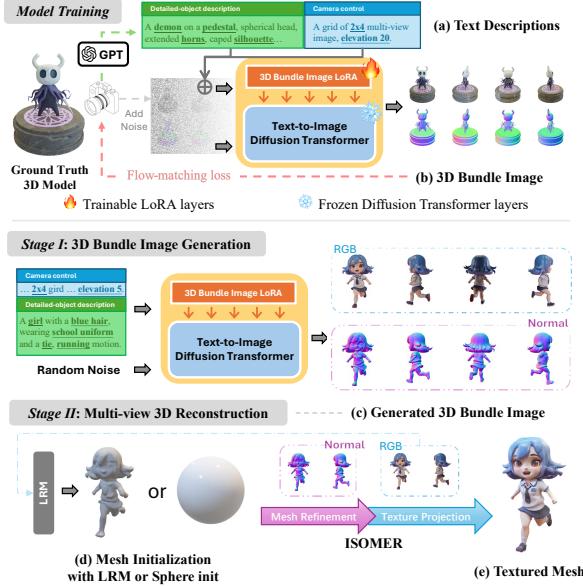


Figure 2. The overview of our text-to-3D training and generation framework. In this work, we curate a high-quality text-3D dataset, then train a LoRA [15] layer for text to 3D bundle image (Sec. 3) generation upon a pretrained text-to-image diffusion transformer model with flow matching. Our framework generates 3D assets with text condition in two stages: the 3D-Bundle-Image generation (*Stage I*) and the 3D reconstruction (*Stage II*). In *Stage I*, we generate 3D bundle image with our Kiss3DGen base model guided by text prompts. In *Stage II*, we reconstruct the geometry and texture of the 3D asset via LRM [14, 59] or sphere initialization followed by optimization-based mesh refinement and texture projection approach, i.e., ISOMER [55]. Zoom in for details.

Sec. 3.1, we explain how to train the base model to generate 3D Bundle Images, ultimately enabling text-to-3D generation, termed as Kiss3DGen-Base. Notably, Kiss3DGen-Base is essentially an image generation model that can be combined with many existing techniques to achieve more advanced functionalities, which will be discussed in Sec. 3.2.

3.1. Kiss3DGen-Base

Kiss3DGen-Base is designed to generate high-quality 3D Bundle Images that encapsulate both the geometry and texture information of 3D objects. Then, we leverage ISOMER [55] to generate textured 3D meshes with these 3D Bundle Images. The overarching design principle is to align the 3D Bundle Image with the prior distribution of pre-trained image diffusion models, thereby preserving the original generative capabilities of the pre-trained model to the greatest extent possible.

3D Bundle Image. Given a 3D object, we render it into four distinct views along with their corresponding normal maps, creating a comprehensive multi-view representation called the 3D Bundle Image, as shown in Fig.

2(b). This approach is based on two key insights. First, the combination of normal and RGB images from multiple views capture all necessary information to form a complete 3D object, which is then converted into a 3D mesh with ISOMER [55]. Second, since the 3D Bundle Image is essentially a 2D representation, it naturally aligns with the prior information in pre-trained 2D diffusion models, allowing us to leverage their generative capabilities without altering the input-output structure. This ensures the model effectively integrates and utilizes the learned priors.

Model. Learning the 3D Bundle Image is a non-trivial task, as the model must capture the relationships between different views and the correspondences between RGB images and normal maps, which are inherently complicated. Given the spatial distances between RGB images and normal maps, modeling long-range dependencies is crucial. To address this, we employ a DiT model, i.e., Flux [1], whose attention blocks are particularly effective at capturing these long-range dependencies, ensuring coherent multi-view and cross-modal relationships are properly modeled.

Captioning. To enhance the training process and leverage text-image correspondence, we generate descriptive captions for each 3D Bundle Image. We use GPT-4V to provide detailed captions that describe the content of each Bundle Image, including visual attributes such as color, shape, and surface properties. These captions help encode semantic information about the 3D objects, providing an additional supervisory signal during training that ensures the model learns to associate textual descriptions with specific geometric and visual features. An example caption can be found in Fig. 2(a).

Training and Inference. With the 3D bundle images and the captions, we train a LoRA to retarget the pre-trained Flux [1] model to generate the 3D Bundle Image (Fig. 2(b)). As a result, the model can generate a 3D Bundle Image from text prompt, producing a set of image and normal map pairs from four distinct viewpoints. Then, we can employ ISOMER [55] to optimize a mesh, which could be initialized with a sphere, as shown in Fig. 2. In practice, we also found that initializing the mesh with LRM [59] is more robust with a bit more inference time. Thus we adopt this strategy in this paper. A detailed study will be shown in the supplementary file. This process enables Text-to-3D Generation.

3.2. Kiss3DGen-ControlNet

To extend the capabilities of Kiss3DGen beyond direct generation, we introduce Kiss3DGen-ControlNet, which incorporates ControlNet [62] to handle a variety of 3D-related tasks such as enhancement, editing, and image-to-3D generation. In this section, we start with 3D enhancement as a basic application and ex-

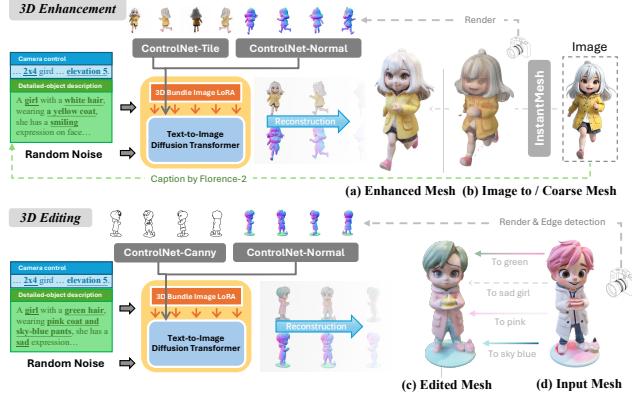


Figure 3. 3D enhancement and editing with Kiss3DGen. In order to achieve high-quality image-to-3D generation, we incorporate the existing image-to-3D pipeline [59] with our general 3D enhancement pipeline. Please zoom in for details.

pand into multiple use cases, providing examples of how Kiss3DGen-ControlNet can be applied for the aforementioned tasks. It should be noted that its potential usages are far more extensive.

