UNILAT3D: GEOMETRY-APPEARANCE UNIFIED LATENTS FOR SINGLE-STAGE 3D GENERATION

Anonymous authors
Paper under double-blind review

Figure 1: Gallery of UniLat3D. Our method generates high quality 3D assets in seconds.

ABSTRACT

High-fidelity 3D asset generation is crucial for various industries. While recent 3D pretrained models show strong capability in producing realistic content, most are built upon diffusion models and follow a two-stage pipeline that first generates geometry and then synthesizes appearance. Such a decoupled design tends to produce geometry–texture misalignment and non-negligible cost. In this paper, we propose **UniLat3D**, a unified framework that encodes geometry and appearance in a single latent space, enabling direct single-stage generation. Our key contribution is a geometry–appearance Unified VAE, which compresses high-resolution sparse features into a compact latent representation – **UniLat**. UniLat integrates structural and visual information into a dense low-resolution latent, which can be efficiently decoded into diverse 3D formats, *e.g.*, 3D Gaussians and meshes. Based on this unified representation, we train a single flow-matching model to map Gaussian noise directly into UniLat, eliminating redundant stages. Trained solely on public datasets, UniLat3D produces high-quality 3D assets in seconds from a single image, achieving superior appearance fidelity and geometric quality.

1 INTRODUCTION

3D content generation has witnessed rapid growth in recent years, becoming an increasingly essential capability across various applications, including game/film production, virtual/augmented reality, industrial design, and embodied AI. Recent advances in 3D generative frameworks (Zhang et al., 2024c; Hunyuan3D et al., 2025; Lai et al., 2025; Yang et al., 2024c; Zhao et al., 2025; Xiang et al., 2024; Li et al., 2025a; Hong et al., 2023; Zhang et al., 2024b; Ma et al., 2025; Ren et al., 2024a; Zou et al., 2024) have demonstrated impressive progress in synthesizing vivid and realistic

3D assets, while some approaches (Li et al., 2024; Wu et al., 2024b; 2025b; Chen et al., 2025d; Li et al., 2025c; Ye et al., 2025) dive into accurate geometry and fine-grained shape generation.

Despite this rapid progress, the majority of recent high-quality 3D generation frameworks are diffusion-based, and typically adopt a multi-stage design: geometry is generated first, followed by texture or appearance synthesis. This paradigm, rooted in the conventional separation of geometry and appearance, has been adopted by both latent-based pipelines (Xiang et al., 2024) and mesh-based frameworks (Li et al., 2025a; Hunyuan3D et al., 2025), remaining the prevailing design but entailing inherent drawbacks. First, the separate generation introduces an inevitable gap between geometry and appearance, potentially leading to misalignment with the target 3D asset. Second, the two-stage process introduces additional computation budget, *e.g.*, current mesh-based methods (Hunyuan3D et al., 2025) first generate the geometry, and then synthesize the corresponding texture based on both the condition image and geometry generated in the first stage. Notably, the research trajectory in both vision and graphics (Mildenhall et al., 2020; Kerbl et al., 2023) has long favored unification over separation – just as object detection evolved from multi-stage Faster R-CNN (Girshick, 2015) to single-stage YOLO (Redmon et al., 2016). We aim to create a similar unification of geometry and appearance generation, which is expected to offer more convenience and possibilities for exploring 3D generation under a more extensible and unified framework.

To this end, we introduce a unified 3D representation that inherently encodes geometry and appearance in a single latent space, enabling direct single-stage generation. Our key insight is that such a representation is naturally aligned—free from geometry–texture mismatches—and highly efficient, as it avoids redundant intermediate steps. Inspired by TRELLIS (Xiang et al., 2024), we first transform the 3D asset into sparse structured features. A unified variational autoencoder, UniVAE, is designed to compress high-resolution sparse features into a compact latent space, termed **UniLat**. The UniLat can then be efficiently upsampled and sparsified back onto high-resolution latents that serve as a universal basis for decoding into various renderable 3D representations, such as 3D Gaussians (Kerbl et al., 2023) and meshes. Thanks to the simplicity and expressive design of UniLat, we are able to, for the first time, achieve single-stage 3D generation through one flow-matching model that maps cubic Gaussian noise directly into the geometry-appearance unified latents. Beyond efficiency, UniLat also offers strong extensibility, which can serve as a versatile 3D prior that can be seamlessly integrated into large multimodal models, facilitating cross-modal understanding and generation. Our method, **UniLat3D**, trained only on publicly available datasets, achieves superior appearance fidelity while maintaining strong geometric accuracy, demonstrating the effectiveness of unifying geometry and appearance within a single-stage paradigm.

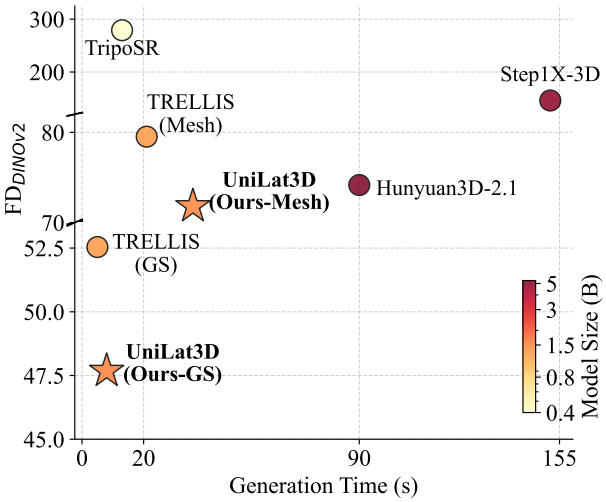

Figure 2: Evaluation on Toys4K (Stojanov et al., 2021). Colors stand for model sizes. Lower generation time and smaller FD_{DINOv2} indicate better performance, *i.e.* the left bottom corner.

Our contributions are summarized as follows.

- We propose a novel framework, UniLat3D, which bridges the gap between geometry and appearance by a single diffusion model in high-quality 3D generation.
- A novel UniLat representation is introduced by encoding geometry and appearance into a unified latent space, ensuring high-efficiency feature fusion.
- As in Fig. 2, extensive experiments demonstrate UniLat3D's state-of-the-art performance. We expect our framework to pave a novel way for exploring 3D generation in a more unified and scalable paradigm.

2 Related Works

2.1 3D Generation by Lifting 2D Diffusion Models

Lifting 2D diffusion models to 3D has been an effective but challenging approach. DreamFusion (Poole et al., 2022) proposes Score Distillation Sampling (SDS) to distill knowledge from the 2D diffusion model into a radiance field. Tang et al. (2023); Yi et al. (2023; 2024); Yin et al. (2023); Ren et al. (2023); Liu et al. (2024); Wang et al. (2023) follow this methodology to generate high-quality 3D Gaussians (Kerbl et al., 2023) in minutes. Meanwhile, Jain et al. (2022); Liu et al. (2023b); Shi et al. (2023); Huang et al. (2024); Long et al. (2023); Liu et al. (2023a); Yang et al. (2024a) fine-tune the image diffusion model to generate multi-view consistent images for synthesizing 3D assets. Video diffusion models (Yang et al., 2024d; Yu et al., 2024; Xing et al., 2024; Ren et al., 2025; Zhao et al., 2024; Gao et al., 2024; Wu et al., 2025a; Liang et al., 2024) are also explored to synthesize high-quality 3D/4D representations (Wu et al., 2024a; Yang et al., 2023; Zhang et al., 2024d; 2025b). However, most of these methods need iterative optimization from different views in each generation process, which takes a non-negligible cost, while hallucination may appear, *e.g.*, Janus phenomenon, due to the lack of 3D priors.

2.2 3D Generation by Pretraining 3D Foundation Models

With the emergence of large-scale 3D datasets, *e.g.*, Objaverse (Deitke et al., 2023), 3D foundation models have been constructed and pretrained to have strong reconstruction and generation abilities.

