Structured 3D Latents for Scalable and Versatile 3D Generation

Jianfeng Xiang[1,3] Zelong Lv[2,3] Sicheng Xu[3] Yu Deng[3] Ruicheng Wang[2,3]
Bowen Zhang[2,3] Dong Chen[3] Xin Tong[3] Jiaolong Yang[3]
[1]Tsinghua University [2]USTC [3]Microsoft Research

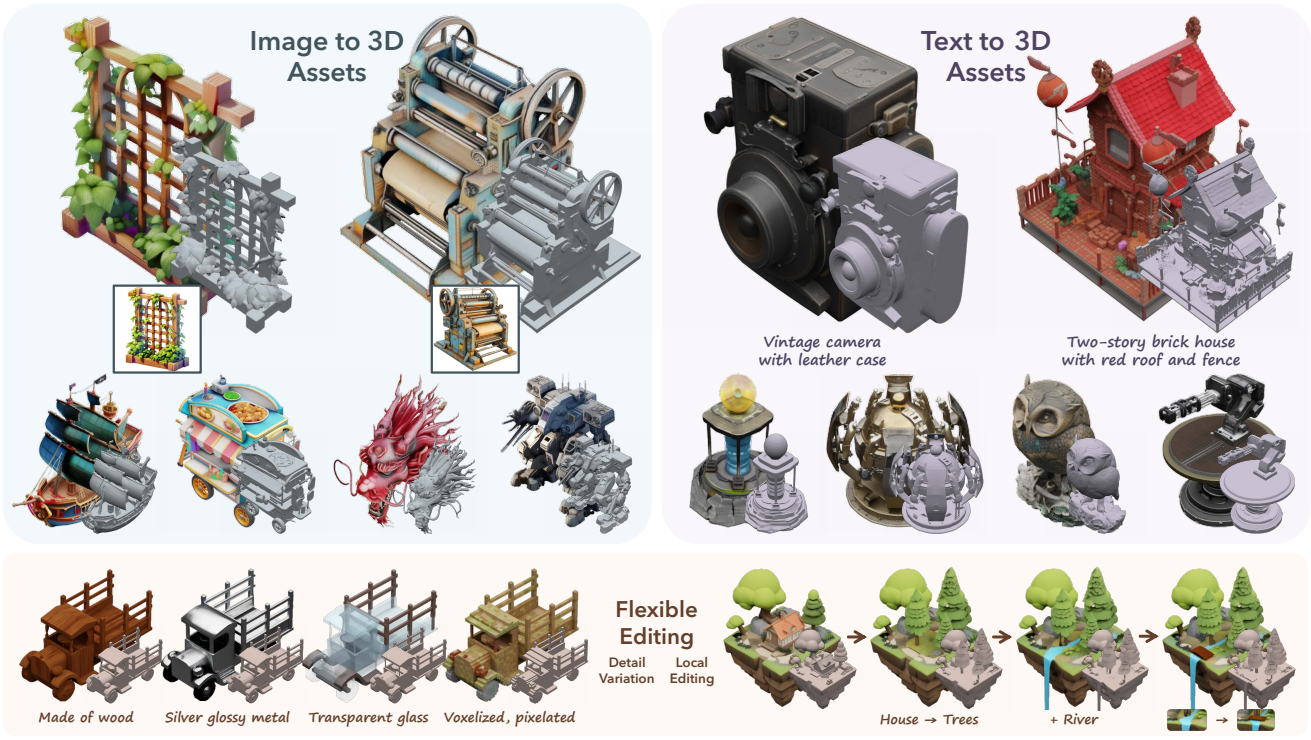

Figure 1. High-quality 3D assets generated by our method in various formats from text or image prompts (using GPT-4o and DALL-E 3). Our method enables versatile generation in about 10 seconds, offering vivid appearances with 3D Gaussians or Radiance Fields and detailed geometries with meshes. It also supports flexible 3D editing. ***Best viewed with zoom-in.***

Abstract

*We introduce a novel 3D generation method for versatile and high-quality 3D asset creation. The cornerstone is a unified Structured LATent (*SLAT*) representation which allows decoding to different output formats, such as Radiance Fields, 3D Gaussians, and meshes. This is achieved by integrating a sparsely-populated 3D grid with dense multiview visual features extracted from a powerful vision foundation model, comprehensively capturing both structural (geometry) and textural (appearance) information while maintaining flexibility during decoding.*

We employ rectified flow transformers tailored for SLAT *as our 3D generation models and train models with up to 2 billion parameters on a large 3D asset dataset of 500K diverse objects. Our model generates high-quality results with text or image conditions, significantly surpassing existing methods, including recent ones at similar scales. We showcase flexible output format selection and local 3D editing capabilities which were not offered by previous models. Project Page:* `trellis3d.github.io`.