3D Enhancement. Given a low-quality mesh (e.g., Fig. 3(b)), which may suffer from geometry lacking detail or overly blurred textures, we can render it into a 3D Bundle Image and then process it through the ControlNet-Tile and ControlNet-Normal model within Kiss3DGen. Originally, ControlNet-Tile was used for super-resolution of images, preserving color and content while leveraging diffusion model priors to enhance details. We found that this approach is also suitable to enhance our 3D Bundle Images, including the RGB images and normal maps. Note that 3D enhancement requires retaining the semantic integrity of the original mesh as much as possible. To facilitate this, we use the Florence-2 [57] to generate captions for the RGB portion of the 3D Bundle Image. With the enhanced 3D Bundle Images, we use ISOMER [55] to further refine the input mesh. The enhanced RGB images and normal maps enrich the details of texture and geometry respectively.

It should be noted that incorporating ControlNet can limit changes to the edited model, potentially constraining improvements in geometry and texture details. To address this, we introduce two hyperparameters, λ_1 and λ_2 . $\lambda_1 \in [0, 1]$ represents the ControlNet Strength, determining how strongly the ControlNet Branch adds to the original network, i.e., $y = F(x) + \lambda_1 F_c(c)$, where $F(x)$ is the feature of main branch, $F_c(c)$ is the feature of the control branch. $\lambda_2 \in [0, 1]$ represents the fraction of diffusion steps during which ControlNet is active. Specifically, ControlNet is applied from step 0 to $\lambda_2 T$, and is not used from step $\lambda_2 T + 1$ to T . Empirically, λ_1 values between 0.05 and 0.8, and λ_2 values between 0.1 and 0.7, yield good results. In most experiments, we set $\lambda_1 = 0.6$ and $\lambda_2 = 0.3$ for a balance between

enhancement and flexibility.

3D Editing. Similar to the 3D enhancement mentioned above, by decreasing the weight of ControlNet and allowing users to provide customized captions, Kiss3DGen-ControlNet can effectively perform 3D editing. As shown in Fig. 3(c, d), this pipeline allows users to alter specific attributes of the 3D object, such as shape or texture, while maintaining overall coherence with the original model. Note that this is not the only way to achieve 3D editing. We have tried to convert the 3D Bundle Image with Canny operation, then apply ControlNet-Canny, or simply use SDEdit to edit the 3D Bundle Image; both work well. In this paper, we simply adopt ControlNet-Canny and ControlNet-Normal, then apply $\lambda_1 = 0.3$, $\lambda_2 = 0.5$.

Image-to-3D Generation. Kiss3DGen-ControlNet supports Image-to-3D Generation. By using existing methods (e.g., InstantMesh [59]) to generate a coarse mesh from a given 2D image, Kiss3DGen-ControlNet can refine this initial output, transforming it from a low-quality, rough mesh into a high-quality 3D model. In Fig. 3(a, b), we demonstrate an example of image-to-3D generation by reusing our 3D enhancement pipeline, where we further replace one of the RGB views in the rendered 3D Bundle Image with the input image. This two-stage approach allows for the efficient transformation of 2D inputs into detailed 3D objects, utilizing the enhancement capabilities to improve mesh quality.

4. Experiments

4.1. Dataset

Due to the inconsistent quality of the original Objaverse dataset [6], we initially excluded objects that lacked texture maps or had low polygon counts. We then conducted meticulous manual curation to remove low-quality samples, such as incomplete objects, scanned flat surfaces, and large-scale scene samples. Additionally, given the irregular orientations of objects in the Objaverse dataset, we manually annotated and corrected the front-facing angle for each object. This rigorous refinement process resulted in a collection of 147k high-quality objects, which we used to train the Kiss3DGen-Base model. In addition, we curated a specialized dataset of 4k high-quality 3D cartoon-style human body models from the internet. This dataset was used to train the Kiss3DGen-Doll model, specifically designed to support cartoon-style character generation. To render these objects, we used Blender with a camera distance set to 4.5 units and a field of view (FoV) of 30 degrees. For each object, we used Blender to render four views separated by 90 degrees in azimuth, with the first view aligned to the front-facing angle of the object. The elevation is fixed as 5 degree. Both RGB and Normal maps

were rendered for each view at a resolution of 512×512 . The four view images were then used as inputs for 3D-aware caption annotation in GPT-4V.

4.2. Implementation Details

For the training of Kiss3DGen-Base, we utilized FLUX.1-dev [1] as our base model, training it with Low-Rank Adaptation (LoRA) [15] on 8 NVIDIA A800 80GB GPUs for 3 days, completing 16 epochs with a batch size of 4. The Adam optimizer was employed with a fixed learning rate of 8×10^{-4} , and training was conducted with bf16 precision. The LoRA rank (network dimension) was set to 128.

4.3. Evaluation

For evaluation dataset, we conducted quantitative comparisons using the Google Scanned Objects (GSO) dataset [9]. For the *text-to-3D* evaluation, we randomly sampled 100 objects, rendering four orthogonal views per object and then annotated using GPT-4V to generate text inputs for the test set. For the *image-to-3D* evaluation, we sampled 200 objects and rendered a single front-facing view for each, serving as the input for quantitative assessment.

For the evaluation protocol, we assessed alignment with text descriptions and the visual quality of the generated results. For the *text-to-3D* task, without precise 3D ground truth, we measured alignment using CLIP score [38] by rendering four orthogonal views of each generated object. Also, we used Q-Align [54], a large multi-modal model, to evaluate the quality and aesthetics of the rendered images. For the *image-to-3D* task, we evaluated both 2D visual quality and 3D geometric quality. For the 2D visual evaluation, we rendered novel views from the generation 3D mesh and compared them to the ground truth views using PSNR, SSIM, and LPIPS as metrics. For the 3D geometric evaluation, we focused on comparing the generation 3D meshes to the ground truth. First, we aligned the coordinate systems of the generated and ground truth meshes. Next, we repositioned and rescaled all meshes into a cube of size $[-1, 1]^3$. We reported Chamfer Distance (CD) and F-Score (FS) at a threshold of 0.1, calculated by uniformly sampling 16K points from the mesh surfaces.

4.4. Comparison with State-of-the-Art Methods

In this section, we compare our approach with several state-of-the-art methods across three tasks: text-to-multi-view synthesis, text-to-3D generation, and image-to-3D generation. For text-to-multi-view synthesis, we compare with MVDream [47], which introduces a multi-view attention mechanism within its model to facilitate multi-view information interaction. This approach aims to improve consistency across generated views by al-

Table 1. The quantitative comparison with MVDream in **text-to-multi-view synthesis** shows that our method outperforms MVDream [47] by a large margin. Notably, our method even surpasses the “Real Data” in both “Quality” and “Aesthetic” metrics, where “Real Data” refers to the multiple views used to generate text by GPT-4V.