3D Foundation Reconstruction Models. Some feed-forward 3D reconstruction methods (Wang et al., 2024; 2025a; Zhang et al., 2024a; Smart et al., 2024; Li et al., 2025b; Wang et al., 2025b; Yang et al., 2025a), using vision Transformer (Dosovitskiy et al., 2020) (VIT) to encode and match input images' features and recover their relative 3D poses, depths, semantics (Sun et al., 2025; Xu et al., 2025), and other 3D information (Jiang et al., 2025; Smart et al., 2024). Those methods achieve nearly real-time reconstruction given an image sequence, while maintaining accurate pose/depth estimation, and high-quality novel view synthesis.

3D Foundation Generation Models. A series of 3D foundation models aims to generate high-quality 3D representations with few or a single image(s) as input in seconds. In the early stage, 3D Generation mainly focuses on structure&shape generation (Ren et al., 2024b; Vahdat et al., 2022) or other latent representation (Yang et al., 2024b). Point-E (Nichol et al., 2022) trains a 3D diffusion model, which is used for generating point clouds from text/image prompts. VecSet (Zhang et al., 2023) proposes to encode 3D assets into vector representations, which are further applied in the geometry diffusion models (Chen et al., 2025d; Hunyuan3D et al., 2025; Lai et al., 2025; Zhang et al., 2024c; Li et al., 2024; Xiong et al., 2025). Then, texture diffusion models (Hunyuan3D et al., 2025; Li et al., 2025a) are followed to color the high-quality mesh. TRELLIS (Xiang et al., 2024) and some recent works (Ye et al., 2025; Wu et al., 2025b; Li et al., 2025c; Chen et al., 2025d) encode multiview images into sparse 3D voxel representations and then decode them into high-quality 3D assets. Several methods are proposed to generate dynamic objects (Chen et al., 2025a; Zhang et al., 2025a; Wu et al., 2025c) or extend 3D generation to the part level (Chen et al., 2025b; Dong et al., 2025; Chen et al., 2025c; Yang et al., 2025b).

We observe that most 3D diffusion models split the generation process into two phases – geometry and appearance. Our research aims to bridge the gap between geometry and appearance in 3D generation by introducing a unified latent space while maintaining the strong performance of 3D diffusion models.

3 Preliminary

Recently, TRELLIS (Xiang et al., 2024), a powerful 3D generation framework, has enabled generating high-quality 3D assets in seconds. This is achieved by proposing sparse structured latents (SLATs) $\mathbf{z}_{\text{slat}}$ to represent the 3D asset, which can be decoded into different 3D representations.

Sparse Structured Latent Representation. SLAT is defined as a series of latents located at activated surface voxels of the 3D asset, which can be formulated as $\mathbf{z}_{\text{slat}} = \{z_i, p_i\}_{i=1}^{L}$, where $z_i \in R^c$ is a c-dimensional latent at the voxel position $p_i \in R^3$, $i = \{1, 2, ...L\}$, N denotes the grid resolution

and $L << N^3$. The coordinates $\{p_i\}$, representing coarse geometry, are computed by voxelizing the 3D asset. The latents $\{z_i\}$, representing appearance and detailed geometry[1], are obtained by aggregating and encoding visual features $\mathbf{f} = \{f_i, p_i\}_{i=1}^{L}$, extracted by a vision encoder (Oquab et al., 2023) from multiple views of the asset. To learn geometry and appearance respectively, TRELLIS constructs two separate VAE models, *i.e.*, geometry VAE $\{\mathcal{E}_{\text{geo}}, \mathcal{D}_{\text{geo}}\}$ and appearance VAE $\{\mathcal{E}_{\text{app}}, \mathcal{D}_{\text{app}}\}$.

Specifically, the encoder of the geometry VAE transforms activated voxels $\mathbf{p} = \{p_i\}$ to geometry latents $\mathbf{z}_{\text{geo}} \in R^{\frac{N}{s} \times \frac{N}{s} \times \frac{N}{s} \times c}$ with a downsampling factor s:

$$\mathbf{z}_{\text{geo}} = \mathcal{E}_{\text{geo}}(\mathbf{p}); \quad \mathbf{p} = \mathcal{D}_{\text{geo}}(\mathbf{z}_{\text{geo}}). \tag{1}$$

The sparse appearance VAEs encodes the sparse 3D features $\mathbf{f}$ into SLATs $\mathbf{z}_{\text{slat}}$, and decodes SLATs into 3D representations $\mathcal{O}$ as:

$$\mathbf{z}_{\text{slat}} = \mathcal{E}_{\text{app}}(\mathbf{f}); \quad \mathcal{O} = \mathcal{D}_{\text{app}}(\mathbf{z}_{\text{slat}}). \tag{2}$$

Note that $\mathcal{E}_{\text{app}}$ only converts $\mathbf{f}$ in the feature dimension. The coordinate information is modeled by $\mathcal{E}_{\text{geo}}$ individually.

Sparse Structured Latent Generation. To generate SLAT $\mathbf{z}_{\text{slat}}$, TRELLIS proposes a two-stage generation pipeline. Given the condition image $\mathbf{I}$, TRELLIS builds a geometry generation flow Transformer $\mathcal{F}_{\text{geo}}$ to synthesize geometry latents $\mathbf{z}_{\text{geo}}$ from the noise ϵ. Then, the activated voxels $\mathbf{p}$ can be decoded by $\mathcal{D}_{\text{geo}}$:

$$\mathcal{F}_{\text{geo}} : (\epsilon, t, \mathbf{I}) \rightarrow \mathbf{z}_{\text{geo}}; \quad \mathbf{p} = \mathcal{D}_{\text{geo}}(\mathbf{z}_{\text{geo}}), \tag{3}$$

where t is the denoising timestep. After that, the appearance noise can be added to the activated voxels $\mathbf{p}$ to get the structured noise $\epsilon_{\text{app}} = \{\epsilon_i, p_i\}$. The sparse appearance flow Transformer is optimized to predict $\mathbf{z}_{\text{slat}}$, and the final 3D representation $\mathcal{O}$ can be computed by the appearance decoder $\mathcal{D}_{\text{app}}$:

$$\mathcal{F}_{\text{app}} : (\epsilon_{\text{app}}, t, \mathbf{I}) \rightarrow \mathbf{z}_{\text{slat}}; \quad \mathcal{O} = \mathcal{D}_{\text{app}}(\mathbf{z}_{\text{slat}}). \tag{4}$$

4 METHOD

4.1 OVERALL FRAMEWORK

Geometry-Appearance Unified Latent Representation. Different from TRELLIS (Xiang et al., 2024), which obtains sparse structured latents $\mathbf{z}_{\text{slat}} = \{z_i, p_i\}_{i=1}^{L}$ in two separate stages, we propose a dense compressed **Lat**ent representation with geometry and appearance **Uni**fied (UniLat) $\mathbf{z}_{\text{uni}} \in R^{M \times M \times M \times d}$ which can be obtained in one single stage, where d is the number of unified latent's channels, $M = \frac{N}{V}$, and V denotes the compression ratio. In the reconstruction stage, we construct a UniLat variational autoencoder (Uni-VAE) $\{\mathcal{E}_{\text{uni}}, \mathcal{D}_{\text{uni},\{\text{gs,mesh}\}}\}$ to encode the 3D assets efficiently. The rich geometry and appearance of an assets $\mathcal{O}$ can be encoded into the UniLat $\mathbf{z}_{\text{uni}}$, which can be further decoded into 3D representations via decoder $\mathcal{D}_{\text{uni}}$ as:

$$\mathbf{z}_{\text{uni}} \leftarrow \mathcal{E}_{\text{uni}}(\mathcal{O}); \quad \mathcal{O} = \mathcal{D}_{\text{uni}}(\mathbf{z}_{\text{uni}}). \tag{5}$$

The unified decoder $\mathcal{D}_{\text{uni}}$ is composed of a upsampling block $\mathcal{D}_{\text{up}}$ and 3D representation decoders $\mathcal{D}_{\text{gs,mesh}}$. For more details, please refer to Sec. 4.2.2.