Method	Dataset size	CLIP↑	Quality↑	Aesthetic↑
Real Data	N/A	0.884	3.138	1.911
MVDream	350K	0.809	2.509	1.526
Ours-Base	147K	0.844	3.248	1.94
Ours-50K	50K	0.804	2.972	1.879

Table 2. Quantitative comparison of **text-to-3D generation** results after rendering, evaluated in terms of CLIP-score, Quality, and Aesthetic metrics. Our method outperforms 3DTopia [13], Direct2.5 [31], and Hunyuan3D-1.0 [60] across all metrics, indicating a significant improvement in alignment with textual descriptions and visual quality.

Method	Dataset size	CLIP↑	Quality↑	Aesthetic↑
3DTopia	320k	0.694	2.145	1.538
Direct2.5	500k	0.773	2.158	1.459
Hunyuan3D-1.0	N/A	0.792	2.517	1.504
Ours-Base	147k	0.837	2.700	1.800
Ours-50K	50k	0.804	2.716	1.601

Table 3. Quantitative comparison of **image-to-3D generation** methods across multiple metrics. Since CraftsMan generates only geometry without textures, the comparison with this method is limited to 3D geometry metrics (Chamfer Distance (CD) and F-Score (FS)).

Method	Dataset size	CD↓	FS↑	PSNR↑	SSIM↑	LPIPS↓
CraftsMan	170k	0.178	0.739	N/A	N/A	N/A
Unique3D	50k	0.217	0.654	19.237	0.898	0.127
Hunyuan3D-1.0	N/A	0.153	0.768	16.652	0.885	0.123
Ours-Base	147k	0.149	0.769	20.348	0.902	0.116
Ours-50K	50k	0.151	0.766	20.215	0.884	0.131

lowing information to be shared among them. For text-to-3D generation, we compare our method with three recent approaches—3DTopia [13], Direct2.5 [31], and Hunyuan3D-1.0 [60]—that claim capabilities in text-to-3D generation. 3DTopia trains a latent diffusion model to generate Tri-plane representations for 3D object synthesis. Direct2.5 takes a different approach by utilizing two separate diffusion models to generate Normal maps and corresponding RGB maps, which are then used to reconstruct the geometry and apply textures. In contrast, Hunyuan3D-1.0 follows large reconstruction model (LRM) that first generates a single image from text and then expands this into six multi-view images for sparse-view reconstruction. For image-to-3D generation, we compare the Hunyuan3D-1.0, the diffusion-based model CraftsMan [21], and Unique3D [55], a two stage generation method similar to ours.

Text-to-Multi-View Synthesis. As shown in Tab. 1, we conducted a quantitative evaluation comparing our method to MVDream. Our method demonstrates a sub-

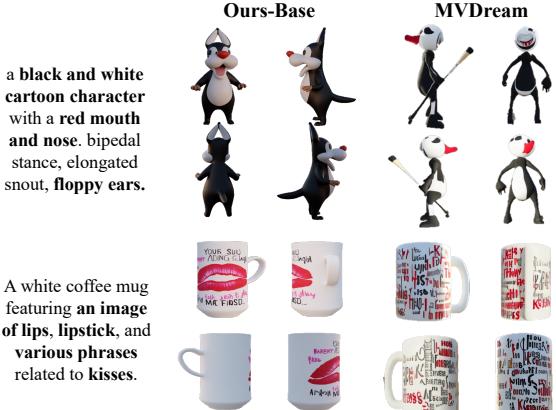


Figure 4. **Qualitative comparisons with MVDream [47] in text-to-multiview generation.** In comparison, our method produces significantly better results in both text-image alignment and geometric coherence.

stantial improvement over MVDream across all metrics, indicating enhanced consistency and quality across multiple views. Notably, our approach outperforms the “Real Data” in both “Quality” and “Aesthetic” scores, which may seem counter-intuitive. However, this makes sense considering that our model inherits a lot of knowledge from Flux model which tends to produce high-quality images. Furthermore, even the model trained on a reduced dataset of just 50K samples achieves competitive results, indicating that our approach is highly data-efficient and can yield strong performance with limited training data. Also, we performed qualitative comparisons in Fig. 4 which illustrate the superior multi-view coherence and realism achieved by our approach.

Text-to-3D Generation. As shown in Tab. 2, we quantitatively compared our approach with 3DTopia, Direct2.5, and Hunyuan3D-1.0 in terms of alignment with text, visual quality, and aesthetic appeal. Our method consistently outperforms the others across all metrics, particularly in CLIP-score, which reflects improved alignment with the input text. Furthermore, we achieve higher Quality and Aesthetic scores, showcasing our method’s ability to generate not only accurate but also visually pleasing 3D representations. Noticed that these metrics are calculated by rendering the generated 3D results into four orthogonal views. In addition, qualitative comparisons are provided in Fig. 5. In Fig. 6, we show both our Base and Doll models can serve high-quality text-to-3D generation, even for cases that are out of the training domain. We also conducted qualitative comparisons on 3D mesh enhancement and editing with MVEdit [3] in Fig. 7, demonstrating our ability to perform more precise mesh enhancement or editing.

Image to 3D generation In Tab. 3, we present a quantitative comparison of various image-to-3D generation methods across multiple metrics. Our approach demon-

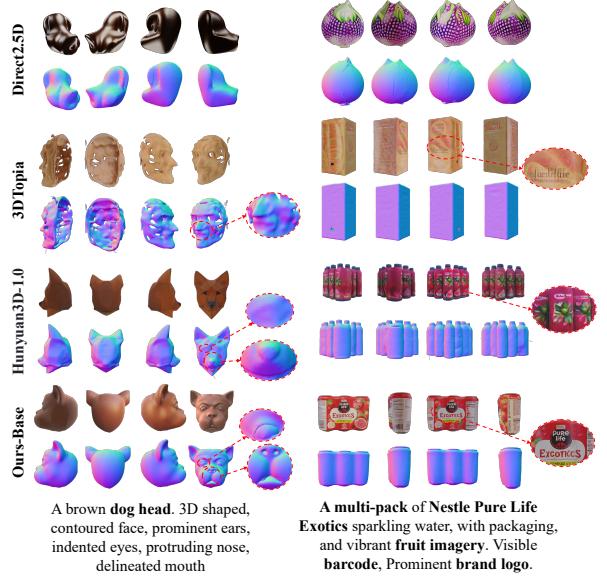


Figure 5. **Qualitative comparisons with state-of-the-art methods for text-to-3D generation.** It demonstrates that Kiss3DGen achieves the highest quality 3D mesh, delivering more accurate texture generation from the input prompts compared to others.



Figure 6. Text-to-3D comparison between our Base and Doll models (Sec. 4.1). Each model generates different results with different seeds. All images are rendered from 3D mesh.

strates superior performance, outperforming other methods in Chamfer Distance (CD), F-Score (FS), PSNR, SSIM, and LPIPS scores. Notably, our model achieves strong results even with a reduced training dataset (Ours-50K), indicating its efficiency and robustness in generating high-quality 3D representations from 2D images. In addition to these quantitative results, Fig. 8 provides qualitative comparisons that highlight the improved texture fidelity, structural coherence, and overall visual quality achieved by our approach.

4.5. Ablation Study

The mechanism of 3D Bundle Image As discussed earlier, we propose a novel approach of combining RGB and normal maps into a single image, referred to as the

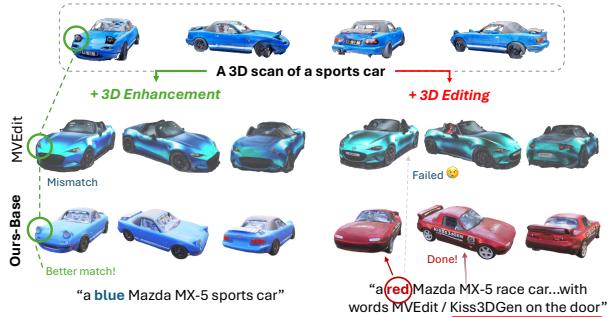


Figure 7. Qualitative comparison on 3D mesh Enhancement and Editing with MVEdit [3]. Our results (line 2) maintain much better consistency with the input mesh in enhancement and better align with the given text in editing.

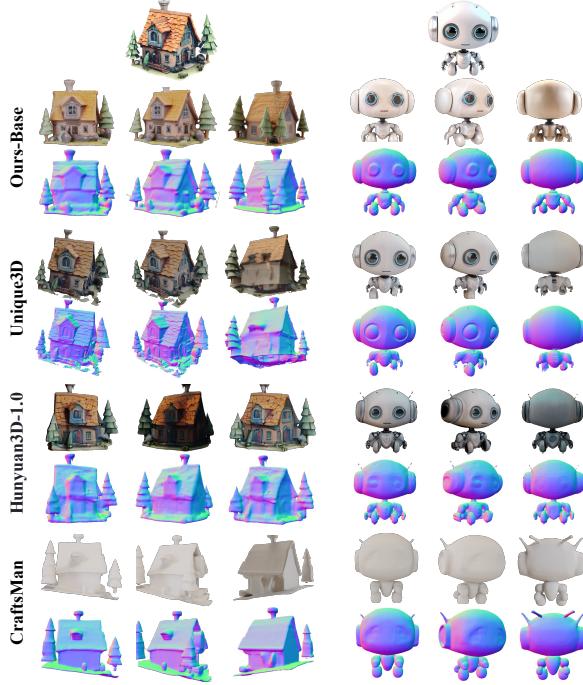


Figure 8. Qualitative comparisons with state-of-the-art methods for image-to-3D generation. We achieve the highest quality 3D mesh, delivering more accurate and realistic texture generation from input images compared to other models.

“3D Bundle Image,” for model training. We compare this approach with the “Switcher” mechanism used in prior works [20, 29]. Unlike our model, which generates RGB images and normal maps concurrently, these works employ a “Switcher” to selectively produce either RGB images or normal maps. As illustrated in Fig. 9, the “Switcher” mechanism fails to maintain coherence between the two outputs. In contrast, our “3D Bundle Image” achieves significantly higher consistency. This improvement is due to the DiT model’s architecture, particularly its attention blocks, which effectively capture long-range dependencies and interactions. However the

“Switcher” mechanism processes the modalities separately and does not leverage this advantage.

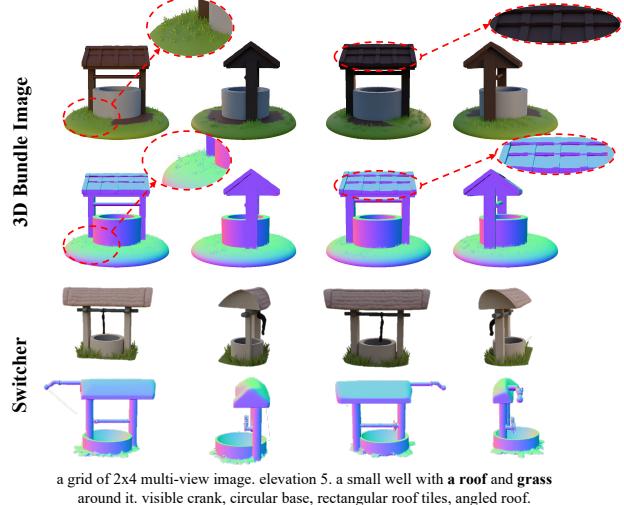


Figure 9. Ablating the mechanisms of generating multiview RGB and normal maps. Both our “3D Bundle Image” and “Switcher” [20, 29] are built upon Flux.1-dev [1] model.

Dataset scale In Sec. 4, we compare the Kiss3DGen-Base model with baseline models, demonstrating that it achieves state-of-the-art performance. However, we note that the scale of datasets varies significantly across different works. For instance, 3DTopia [13] was trained on over 320k 3D objects, Direct2.5 [31] used 500k, and Unique3D [55] employed only 50k objects. We reduce our dataset to 50k objects and train a model named Kiss3DGen-50k, and find that the model still performs well in most tasks (Tab. 2), proving the effectiveness of our framework. In comparison, we find that the Kiss3DGen model trained on a small scale of data sometimes failed to generate a 3D Bundle Image. See more details in our supplement.

5. Concluding Remarks

We introduce Kiss3DGen, a straightforward yet highly effective approach for a variety of 3D generation tasks. By leveraging knowledge from pretrained 2D diffusion models, it seamlessly integrates with existing techniques like ControlNet. Despite its simplicity, Kiss3DGen excels across tasks such as text-to-3D, image-to-3D, 3D enhancement, and 3D editing. It also demonstrates strong performance even with limited training data, benefiting from the preservation of knowledge in Flux. Additionally, the model generalizes well, enabling the generation of objects beyond its original 3D training set.

There is significant potential for further improvement, particularly in exploring the optimal representation of geometry (e.g., normal maps) and developing efficient methods for generating high-resolution views. These challenges will be addressed in our future work.