Geometry–Appearance Unified Latent Generation. With geometry and appearance already fused in our UniLat representation $\mathbf{z}_{\text{uni}}$, the generation process becomes naturally streamlined. A unified generative model $\mathcal{F}_{\text{uni}}$ is employed to directly denoise compact noises ϵ into UniLat $\mathbf{z}_{\text{uni}}$, which can then be decoded by $\mathcal{D}_{\text{uni}}$ into the desired 3D representation:

$$\mathcal{F}_{\text{uni}} : (\epsilon, t, \mathbf{I}) \rightarrow \mathbf{z}_{\text{uni}}; \quad \mathcal{O} = \mathcal{D}_{\text{uni}}(\mathbf{z}_{\text{uni}}). \tag{6}$$

[1]Some detailed geometry properties will be decoded from latents $\{z_i\}$, *e.g.* 3D Gaussian positions and mesh vertices. This will be denoted as 'appearance' for short in the following content.

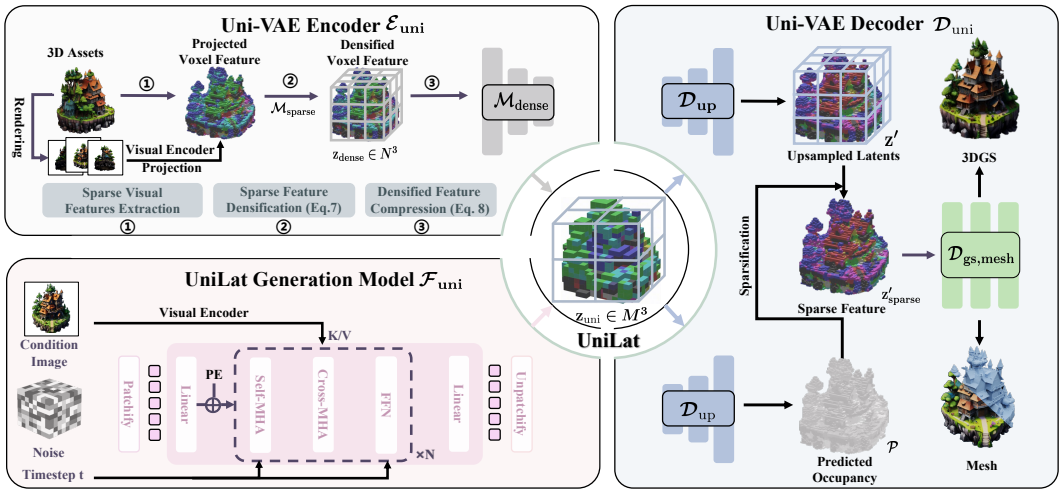

Figure 3: Illustration of the UniLat3D framework. In the reconstruction stage, the encoder of Uni-VAE $\mathcal{E}_{\text{uni}}$ converts the 3D asset $\mathcal{O}$ to the unified latent – UniLat $\mathbf{z}_{\text{uni}}$, which can be directly denoised from noise ϵ by a single flow model $\mathcal{F}_{\text{uni}}$ in the generation stage. The obtained UniLat can be transformed into target 3D representations by the decoder $\mathcal{D}_{\text{uni}}$.

4.2 UniLat Variational Autoencoder

4.2.1 Encoder

The encoder of Uni-VAE $\mathcal{E}_{\text{uni}}$ includes two main stages: **Sparse Feature Densification** and **Densified Feature Compression**, and two modules named sparse appearance feature module $\mathcal{M}_{\text{sparse}}$, dense feature compression module $\mathcal{M}_{\text{dense}}$. The key point is the computational budget. We want to get the UniLat $\mathbf{z}_{\text{uni}}$ and begin with the high dimensional voxelized 3D features $\mathbf{f} = \{f_i, p_i\}$. Firstly, we follow TRELLIS (Xiang et al., 2024) to convert 3D object $\mathcal{O}$ to sparse visual features $\mathbf{f} = \{f_i, p_i\}$ and employ sparse appearance feature module $\mathcal{M}_{\text{sparse}}$ to get $\mathbf{z}_{\text{sparse}}$ by $\mathbf{z}_{\text{sparse}} = \mathcal{M}_{\text{sparse}}(\mathbf{f})$. Then, we introduce the **Sparse Feature Densification** process to fill the empty space in the sparse latents and get $\mathbf{z}_{\text{dense}}$. As computation on $\mathbf{z}_{\text{dense}}$ is expensive, we perform **Densified Feature Compression** phase encodes the processed features into lower-resolution compact latents, *i.e.* UniLat $\mathbf{z}_{\text{uni}}$. Finally, the UniLat decoder $\mathcal{D}_{\text{uni}}$ upsamples the compressed unified latents $\mathbf{z}_{\text{uni}}$ back onto high-resolution 3D representations, supporting both 3D Gaussian and mesh outputs.

Sparse Feature Densification. Yet we have only obtained the sparse feature $\mathbf{z}_{\text{sparse}}$ with appearance encoded, where the geometry is given by indicating which location is empty. To merge both geometry and appearance information into unified latents $\mathbf{z}_{\text{uni}}$, the structured appearance latents $\mathbf{z}_{\text{sparse}} = \{(z_{\text{sparse,i}}, p_i)\}_{i=1}^{L}$ would be converted to dense features $\mathbf{z}_{\text{dense}}$. All the empty space is assigned with zero features $\{0, p_j\}_{j \neq i}^{N^3 - L}$. Then, the sparse structured uni-latents could be transformed to dense uni-latents:

$$\mathbf{z}_{\text{dense}} : \{\mathbf{z}_{\text{dense}}[p_i] = \mathbf{z}_{\text{sparse,i}}; \mathbf{z}_{\text{dense}}[p_j] = 0\} \tag{7}$$

Here, z_{dense} is a compact dense latent that includes the whole space information.

Densified Feature Compression. Then we use $\mathcal{M}_{\text{dense}}$ to encode both the geometry and appearance features. Similar to 2D/2.5D diffusion models, $\mathcal{M}_{\text{dense}}$ downsample the $\mathbf{z}_{\text{dense}} \in N^3$ to **UniLats** $\mathbf{z}_{\text{uni}} \in M^3$ with downsampling factor s:

$$\mathbf{z}_{\text{uni}} = \mathcal{M}_{\text{dense}}(\mathbf{z}_{\text{dense}}). \tag{8}$$

The geometry and appearance features are further fused by the downsampling encoding process, ensuring rich information in the UniLat $\mathbf{z}_{\text{uni}}$ at the low resolution.

4.2.2 DECODER

Uni-Decoder $\mathcal{D}_{\text{uni}}$ includes two modules: upsampling block $\mathcal{D}_{\text{up}}$ and 3D representation decoders $\mathcal{D}_{\text{gs,mesh}}$. The high-resolution dense coordinate and features $\mathbf{z}'_{\text{dense}} \in R^{N^3 \times (C+1)}$ are computed by $\mathcal{D}_{\text{up}}$, then the pruning process is performed on the dense features $\mathbf{z}'_{\text{dense}}$ to obtain sparse features $\mathbf{z}'_{\text{sparse}}$. Finally, representation decoders $\mathcal{O}_{\text{gs,mesh}}$ output the final 3D representations.

Latent Upsampling and Sparsification. Given a compact but low-resolution UniLat $\mathbf{z}_{\text{uni}}$, the core challenge is to reconstruct high-quality 3D assets in a detailed manner. To address this, we introduce an upsampling block that lifts $\mathbf{z}_{\text{uni}}$ to higher-resolution latents. Leveraging our geometry–appearance unified representation, we can simultaneously predict voxel occupancy, which guides a pruning step to remove redundant regions among the upsampled latents. This yields a sparse set of high-resolution latents that retain both efficiency and fidelity.

Given UniLat $\mathbf{z}_{\text{uni}} \in R^{M^3 \times d}$, our proposed upsampling blocks $\mathcal{D}_{\text{up}}$ compute the appearance and geometry features at resolution N as :

$$\mathbf{z}'_{\text{dense}} = \mathcal{D}_{\text{up}}(\mathbf{z}_{\text{uni}}). \tag{9}$$

Note that both $\mathbf{z}'_{\text{dense}} = \{\mathcal{P} \in R^{N^3 \times 1}, \mathbf{z}' \in N^3 \times c\}$ are high-resolution dense features. Note that directly performing computation on $\mathbf{z}_{\text{dense}}$ is expensive, so we propose to prune the low-importance area to enhance efficiency. The sparse features $\mathbf{z}_{\text{sparse}}$ are filtered with a signed function:

$$\mathbf{z}'_{\text{sparse}} : \{\mathbf{z}'_i, p_i \mid \mathcal{P}[p_i] > 0\}, \tag{10}$$

3D Representation Decoders. Two 3D representation decoders are designed to transform the pruned latents into renderable 3D outputs, *i.e.*, 3D Gaussians and meshes. Both decoders share a backbone of sparse Transformer blocks, similar to TRELLIS, but differ in their task-specific output heads. For 3D Gaussians, the decoder $\mathcal{D}_{\text{gs}}$ maps the latent $\mathbf{z}_{\text{uni}}$ to attributes of 3D Gaussian primitives $\mathcal{O}_{\text{gs}}$ using sparse Transformer blocks and 3D linear projection layers. An additional occupancy head is employed to predicts voxel occupancy, enabling direct supervision of the reconstructed geometry.

For meshes, the decoder $\mathcal{D}_{\text{mesh}}$ progressively upsamples latent features through sparse Transformer and 3D CNN upsampling blocks, followed by a 3D linear output layer that predicts SDF values, voxel-corner deformations, and interpolation weights. The final mesh is extracted with the efficient SparseFlex (He et al., 2025) based on these predicted parameters. To enable multi-scale geometry supervision, occupancy heads are attached at each upsampled resolution. To scale $\mathcal{D}_{\text{mesh}}$ to higher resolutions, we adopt a pruning strategy that removes voxels entirely outside or inside object boundaries, thereby reducing computational overhead. We further introduce a detail augmentation strategy, where depth and normal maps are rendered from zoomed-in camera views with a differentiable rasterizer, enabling the decoder to learn fine-grained surface details from localized partial observations. With these techniques, UniLat3D produces meshes at a resolution of 512^3, doubling the resolution achieved by TRELLIS (256^3).

4.3 UNILAT GENERATION MODEL

With the UniLat VAE model, we construct a generation model $\mathcal{F}_{\text{uni}}$ based on rectified flow matching to denoise compact noise ϵ into condition-followed UniLats $\mathbf{z}_{\text{uni}}$. A single flow Transformer model $\mathcal{F}_{\text{uni}}$ with full attention layers is built to predict the velocity at timestamp t under the noise level as $v = \mathcal{F}_{\text{uni}}(\mathbf{x}_{\text{uni}}, t, I)$ and $\mathbf{x}_{\text{uni}}$ denotes the denoised noise ϵ and timestamp t. The whole flow Transformer optimization process follows the diffusion guidance given condition $\mathbf{I}$ with its condition encoder. The latent features with both geometry and appearance information are denoised. The obtained UniLat $\mathbf{z}_{\text{uni}}$ can be directly fed into the representation decoder $\mathcal{D}_{\text{uni}}$ to predict the final 3D representation $\mathcal{O}$.

4.4 OPTIMIZATION

Uni-VAE. We use both geometry and appearance supervision to train the $\mathcal{D}_{\text{uni}}$. Following TRELLIS (Xiang et al., 2024), we joint optimize $\mathcal{E}_{\text{uni}}$ and $\mathcal{D}_{\text{gs}}$ with the following loss:

$$\mathcal{L} = \lambda_{\text{l1}}\mathcal{L}_{\text{l1}} + \lambda_{\text{lpips}}\mathcal{L}_{\text{lpips}} + \lambda_{\text{ssim}}\mathcal{L}_{\text{ssim}} + \lambda_{\text{kl}}\mathcal{L}_{\text{kl}} + \lambda_{\text{dice}}\mathcal{L}_{\text{dice}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}. \tag{11}$$

$\mathcal{L}_{\text{l1}}$ denotes the L1 color loss, and $\mathcal{L}_{\text{lpips}}$ and $\mathcal{L}_{\text{ssim}}$ stand for inception-based losses. $\mathcal{L}_{\text{kl}}$ is employed for optimizing $\mathcal{E}_{\text{uni}}$. $\mathcal{L}_{\text{dice}}$ and $\mathcal{L}_{\text{reg}}$ are used to supervise geometry and decoded representations.

Rectified Flow Models. After trained Uni-VAE, all the UniLat $\mathbf{z}_{\text{uni}}$ are predicted by Uni-Encoder $\mathcal{E}_{\text{uni}}$. For optimizing the rectified flow Transformer, we mainly follow the CFM Loss. Given encoded latents $\mathbf{x}_{\text{uni}}$ and noise ϵ, we minimize the objective function $\mathcal{L}_{\text{CFM}}$ (Lipman et al., 2022) as:

$$L_{\text{CFM}}(\theta) = \mathbb{E}_{t,x_0,\epsilon}\|v(\mathbf{x}_{\text{uni}}, t) - (\epsilon - \mathbf{z}_{\text{uni}})\|_2^2. \tag{12}$$

5 Experiments

5.1 Implementation Details

Our framework is implemented in PyTorch (Paszke et al., 2019) and built upon the open-source project TRELLIS (Xiang et al., 2024). FlashAttention-3(Shah et al., 2024) is employed to accelerate Transformer training, yielding a $1.5\times$ speedup. Both VAE and flow models are trained on 64 GPUs within two weeks. More implementation details are provided in the appendix.

5.2 Experimental Setup

Training Datasets. UniLat3D is trained exclusively on publicly available datasets. Following the data preparation pipeline of TRELLIS (Xiang et al., 2024), we curate and process approximately 450k high-quality 3D assets from Objaverse (XL) (Deitke et al., 2023), ABO (Collins et al., 2022), 3D-FUTURE (Fu et al., 2021), and HSSD (Khanna* et al., 2023). To enable occupancy supervision at multiple scales, we perform voxelization at each resolution. Additional details on data preprocessing can be found in (Xiang et al., 2024).

Evaluation Datasets. The evaluation is performed on two datasets. One is the whole Toys4K (Stojanov et al., 2021) dataset, including 3218 high-quality 3D assets, which is also used in the previous method (Xiang et al., 2024). However, we observe that many samples of Toys4K tend to have simple geometry or appearance details. We construct a more complex dataset for comprehensive evaluation, including 500 high-quality assets collected from the Sketchfab platform and 500 assets sampled from Toys4K. Condition images for qualitative comparisons and user studies are collected from (Chen et al., 2025d; Wu et al., 2025b) or generated via VLMs.

Evaluation Setups. For VAE reconstruction evaluation, we use the PSNR, SSIM, and LPIPS metrics. For appearance generation quality, we compute the CLIP (Radford et al., 2021) score – similarity between rendered images and condition images, and FD (Fréchet distance) (Heusel et al., 2017) measured by DINOv2 (Oquab et al., 2023) on 4 views of each generated asset and ground truth images. We evaluate and compare our method with recent SOTA 3D generation models, *i.e.* Hunyuan3D-2.1 (Hunyuan3D et al., 2025), TRELLIS (Xiang et al., 2024), Step1X-3D (Li et al., 2025a), TripoSR (Tochilkin et al., 2024) for image-conditioned generation, and Stable3DGen (Ye et al., 2025) and Direct3D-S2 (Wu et al., 2025b) for geometry generation quality comparison. We report Uni3D (Zhou et al., 2023) and ULIP (Xue et al., 2023) metrics for mesh geometry quality. The Blender rendering pipeline is adopted for all generated mesh assets.