References

- [1] BlackForestLabs. Flux.1 model family. 2024. [2](#), [4](#), [6](#), [8](#)
- [2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. [3](#)
- [3] Hansheng Chen, Ruoxi Shi, Yulin Liu, Bokui Shen, Jiayuan Gu, Gordon Wetzstein, Hao Su, and Leonidas Guibas. Generic 3d diffusion adapter using controlled multi-view editing. *arXiv*, 2024. [3](#), [7](#), [8](#)
- [4] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *ICCV*, 2023. [2](#)
- [5] Xinhua Cheng, Tianyu Yang, Jianan Wang, Yu Li, Lei Zhang, Jian Zhang, and Li Yuan. Progressive3d: Progressively local editing for text-to-3d content creation with complex semantic prompts. *arXiv*, 2023. [3](#)
- [6] 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 (CVPR)*, 2023. [5](#)
- [7] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 2024. [2](#)
- [8] Wenqi Dong, Bangbang Yang, Lin Ma, Xiao Liu, Liyuan Cui, Hujun Bao, Yuwen Ma, and Zhaopeng Cui. Coin3d: Controllable and interactive 3d assets generation with proxy-guided conditioning. In *SIGGRAPH*, 2024. [3](#)
- [9] 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 *2022 International Conference on Robotics and Automation (ICRA)*, 2022. [6](#)
- [10] Wenhang Ge, Jiantao Lin, Guibao Shen, Jiawei Feng, Tao Hu, Xinli Xu, and Ying-Cong Chen. Prm: Photometric stereo based large reconstruction model. *arXiv preprint arXiv:2412.07371*, 2024. [2](#)
- [11] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instructnerf2nerf: Editing 3d scenes with instructions. In *ICCV*, 2023. [3](#)
- [12] Jing He, Haodong Li, Wei Yin, Yixun Liang, Leheng Li, Kaiqiang Zhou, Hongbo Liu, Bingbing Liu, and Ying-Cong Chen. Lotus: Diffusion-based visual foundation model for high-quality dense prediction. *arXiv*, 2024. [2](#)
- [13] Fangzhou Hong, Jiaxiang Tang, Ziang Cao, Min Shi, Tong Wu, Zhaoxi Chen, Shuai Yang, Tengfei Wang, Liang Pan, Dahua Lin, et al. 3dtopia: Large text-to-3d generation model with hybrid diffusion priors. *arXiv*, 2024. [6](#), [8](#)
- [14] 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. *arXiv*, 2024. [4](#)
- [15] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. [2](#), [4](#), [6](#)
- [16] Hiromichi Kamata, Yuiko Sakuma, Akio Hayakawa, Masato Ishii, and Takuya Narihira. Instruct 3d-to-3d: Text instruction guided 3d-to-3d conversion. *arXiv*, 2023. [3](#)
- [17] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, 2024. [2](#)
- [18] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *TOG*, 2023. [2](#)
- [19] 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. *arXiv*, 2023. [3](#)
- [20] Peng Li, Yuan Liu, Xiaoxiao Long, Feihu Zhang, Cheng Lin, Mengfei Li, Xingqun Qi, Shanghang Zhang, Wenhao Luo, Ping Tan, et al. Era3d: High-resolution multi-view diffusion using efficient row-wise attention. *arXiv*, 2024. [3](#), [8](#)
- [21] 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*, 2024. [2](#), [3](#), [6](#)
- [22] Yuhang Li, Yishun Dou, Yue Shi, Yu Lei, Xuanhong Chen, Yi Zhang, Peng Zhou, and Bingbing Ni. Focaldreamer: Text-driven 3d editing via focal-fusion assembly. In *AAAI*, 2024. [3](#)
- [23] Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In *CVPR*, 2024. [2](#)
- [24] 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. [2](#)
- [25] 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*, 2024. [2](#)
- [26] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. *arXiv*, 2023. [3](#)
- [27] Steven Liu, Xiuming Zhang, Zhoutong Zhang, Richard Zhang, Jun-Yan Zhu, and Bryan Russell. Editing conditional radiance fields. In *ICCV*, 2021. [3](#)

- [28] Ying-Tian Liu, Yuan-Chen Guo, Guan Luo, Heyi Sun, Wei Yin, and Song-Hai Zhang. Pi3d: Efficient text-to-3d generation with pseudo-image diffusion. In *CVPR*, 2024. 3
- [29] 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*, 2023. 2, 3, 8
- [30] Jonathan Lorraine, Kevin Xie, Xiaohui Zeng, Chen-Hsuan Lin, Towaki Takikawa, Nicholas Sharp, Tsung-Yi Lin, Ming-Yu Liu, Sanja Fidler, and James Lucas. Att3d: Amortized text-to-3d object synthesis. *ICCV*, 2023. 3
- [31] 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. *CVPR*, 2024. 3, 6, 8
- [32] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, and Andrea Vedaldi. Realfusion: 360deg reconstruction of any object from a single image. In *CVPR*, 2023. 2
- [33] Luke Melas-Kyriazi, Iro Laina, Christian Rupprecht, Natalia Neverova, Andrea Vedaldi, Oran Gafni, and Filippos Kokkinos. Im-3d: Iterative multiview diffusion and reconstruction for high-quality 3d generation. *ICML*, 2024. 3
- [34] 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. *ACM*, 2021. 2
- [35] Francesco Palandra, Andrea Sanchietti, Daniele Baieri, and Emanuele Rodolà. Gsedit: Efficient text-guided editing of 3d objects via gaussian splatting. *arXiv*, 2024. 3
- [36] Jangho Park, Gihyun Kwon, and Jong Chul Ye. Ednerf: Efficient text-guided editing of 3d scene using latent space nerf. *arXiv*, 2023. 3
- [37] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 2
- [38] 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. 6
- [39] 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. In *ICCV*, 2023. 2
- [40] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv*, 2022. 2
- [41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 3
- [42] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasempour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35, 2022. 2
- [43] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 2
- [44] Etai Sella, Gal Fiebelman, Peter Hedman, and Hadar Averbuch-Elor. Vox-e: Text-guided voxel editing of 3d objects. In *ICCV*, 2023. 3
- [45] Tianchang Shen, Jun Gao, Kangxue Yin, Ming-Yu Liu, and Sanja Fidler. Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. In *NIPS*, 2021. 3
- [46] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv*, 2023. 3
- [47] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv*, 2023. 3, 6, 7
- [48] 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. In *ECCV*, 2025. 3
- [49] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitrii Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. SV3D: Novel multi-view synthesis and 3D generation from a single image using latent video diffusion. In *ECCV*, 2024. 3
- [50] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *CVPR*, 2022. 3
- [51] 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. 2
- [52] 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. *NIPS*, 2024. 2
- [53] 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*, 2024. 3
- [54] Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Chunyi Li, Liang Liao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtai Zhai, and Weisi Lin. Q-align: Teaching lmms for visual scoring via discrete text-defined levels. *arXiv*, 2023. 6
- [55] 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*, 2024. 2, 3, 4, 5, 6, 8