5.3 Results

We provide qualitative comparisons in Fig. 4, where our method achieves competitive generation quality and demonstrates stronger alignment with the conditional image, benefiting from the unified representation. Note that Hunyuan3D-2.1 (Hunyuan3D et al., 2025), Step1X-3D (Li et al., 2025a), and TripoSR (Tochilkin et al., 2024) only provide mesh-based results. Importantly, Ours, TripoSR, TRELLIS, and Direct3D-S2 are trained exclusively on publicly available datasets, while other methods leverage additional private data.

Quantitative evaluations on Toys4k (Stojanov et al., 2021) are reported in Table 1. Additional results on our self-collected complex set are provided in the appendix. Compared with other two-stage methods, UniLat3D achieves leading appearance performance, reaching 47.68 in $\text{FD}_{\text{DINOv2}}$. The CLIP score of 90.87 further demonstrates the effectiveness of UniLat3D in aligning images and 3D assets. In terms of geometry synthesis, our mesh version also achieves competitive results in ULIP (Xue et al., 2023), with a score of 42.69.

Figure 4: Qualitative comparisons with other methods. Thanks to our unified representation, UniLat3D achieves superior performance and better correspondence with input images.

Figure 5: 3D mesh assets generated by our UniLat3D.

Beyond accuracy, UniLat3D also demonstrates notable efficiency: 3D Gaussian generation is completed within 8 seconds on a single A100 GPU and can be further reduced to 3 seconds with FlashAttention-3 (Shah et al., 2024). Mesh generation requires 36 seconds, primarily due to the higher resolution with more vertices and longer post-processing compared to TRELLIS, but remains competitive considering the improved output quality.

Besides, we conducted a user study with 22 participants over 3D assets generated from 23 image prompts. Four models with both geometry and

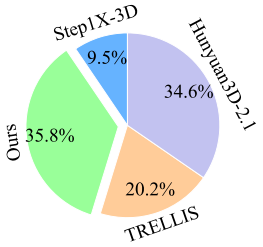

Figure 6: User study on different models.

Table 1: Comparisons on the Toys4K dataset. "Rep." denotes the output representation type.

Model	Rep.	#Param.	Time	CLIP↑	FD_{DINOv2}↓	ULIP↑	Uni3D↑
TripoSR	Mesh	0.4B	13s	88.76	279.06	35.30	30.98
TRELLIS	Mesh	1.31B	21s	87.81	79.52	42.51	37.67
TRELLIS	3DGS	1.31B	5s	90.70	52.54	–	–
Stable3DGen†⋆	Mesh	2.63B	4s	–	–	40.33	35.98
Step1X-3D†	Mesh	4.8B	152s	85.85	146.08	41.37	36.51
Direct3D-S2⋆	Mesh	2.1B	185s	–	–	41.51	36.64
Hunyuan3D-2.1†	Mesh	5.3B	90s	88.44	74.16	42.67	**37.74**
Ours	Mesh	1.58B	36s	87.93	71.81	**42.69**	37.62
Ours	3DGS	1.55B	8s	**90.87**	**47.68**	–	–

† Using proprietary or non-public training data.
⋆ Only generating geometry without appearance.

appearance generation are involved. For each prompt, participants judged generated assets by both image alignment and object quality, and chose the overall best case. As shown in Fig. 6, UniLat3D received over 35% of the votes, outperforming Huanyuan3D-2.1 and other models.

5.4 Ablation Study

Resolution of Latents. We explore the latent space of reconstruction quality in Uni-VAE. We train Uni-VAE at different latent resolutions, including 8^3, 16^3, and 32^3. As shown in Table 2, higher UniLat resolutions lead to better reconstruction results. Note that our Uni-VAE achieves similar or even better reconstruction performance than TRELLIS with smaller resolutions. In our experiments, when training the flow Transformer at a higher resolution of 32, the computational cost increases evidently. We would explore more efficient approaches on flow Transformers for higher resolutions in future works, *e.g.*, block-wise computation and lightweight attention.

Table 2: VAE reconstruction results with latents of different resolutions.

Model	Res.	PSNR↑	SSIM↑	LPIPS↓
TRELLIS (Mesh)	64^3	31.91	97.44	0.0328
Ours (Mesh)	16^3	32.35	98.03	0.0305
TRELLIS (GS)	64^3	34.74	98.52	0.0146
Ours (GS)	8^3	33.51	98.13	0.0200
Ours (GS)	16^3	34.80	98.49	0.0158
Ours (GS)	32^3	**34.92**	**98.53**	**0.0145**

Another ablation study about the condition visual encoder is included in the appendix.

6 Discussion & Conclusion

We propose a novel 3D generation framework – UniLat3D to achieve high-quality 3D asset generation in seconds with a single-stage flow model. Apart from that the proposed method unifies geometry and appearance in a single, concise framework, it achieves quite competitive performance compared with popular two-stage methods. We expect our exploration to provide a more convenient and extensible choice to the 3D generation field, *e.g.*, further unifying object and scene generation with the compact unified representation, extending UniLat to 4D representations, and integrating UniLat into large multimodal models *etc*.

However, the UniLat3D model implemented in this paper is still a preliminary exploration. The training data we used just follows TRELLIS, totally from public datasets. Injecting more high-quality data for training will undoubtedly improve the performance and may further scale up the model. Exploring more efficient designs on the flow model would adapt to higher resolutions of latents, leading to more detailed generation results.

SUBMISSION STATEMENT

ETHICS STATEMENT

Our UniLat3D only adopts publicly available datasets for training. The generative 3D method may raise potential ethical concerns. Users may use images to create harmful or misleading 3D assets. We encourage the responsible use of our models within legal and ethical boundaries.

REPRODUCIBILITY STATEMENT

To ensure reproducibility, we introduce implementation details in Sec. A.2 and Sec. A.4.

REFERENCES

Blender Foundation. Blender: Open source 3d creation suite. `https://www.blender.org`, 2025. Accessed: 2025-09-24.

Jianqi Chen, Biao Zhang, Xiangjun Tang, and Peter Wonka. V2m4: 4d mesh animation reconstruction from a single monocular video. *arXiv preprint arXiv:2503.09631*, 2025a.

Minghao Chen, Roman Shapovalov, Iro Laina, Tom Monnier, Jianyuan Wang, David Novotny, and Andrea Vedaldi. Partgen: Part-level 3d generation and reconstruction with multi-view diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5881–5892, 2025b.

Minghao Chen, Jianyuan Wang, Roman Shapovalov, Tom Monnier, Hyunyoung Jung, Dilin Wang, Rakesh Ranjan, Iro Laina, and Andrea Vedaldi. Autopartgen: Autogressive 3d part generation and discovery. *arXiv preprint arXiv:2507.13346*, 2025c.

Yiwen Chen, Zhihao Li, Yikai Wang, Hu Zhang, Qin Li, Chi Zhang, and Guosheng Lin. Ultra3d: Efficient and high-fidelity 3d generation with part attention. *arXiv preprint arXiv:2507.17745*, 2025d.

Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. Abo: Dataset and benchmarks for real-world 3d object understanding. *CVPR*, 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 *CVPR*, pp. 13142–13153, 2023.

Shaocong Dong, Lihe Ding, Xiao Chen, Yaokun Li, Yuxin Wang, Yucheng Wang, Qi Wang, Jaehyeok Kim, Chenjian Gao, Zhanpeng Huang, et al. From one to more: Contextual part latents for 3d generation. *arXiv preprint arXiv:2507.08772*, 2025.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129 (12):3313–3337, 2021.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

Xianglong He, Zi-Xin Zou, Chia-Hao Chen, Yuan-Chen Guo, Ding Liang, Chun Yuan, Wanli Ouyang, Yan-Pei Cao, and Yangguang Li. Sparseflex: High-resolution and arbitrary-topology 3d shape modeling. *arXiv preprint arXiv:2503.21732*, 2025.

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

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

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 preprint arXiv:2311.04400*, 2023.

Zehuan Huang, Yuan-Chen Guo, Haoran Wang, Ran Yi, Lizhuang Ma, Yan-Pei Cao, and Lu Sheng. Mv-adapter: Multi-view consistent image generation made easy. *arXiv preprint arXiv:2412.03632*, 2024.

Team Hunyuan3D, Shuhui Yang, Mingxin Yang, Yifei Feng, Xin Huang, Sheng Zhang, Zebin He, Di Luo, Haolin Liu, Yunfei Zhao, et al. Hunyuan3d 2.1: From images to high-fidelity 3d assets with production-ready pbr material. *arXiv preprint arXiv:2506.15442*, 2025.

Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *CVPR*, pp. 867–876, 2022.

Haoyi Jiang, Liu Liu, Tianheng Cheng, Xinjie Wang, Tianwei Lin, Zhizhong Su, Wenyu Liu, and Xinggang Wang. Gausstr: Foundation model-aligned gaussian transformer for self-supervised 3d spatial understanding. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11960–11970, 2025.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. URL `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`.

Mukul Khanna*, Yongsen Mao*, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat Synthetic Scenes Dataset (HSSD-200): An Analysis of 3D Scene Scale and Realism Tradeoffs for ObjectGoal Navigation. *arXiv preprint*, 2023.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Zeqiang Lai, Yunfei Zhao, Haolin Liu, Zibo Zhao, Qingxiang Lin, Huiwen Shi, Xianghui Yang, Mingxin Yang, Shuhui Yang, Yifei Feng, et al. Hunyuan3d 2.5: Towards high-fidelity 3d assets generation with ultimate details. *arXiv preprint arXiv:2506.16504*, 2025.

Weiyu Li, Jiarui Liu, Rui Chen, Yixun Liang, Xuelin Chen, Ping Tan, and Xiaoxiao Long. Craftsman: High-fidelity mesh generation with 3d native generation and interactive geometry refiner. *arXiv preprint arXiv:2405.14979*, 2024.

Weiyu Li, Xuanyang Zhang, Zheng Sun, Di Qi, Hao Li, Wei Cheng, Weiwei Cai, Shihao Wu, Jiarui Liu, Zihao Wang, et al. Step1x-3d: Towards high-fidelity and controllable generation of textured 3d assets. *arXiv preprint arXiv:2505.07747*, 2025a.

Zhengqi Li, Richard Tucker, Forrester Cole, Qianqian Wang, Linyi Jin, Vickie Ye, Angjoo Kanazawa, Aleksander Holynski, and Noah Snavely. Megasam: Accurate, fast and robust structure and motion from casual dynamic videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10486–10496, 2025b.

Zhihao Li, Yufei Wang, Heliang Zheng, Yihao Luo, and Bihan Wen. Sparc3d: Sparse representation and construction for high-resolution 3d shapes modeling. *arXiv preprint arXiv:2505.14521*, 2025c.

Hanwen Liang, Yuyang Yin, Dejia Xu, Hanxue Liang, Zhangyang Wang, Konstantinos N Plataniotis, Yao Zhao, and Yunchao Wei. Diffusion4d: Fast spatial-temporal consistent 4d generation via video diffusion models. *arXiv preprint arXiv:2405.16645*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Fangfu Liu, Diankun Wu, Yi Wei, Yongming Rao, and Yueqi Duan. Sherpa3d: Boosting high-fidelity text-to-3d generation via coarse 3d prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20763–20774, 2024.

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*, 2023a.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. *arXiv preprint arXiv:2303.11328*, 2023b.

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.

Ziqiao Ma, Xuweiyi Chen, Shoubin Yu, Sai Bi, Kai Zhang, Chen Ziwen, Sihan Xu, Jianing Yang, Zexiang Xu, Kalyan Sunkavalli, et al. 4d-lrm: Large space-time reconstruction model from and to any view at any time. *arXiv preprint arXiv:2506.18890*, 2025.

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, pp. 405–421, 2020.

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.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

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

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*, pp. 8748–8763. PMLR, 2021.

Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

Jiawei Ren, Cheng Xie, Ashkan Mirzaei, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung Wook Kim, Huan Ling, et al. L4gm: Large 4d gaussian reconstruction model. *Advances in Neural Information Processing Systems*, 37:56828–56858, 2024a.

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*, pp. 4209–4219, 2024b.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6121–6132, 2025.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *Advances in Neural Information Processing Systems*, 37:68658–68685, 2024.

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

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Brandon Smart, Chuanxia Zheng, Iro Laina, and Victor Adrian Prisacariu. Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs. *arXiv preprint arXiv:2408.13912*, 2024.

Stefan Stojanov, Anh Thai, and James M Rehg. Using shape to categorize: Low-shot learning with an explicit shape bias. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1798–1808, 2021.

Xiangyu Sun, Liu Liu, Seungtae Nam, Gyeongjin Kang, Wei Sui, Zhizhong Su, Wenyu Liu, Xinggang Wang, Eunbyung Park, et al. Uni3r: Unified 3d reconstruction and semantic understanding via generalizable gaussian splatting from unposed multi-view images. *arXiv preprint arXiv:2508.03643*, 2025.

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.

Dmitry Tochilkin, David Pankratz, Zexiang Liu, Zixuan Huang, Adam Letts, Yangguang Li, Ding Liang, Christian Laforte, Varun Jampani, and Yan-Pei Cao. Triposr: Fast 3d object reconstruction from a single image. *arXiv preprint arXiv:2403.02151*, 2024.

Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. Lion: Latent point diffusion models for 3d shape generation. *NeurIPS*, 35:10021–10039, 2022.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025a.

Qianqian Wang, Yifei Zhang, Aleksander Holynski, Alexei A Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 10510–10522, 2025b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20697–20709, 2024.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *arXiv preprint arXiv:2305.16213*, 2023.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20310–20320, 2024a.

Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 26057–26068, 2025a.

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.

Shuang Wu, Youtian Lin, Feihu Zhang, Yifei Zeng, Yikang Yang, Yajie Bao, Jiachen Qian, Siyu Zhu, Philip Torr, Xun Cao, and Yao Yao. Direct3d-s2: Gigascale 3d generation made easy with spatial sparse attention. *arXiv preprint arXiv:2505.17412*, 2025b.

Zijie Wu, Chaohui Yu, Fan Wang, and Xiang Bai. Animateanymesh: A feed-forward 4d foundation model for text-driven universal mesh animation. *arXiv preprint arXiv:2506.09982*, 2025c.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. *arXiv preprint arXiv:2412.01506*, 2024.

Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pp. 399–417. Springer, 2024.

Bojun Xiong, Si-Tong Wei, Xin-Yang Zheng, Yan-Pei Cao, Zhouhui Lian, and Peng-Shuai Wang. Octfusion: Octree-based diffusion models for 3d shape generation. In *Computer Graphics Forum*, volume 44, pp. e70198. Wiley Online Library, 2025.

Yueming Xu, Jiahui Zhang, Ze Huang, Yurui Chen, Yanpeng Zhou, Zhenyu Chen, Yu-Jie Yuan, Pengxiang Xia, Guowei Huang, Xinyue Cai, et al. Uniugg: Unified 3d understanding and generation via geometric-semantic encoding. *arXiv preprint arXiv:2508.11952*, 2025.

Le Xue, Mingfei Gao, Chen Xing, Roberto Martín-Martín, Jiajun Wu, Caiming Xiong, Ran Xu, Juan Carlos Niebles, and Silvio Savarese. Ulip: Learning a unified representation of language, images, and point clouds for 3d understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1179–1189, 2023.

Chen Yang, Sikuang Li, Jiemin Fang, Ruofan Liang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Gaussianobject: High-quality 3d object reconstruction from four views with gaussian splatting. *ACM Transactions on Graphics (TOG)*, 43(6):1–13, 2024a.