- [56] 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*, 2024. [2](#), [3](#)
- [57] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. *arXiv*, 2023. [5](#)
- [58] Kevin Xie, Jonathan Lorraine, Tianshi Cao, Jun Gao, James Lucas, Antonio Torralba, Sanja Fidler, and Xiaohui Zeng. Latte3d: Large-scale amortized text-to-enhanced3d synthesis. *ECCV*, 2024. [3](#)
- [59] 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*, 2024. [2](#), [3](#), [4](#), [5](#)
- [60] Xianghui Yang, Huiwen Shi, Bowen Zhang, Fan Yang, Jiacheng Wang, Hongxu Zhao, Xinhai Liu, Xinzhou Wang, Qingxiang Lin, Jiaao Yu, et al. Hunyuan3d-1.0: A unified framework for text-to-3d and image-to-3d generation. *arXiv*, 2024. [6](#)
- [61] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *ECCV*, 2025. [3](#)
- [62] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. [2](#), [4](#)
- [63] 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. *arXiv*, 2024. [2](#)
- [64] Jingyu Zhuang, Chen Wang, Liang Lin, Lingjie Liu, and Guanbin Li. Dreameditor: Text-driven 3d scene editing with neural fields. In *SIGGRAPH Asia*, 2023. [3](#)

Kiss3DGen: Repurposing Image Diffusion Models for 3D Asset Generation

Supplementary Material

The supplementary file includes a demo video showcasing the performance of our model in various tasks, including text/image-to-3D generation, and 3D enhancement/editing. Additionally, we provide further studies and detailed explanations below to offer a deeper understanding of the model and its capabilities. We will release the code upon acceptance.

1. Ablating the Initialization of Mesh

In our manuscript, we adopt the off-the-shelf LRM [?] model or a simple sphere shape to initialize the coarse mesh, then refine the mesh with ISOMER [?]. We have also experimented with different settings, such as refining the mesh from a simple, sphere-shaped initialization. As shown in Fig. 1, the results are still of excellent overall quality; however, there appear to be more geometrical errors at unseen surfaces. We also conducted quantitative evaluations, as shown in Table 1 and Table 2. The quantitative results demonstrate that the LRM initialization generally outperforms the sphere initialization across most metrics.

Table 1. Quantitative comparison of generated results for **text-to-3D** with different initializations at the reconstruction stage.

Method	CLIP↑	Quality↑	Aesthetic↑
Init-LRM	0.837	2.700	1.800
Init-Sphere	0.8012	2.559	1.566

Table 2. Quantitative comparison of generated results for **image-to-3D** with different initializations at the reconstruction stage.

Method	CD↓	FS↑	PSNR↑	SSIM↑	LPIPS↓
Init-LRM	0.149	0.769	20.348	0.902	0.116
Init-Sphere	0.173	0.719	20.122	0.902	0.117

2. Ablating the number of steps in ISOMER

In our main manuscript, we proposed using the off-the-shelf LRM [? ?] model to initialize the coarse mesh, followed by ISOMER [?] to optimize and produce the final mesh. In the optimization step, there is a critical parameter that controls the number of geometry optimization steps. This parameter directly impacts the inference time. Specifically, when the number of steps is set to 50, the geometry optimization step takes approximately 5 seconds, while setting it to 100 increases the time to about 10 seconds. To understand the effect of this parameter, we conducted an ablation study, as shown in Fig. 2. The results indicate that increasing the

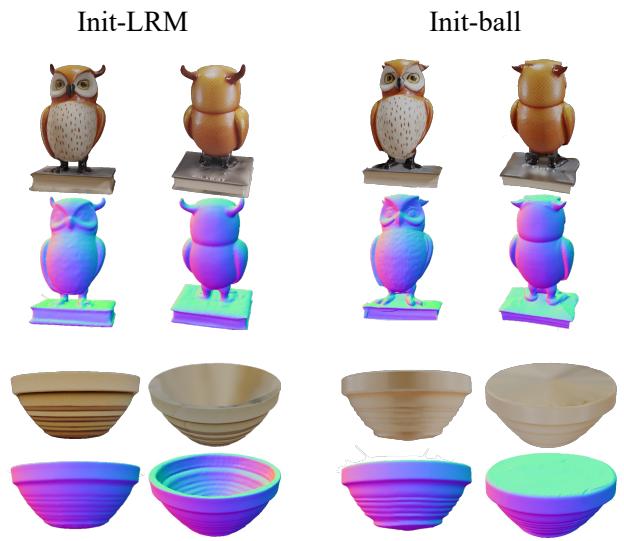


Figure 1. Qualitative comparison of 3D reconstruction results between different initializations in the reconstruction stage of our framework. The upper case (owl) shows that using LRM or sphere initialization yields similar results. The second row (bowl) shows that using sphere initialization may fail at capturing the concave geometric structure, while using LRM mitigates this problem.

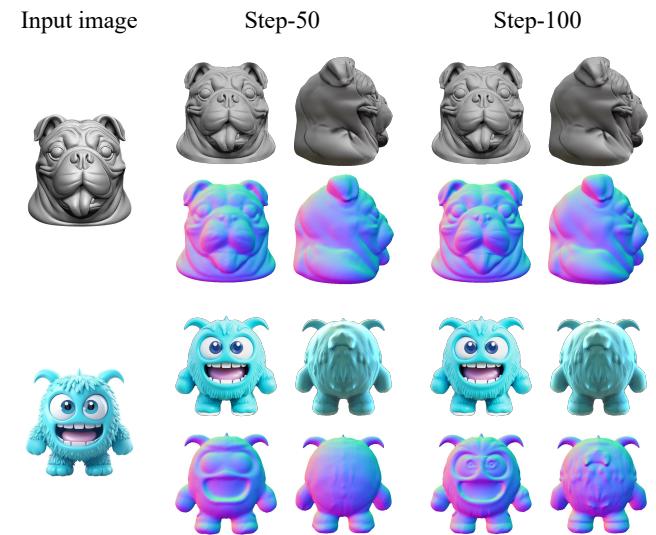


Figure 2. Qualitative comparison of 3D reconstruction results with different optimization steps with ISOMER [?]. As shown, optimizing with more steps leads to finer geometrical details.

number of steps leads to sharper and more refined geometry, albeit at the cost of longer computation time. It is worth

Table 3. Comparison of inference time with other methods in different tasks. (in seconds). “–” means unapplicable.

Task	ours	MVEdit	Hunyuan3D-1.0	CraftsMan	Unique3D	3DTopia	Direct2.5
Text-to-3D	56.8	–	105.0	–	–	240.0	163.6
Image-to-3D	87.3	–	79.9	6.0	37.2	–	–
3D-to-3D	71.7	360.0	–	–	–	–	–

noting that, in our main manuscript, we used a step value of 50 for all experiments to balance experimental efficiency and result quality. This analysis highlights the trade-off between optimization time and geometry refinement, providing guidance for parameter selection based on application requirements.

3. Compatibility and extensibility of methods.

As shown in Fig. 4, our method is compatible with reconstruction techniques besides ISOMER, such as Instant-NSR. Additionally, our approach retains DiT’s full capabilities, enabling seamless integration with tools like IP-Adapter, ControlNet, or Flux Redux¹ (Fig. 5), highlighting its adaptability and extensibility.

4. System efficiency.

In Tab. 3, we quantitatively measure the inference time of our framework and baseline methods on an A800 GPU, our approach achieves the best performance within reasonable inference time.

5. More qualitative comparisons.

We demonstrate more comparisons against Wonder3D++ and Michelangelo for image-to-3D and LucidDreamer for text-to-3D in Fig. 3. Our method achieves better results in texture details, semantic alignment, and text-3D consistency.

6. User Study

In our manuscript, we conduct quantitative evaluations comparing our method with baseline methods, demonstrating its superior performance. We also present a user study to assess user preferences.

The user study was conducted on Amazon Mechanical Turk², involving 180 participants. To ensure quality, we included attention-check questions to filter out inattentive responses, resulting in 80 qualified participants whose responses were analyzed. Ultimately, we collected 2,000 valid responses covering key aspects such as geometry quality, texture quality, and overall quality. The results used in user study are generated with the default hyper-parameters without any cherry-picking.

Figure 6 shows a screenshot of the user study questionnaire. The options included GIFs displaying orbital views

¹<https://blackforestlabs.ai/flux-1-tools>

²<https://www.mturk.com>

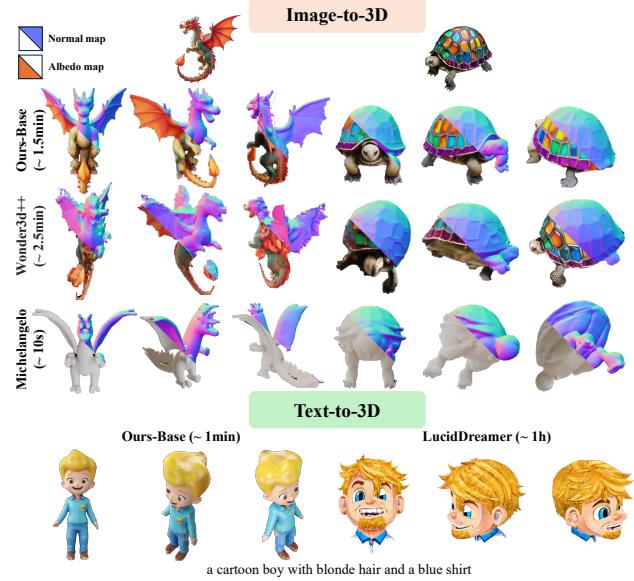


Figure 3. Qualitative comparisons with more state-of-the-art methods for *image-to-3D* and *text-to-3D* generation.



Figure 4. Visual comparisons of different reconstruction methods.



Figure 5. Visualization of *image-to-3d* with *redux*.

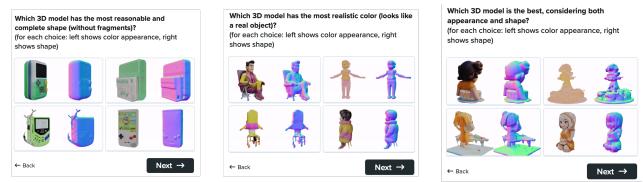


Figure 6. Screenshots of our user study questionnaire.

of the object in both color and normal space, allowing participants to better visualize the 3D structure and texture, thereby enhancing their ability to provide informed feedback.

For each case in the user study, we present a video to the users where the object rotates 360 degrees, with the left side displaying the RGB map and the right side showing the Normal map. Users are asked to select the best result based

on a series of questions. In terms of question design, we focus on several key aspects:

1. **Geometry**: "Which 3D model has the most reasonable and complete shape (without fragments)?"
2. **Texture**: "Which 3D model has the most realistic color (looks like a real object)?"
3. **Overall quality**: "Which 3D model is the best, considering both appearance and shape?"

The results of the user study are summarized in Tab. 4, where it can be observed that our method outperforms the baselines in terms of user preference for both geometry and texture quality, as well as overall impression.

Table 4. Study on user’s preference on 3D generation results of ours and baseline methods.

Category	Method	Percentage
Texture	Ours	35.47%
	Hunyuan	32.37%
	Unique3D	13.13%
	3Dtopia	6.75%
	Direct2.5D	12.28%
Geometry	Ours	37.61%
	Hunyuan	36.24%
	Unique3D	10.45%
	3Dtopia	5.13%
	Direct2.5D	10.56%
Overall Quality	Ours	38.72%
	Hunyuan	32.18%
	Unique3D	15.04%
	3Dtopia	6.49%
	Direct2.5D	7.57%

7. Applications and visualization

In our main paper, we introduce various applications with our model, including text-to-3d, image-to-3d, 3D editing and enhancement. We demonstrate more results in Fig. 8, Fig 9 and Fig. 10. Also, we attach a video to this supplementary to present the 3D generation results in a dynamic approach.

7.1. Advanced image to 3D

In Fig. 7, we illustrate a 3D generation pipeline that utilizes multi-modal conditions, including both images and text. Unlike most existing image-to-3D generation methods that produce 3D assets aligned solely with the input image, our framework introduces textual control over the generation outcomes, significantly enhancing the utility of 3D content creation from images. This capability allows for more nuanced and tailored 3D outputs, catering to specific user requirements. And notably, the application of our model extends beyond the examples presented in this paper.