Haitao Yang, Yuan Dong, Hanwen Jiang, Dejia Xu, Georgios Pavlakos, and Qixing Huang. Atlas gaussians diffusion for 3d generation. *arXiv preprint arXiv:2408.13055*, 2024b.

Jianing Yang, Alexander Sax, Kevin J Liang, Mikael Henaff, Hao Tang, Ang Cao, Joyce Chai, Franziska Meier, and Matt Feiszli. Fast3r: Towards 3d reconstruction of 1000+ images in one forward pass. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 21924–21935, 2025a.

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 preprint arXiv:2411.02293*, 2024c.

Yunhan Yang, Yufan Zhou, Yuan-Chen Guo, Zi-Xin Zou, Yukun Huang, Ying-Tian Liu, Hao Xu, Ding Liang, Yan-Pei Cao, and Xihui Liu. Omnipart: Part-aware 3d generation with semantic decoupling and structural cohesion. *arXiv preprint arXiv:2507.06165*, 2025b.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024d.

Chongjie Ye, Yushuang Wu, Ziteng Lu, Jiahao Chang, Xiaoyang Guo, Jiaqing Zhou, Hao Zhao, and Xiaoguang Han. Hi3dgen: High-fidelity 3d geometry generation from images via normal bridging. *arXiv preprint arXiv:2503.22236*, 3:2, 2025.

Taoran Yi, Jiemin Fang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. *arXiv preprint arXiv:2310.08529*, 2023.

Taoran Yi, Jiemin Fang, Zanwei Zhou, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Xinggang Wang, and Qi Tian. Gaussiandreamerpro: Text to manipulable 3d gaussians with highly enhanced quality. *arXiv preprint arXiv:2406.18462*, 2024.

Yuyang Yin, Dejia Xu, Zhangyang Wang, Yao Zhao, and Yunchao Wei. 4dgen: Grounded 4d content generation with spatial-temporal consistency. *arXiv preprint arXiv:2312.17225*, 2023.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024.

Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions On Graphics (TOG)*, 42(4):1–16, 2023.

Bowen Zhang, Sicheng Xu, Chuxin Wang, Jiaolong Yang, Feng Zhao, Dong Chen, and Baining Guo. Gaussian variation field diffusion for high-fidelity video-to-4d synthesis. *arXiv preprint arXiv:2507.23785*, 2025a.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. Monst3r: A simple approach for estimating geometry in the presence of motion. *arXiv preprint arXiv:2410.03825*, 2024a.

Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. *European Conference on Computer Vision*, 2024b.

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, 2024c.

Shuai Zhang, Guanjun Wu, Zhoufeng Xie, Xinggang Wang, Bin Feng, and Wenyu Liu. Dynamic 2d gaussians: Geometrically accurate radiance fields for dynamic objects. *arXiv preprint arXiv:2409.14072*, 2024d.

Shuai Zhang, Huangxuan Zhao, Zhenghong Zhou, Guanjun Wu, Chuansheng Zheng, Xinggang Wang, and Wenyu Liu. Togs: Gaussian splatting with temporal opacity offset for real-time 4d dsa rendering. *IEEE Journal of Biomedical and Health Informatics*, 2025b.

Yuyang Zhao, Chung-Ching Lin, Kevin Lin, Zhiwen Yan, Linjie Li, Zhengyuan Yang, Jianfeng Wang, Gim Hee Lee, and Lijuan Wang. Genxd: Generating any 3d and 4d scenes. *arXiv preprint arXiv:2411.02319*, 2024.

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

Junsheng Zhou, Jinsheng Wang, Baorui Ma, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Uni3d: Exploring unified 3d representation at scale. *arXiv preprint arXiv:2310.06773*, 2023.

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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10324–10335, 2024.

A Appendix

A.1 Use of LLMs

In this paper, no LLM models are directly used in the writing of the main text.

A.2 More Implementation Details

Uni-VAE. To accelerate and stabilize Uni-VAE training, we initialize $\mathcal{E}_{\text{sparse}}$ and $\mathcal{D}_{\text{sparse}}$ with the pretrained weights from TRELLIS. During the first 240k iterations, only $\mathcal{E}_{\text{dense}}$ and $\mathcal{D}_{\text{up}}$ are optimized, after which the entire Uni-VAE is trained end-to-end for an additional 90k iterations following TRELLIS. For the mesh decoder, we freeze $\mathcal{D}_{\text{uni}}$ and train our high-resolution mesh decoder from scratch. Unless otherwise specified, Adam (Kingma & Ba, 2014) is used with a learning rate of 1×10^{-4}.

Mesh Decoder. We decode the unified latent $\mathbf{z}_{\text{uni}}$ into high-resolution meshes using a high-resolution decoder $\mathcal{D}_{\text{uni,mesh}}$. The decoder first upsamples $\mathbf{z}_{\text{uni}}$ and performs sparsification to obtain $\mathbf{z}_{\text{sparse}}$ (as detailed in Sec. 4.2.2), which is then processed by a stack of sparse Transformer blocks. Two 3D convolutional upsampling layers with skip connections to their corresponding pruning modules further refine $\mathbf{z}_{\text{sparse}}$ to a spatial resolution of 256^3.

To enable multi-scale geometry supervision and reduce computation, we attach an occupancy head at each upsampled scale $s \in \{128, 256\}$ and supervise it with voxel-level occupancy targets. The occupancy loss is

$$\mathcal{L}_{\text{occ}} = \sum_{s \in \{128,256\}} \mathcal{L}_{\text{dice}}^{(s)}.$$

The overall training objective for the mesh branch is

$$\mathcal{L}_{\text{mesh}} = \mathcal{L}_{\text{geo}} + \mathcal{L}_{\text{color}} + \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{occ}}, \tag{13}$$

where $\mathcal{L}_{\text{geo}}$, $\mathcal{L}_{\text{color}}$, and $\mathcal{L}_{\text{reg}}$ follow TRELLIS (Xiang et al., 2024).

Given the computational cost at high resolutions, we adopt a two-stage schedule. *Stage-1* trains $\mathcal{D}_{\text{uni,mesh}}$ up to 256^3 with the multi-scale occupancy supervision in Eq. equation 13. *Stage-2* adds an independent $256 \to 512$ upsampling convolutional block with its own pruning head; the newly added block is optimized while keeping the *Stage-1* pathway frozen, using a similar objective as Eq. equation 13.

After training, $\mathcal{D}_{\text{uni,mesh}}$ directly predicts the SparseFlex (He et al., 2025) parameters from $\mathbf{z}_{\text{uni}}$, from which we extract mesh vertices and faces efficiently. Following TRELLIS and prior 3D generation works, we apply lightweight post-processing to remove invisible and degenerate faces and to fill small holes.

UniLat Flow Transformer. For training the rectified flow models, we adopt DINOv3 (Siméoni et al., 2025) as the image encoder and apply classifier-free guidance (Ho & Salimans, 2022) with a drop rate of 0.1. The model is first trained for 500k iterations with a batch size of 256 and a learning rate of 1×10^{-4}, and then fine-tuned for 160k iterations with a batch size of 1024 and a learning rate of 1×10^{-5}.

A.3 Evaluation Details

Blender Rendering Setups. For mesh rendering, we mainly use Blender (Blender Foundation, 2025) as a mesh renderer to render high-quality images. We set FOV=40, render resolution=512, and set normalization to each loaded object.

More Results We provide evaluation metrics on our self-collected complex test dataset in Tab 3. More comparisons with open-sourced models are shown in Fig. 7. Besides, we provide qualitative comparisons with some commercial models in Fig. 8. Our model still shows notable competitiveness.

Table 3: Comparisons on the self-collected complex test dataset. "Rep." denotes the output representation type, and "#Params" denotes the number of model parameters.