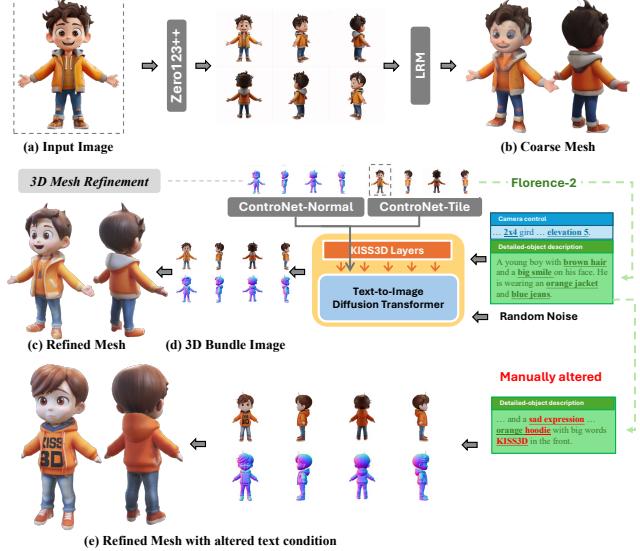


Figure 7. **Advanced image-to-3D pipeline with our framework.** In this case, we alter the text descriptions at the 3D mesh refinement stage and achieve accurate textual control on the refined result. Please zoom in for details.

8. Limitations

In this paper, we effectively adapt the pretrained 2D diffusion transformer model, specifically Flux [?], for the generation of 3D Bundle Images. To maximize the potential of the Flux model, we render our 3D dataset under varying environmental illuminations, enhancing its similarity to real-world images on which the Flux model was trained. As a result, the generated 3D Bundle Image retains lighting information, which was not disentangled from the model texture during the reconstruction phase of this work. We leave this for future study.



a white sugar skull with colorful polka dots, flowers, eyes, and teeth. Cranial dome, hollow eye sockets, nasal aperture, dental arch



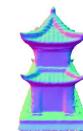
green lizard head with spikes symmetrical design, pronounced mane, detailed textures, elevated ridges, ornamental headpiece, sculptural form



A statue of a lion on a marble pedestal base, prominent wings, ornamental pedestal, sturdy base, beveled edges



A charming owl with festive Christmas details, sitting on a simple branch. The owl wears a small, red Santa hat with fluffy white trim.



a small Chinese pagoda. elevated base, sweeping roof, overhanging eaves, multi-tiered roof, rectangular footprint



a Coca Cola monster can with arms, legs. cylindrical body, two bending arms, two bending legs, extruded circular eyes, short cylindrical snout, protruding ears.

Figure 8. More show cases of **Text-to-3D** generation with our model. Please zoom in for details.

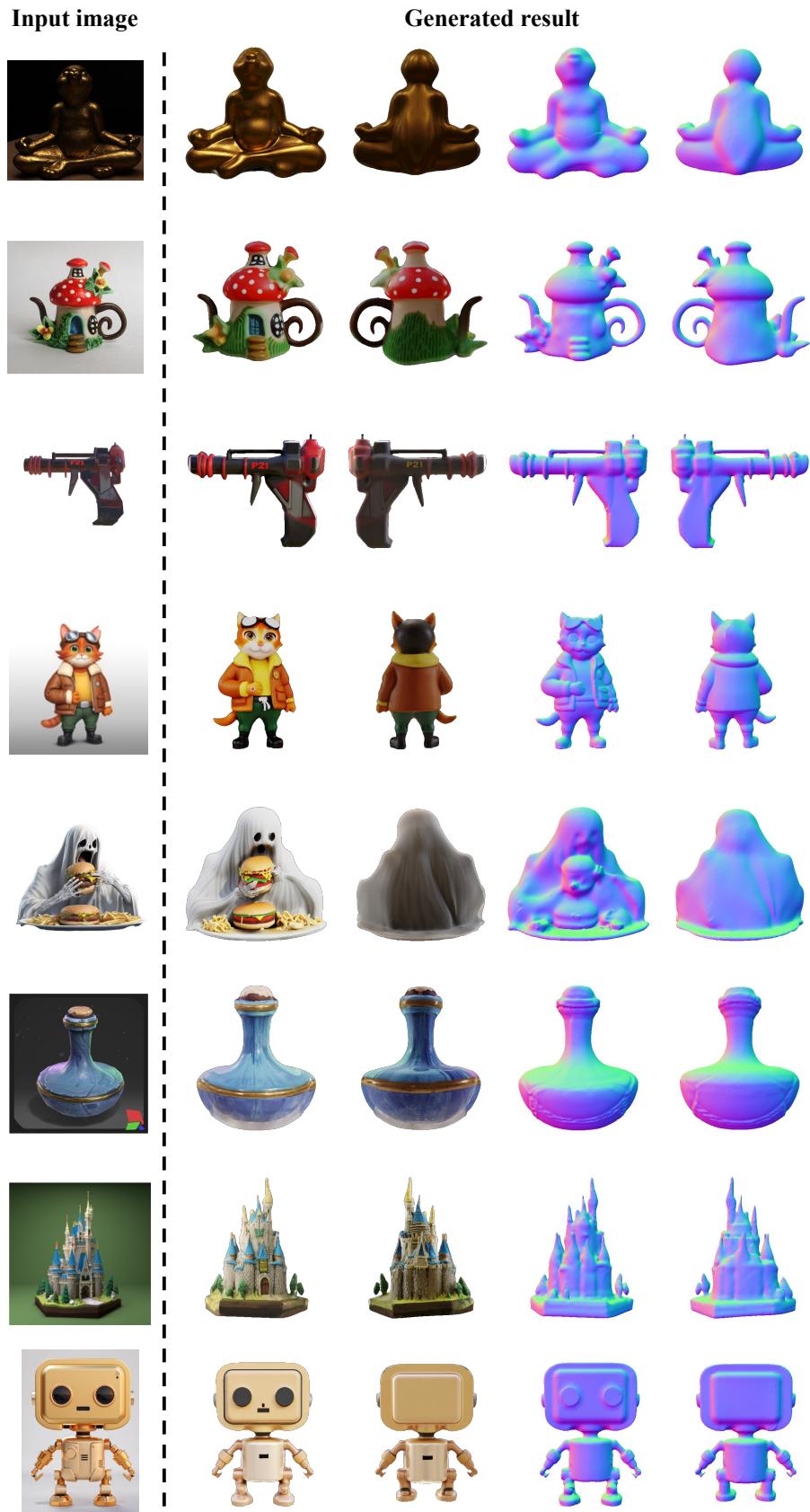


Figure 9. More show cases of **Image-to-3D** generation with our model. Please zoom in for details.



Figure 10. **3D enhancement and editing results with our model.** Notably, we adopt off-the-shelf controlNets [?], e.g., Normal, Canny and Tile, with our Kiss3DGen model to align the generation results with the input 3D models. For simplicity, we denote the fixed camera control caption as "...", and the detailed-object captions are manually crafted to achieve desired results. Please zoom in for details.