Model	Rep.	#Param.	Time	CLIP↑	FD_{DINOv2}↓	ULIP↑	Uni3D↑
TripoSR	Mesh	0.4B	13s	88.00	369.86	33.61	30.44
TRELLIS	Mesh	1.31B	21s	86.40	164.57	41.52	37.30
TRELLIS	3DGS	1.31B	5s	89.67	108.27	-	-
Stable3DGen†⋆	Mesh	2.63B	4s	-	-	39.79	35.93
Step1X-3D†	Mesh	4.8B	152s	84.74	210.49	40.53	36.46
Direct3D-S2⋆	Mesh	2.1B	185s	-	-	40.77	36.47
Hunyuan3D-2.1†	Mesh	5.3B	90s	87.41	150.39	41.70	**37.48**
Ours	Mesh	1.58B	36s	86.44	149.62	**41.71**	37.24
Ours	3DGS	1.55B	8s	**89.83**	**97.22**	-	-

† Using proprietary or non-public training data.
⋆ Only generating geometry without appearance.

Figure 7: Qualitative comparisons with SOTA open-source models.

A.4 Model Architecture

In this section, we mainly provide the model architecture about our Uni-VAE $\{\mathcal{E}_{uni}, \mathcal{D}_{uni}\}$ and UniLat generation model $\mathcal{F}$.

A.4.1 Uni-VAE

For the sparse encoder $\mathcal{M}_{sparse}$, we mainly follow TRELLIS's configurations to build a sparse Transformer. For the dense encoder $\mathcal{M}_{dense}$, a set of conv3D layers is used as the main architecture. The settings of $\mathcal{E}_{sparse}$, $\mathcal{D}_{up}$ are shown in Table 4 and details of $\mathcal{E}_{uni}$ are provided in Table 5.

A.4.2 UniLat Flow Transformer

Structure details about our UniLat flow Transformer $\mathcal{F}_{uni}$ are provided in the Table 6. The main architecture of $\mathcal{F}_{uni}$ is similar to TRELLIS's sparse structure flow Transformer. The input noise ϵ would be flattened to 1D tensors. Positional encoding is applied to a flattened tensor, and it would be fed to Transformer blocks with self&cross-attention layer and modulated by condition signal & timestamps. Finally, the flattened tensor would be unpatchified to 3D results, the shape is the same as ϵ.

Table 4: Model details of Uni-VAE modules $\mathcal{M}_{dense}, \mathcal{D}_{up}$. "Channels" denotes model channels after each up/downsampled convolution layer.

Model	ResBlocks	Channels
$\mathcal{E}_{sparse}$	4	$[32, 128, 512]$
$\mathcal{D}_{up}$	4	$[512, 128, 32]$

Table 5: Model details of Uni-VAE modules $\mathcal{M}_{sparse}, \mathcal{D}_{gs,mesh}$.

Model	Latent Res.	Model Channels	Latent. Channels	Blocks	Attn. Heads	Window Size
$\mathcal{M}_{sparse}, \mathcal{D}_{sparse}$	64	768	8	12	12	8

Table 6: Model details of UniLat3D flow Transformer.

Model	Params	Latent Res.	Latent Channels	Model Channels	Cond. Channels	Blocks	Attn. Heads
$\mathcal{F}_{uni}$	1.30B	16	32	1280	1280	36	32

Table 7: Ablation study on the visual encoder for condition images.

Model	Cond. Encoder	CLIP↑	$\text{FD}_{\text{dinov2}}$↓
Ours	DINOV2	**90.83**	52.58
Ours	DINOV3	90.60	**49.90**

A.5 More Ablation Studies

Visual Encoder of Condition Images Recently, DINOv3 (Siméoni et al., 2025) emerges as a strong visual encoder model that could extract high-quality details from the image. We compare the performance between DINOv2 and DINOv3 for encoding condition images. Flow models with different visual encoders are trained for 500 iterations and tested on Toys4K. In our experiments, the flow Transformer with the DINOv3 encoder shows better quality on complex object generation, which leads to a better $\text{FD}_{\text{dinov2}}$ result as shown in Table 7.

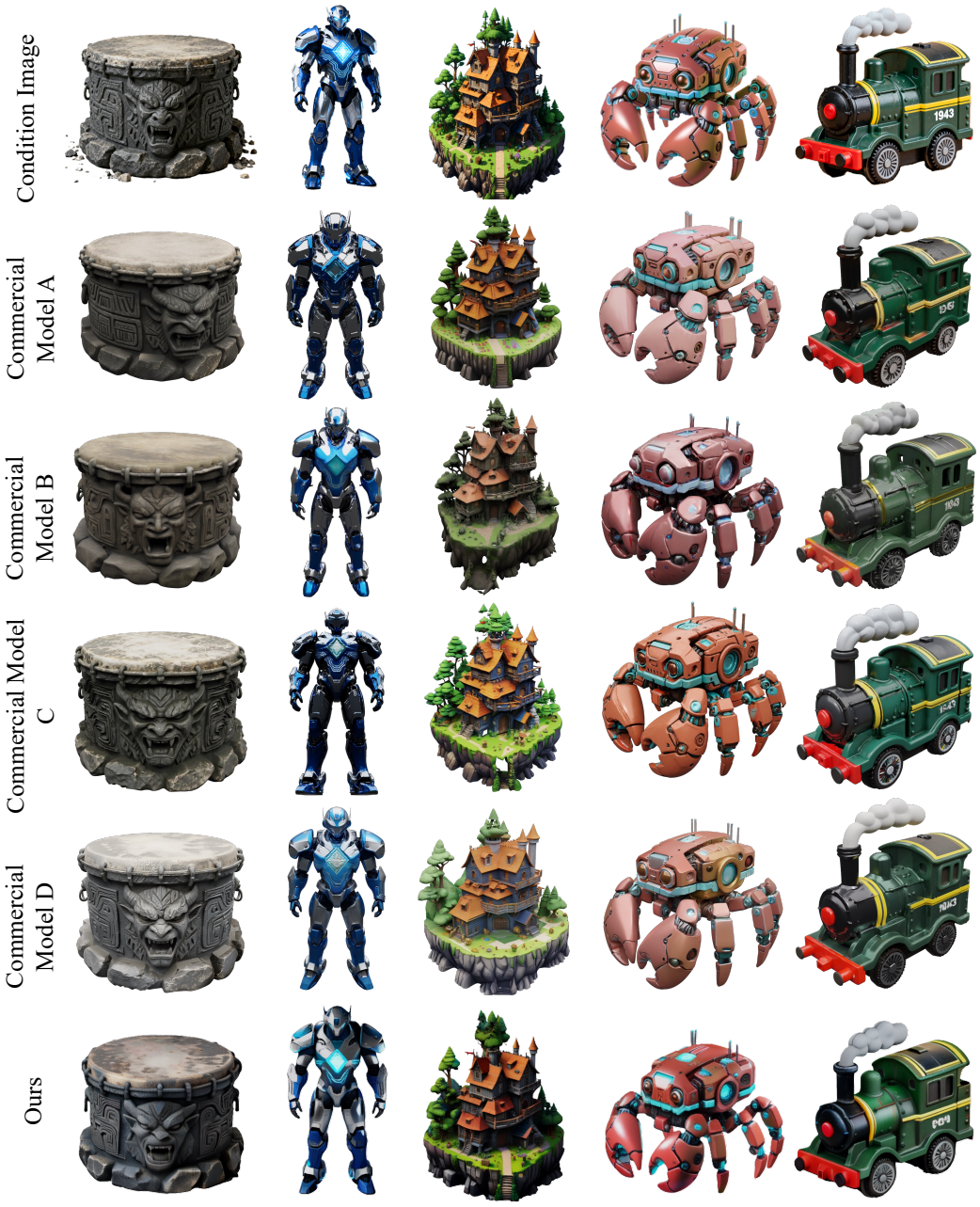

Figure 8: Qualitative comparisons with commercial models. Our UniLat3D shows competitive performance even with only publicly available training data.