1. Introduction

While AI Generated Content (AIGC) for 3D has made tremendous progress in recent years [42, 59, 77], existing 3D generative models still fall short in generation quality compared to their 2D predecessors, where large image generation models [8, 16] have enabled ready-to-use tools that

exert a profound impact on today's digital industry.

Unlike 2D images, typically represented by pixel grids, 3D data encompasses diverse representations like meshes, point clouds, Radiance Fields [51], and 3D Gaussians [29]. Each format is tailored for specific applications and may encounter difficulties when adapted for other tasks. For instance, while numerous studies [11, 21, 35, 63, 85, 89, 93] have utilized 3D representations like meshes or implicit fields [50, 57] for object geometry generation, they often falter in detailed appearance modeling compared to those relying on representations equipped with advanced volumetric rendering capabilities (*e.g.*, 3D Gaussians and Radiance Fields). Conversely, generative models based on Radiance Fields or 3D Gaussians [31, 80, 91] excel in rendering high-quality appearances but strruggle with plausible geometry extraction. Moreover, the unique structured or unstructured characteristics of different representations complicate processing through a consistent network architecture. These issues hinder the development of a standardized 3D generative modeling paradigm, in contrast to the consensus in recent advanced 2D generation methods that learn generative models within a unified latent space [16, 64].

In this paper, we aim to develop a *unified and versatile latent space* that facilitates high-quality 3D generation across various representations, accommodating diverse downstream requirements. This problem is highly challenging and has rarely been addressed by previous approaches. To tackle this, our primary strategy involves introducing explicit sparse 3D structures in the latent space design. These structures enable decoding into different 3D representations by characterizing attributes within the local voxels surrounding an object, as is evidenced by recent advancements in the 3D reconstruction field [19, 47, 65]. This approach also allows for efficient high-resolution modeling by bypassing voxels without 3D information [39, 63], and introduces locality that facilitates flexible editing.

However, even with such structures, achieving high-quality decoding into different 3D representations is still non-trivial, as it requires the latent representation to encapsulate both comprehensive geometry and appearance information of the 3D assets. To address this issue, our second strategy is to equip the sparse structures with a powerful vision foundation model [56] for detailed information encoding, given its demonstrated strong 3D awareness [15] and capability for detailed representation [99]. This approach bypasses the need for a dedicated 3D encoder, and eliminates the costly pre-fitting process of aligning 3D data with specific representations [80, 91].

Given these two strategies, we introduce Structured LATents (SLAT), a unified 3D latent representation for high-quality, versatile 3D generation. SLAT marries *sparse structures* with powerful *visual representations*. It defines local latents on active voxels intersecting the object's surface. The local latents are encoded by fusing and processing image features from densely rendered views of the 3D asset, while attaches them onto active voxels. These features, derived from powerful pretrained vision encoders [56], capture detailed geometric and visual characteristics, complementing the coarse structure provided by the active voxels. Different decoders can then be applied to map SLAT to diverse 3D representations of high quality.

Building on SLAT, we train a family of large 3D generation models, dubbed TRELLIS in this paper, with text prompts or images as conditions. A two stage pipeline is applied which first generates the sparse structure of SLAT, followed by generating the latent vectors for non-empty cells. We employ rectified flow transformers as our backbone models and adapt them properly to handle the sparsity in SLAT. We train TRELLIS with up to 2 billion parameters on a large dataset of carefully-collected 3D assets. Through extensive experiments, we show that our model can create high-quality 3D assets with detailed geometry and vivid texture, significantly surpassing previous methods. Moreover, it can easily generate 3D assets with different output formats to meet diverse downstream requirements.

We summarize the notable features of our method below:

- **High quality.** It produces diverse 3D assets at high-quality with intricate shape and texture details.
- **Versatile generation.** It takes text or image prompts and can generate various final 3D representations including but not limited to Radiance Fields, 3D Gaussians, and meshes.
- **Flexible editing.** It enables flexible tuning-free 3D editing such as the deletion, addition, and replacement of local regions, guided by text or image prompts.
- **Fitting-free training.** No 3D fitting is needed for the training objects in the entire process.

Given these strong performance and multifold advantages, we believe our new models can serve as powerful 3D generation foundations and unlock new possibilities for the 3D vision community. We hope our work can shed some light on 3D-representation-agnostic asset modeling, in contrast to the field's relentless pursuit of and adaptation to new representations.

2. Related Works

3D generative models. Early 3D generation methods primarily leveraged Generative Adversarial Nets (GANs) [20] to model 3D distributions [6, 14, 18, 68, 82, 96, 98], but faced challenges in scaling to more diverse scenarios. Later approaches employed diffusion models [25, 69] for various representations like point clouds [49, 54], voxel grids [27, 52, 75], Triplanes [7, 67, 80, 90], and 3D Gaussians [22, 91]. Some alternatives [9, 53] adopted GPT-style autoregressive models [61] for mesh generation. Despite

these advancements, efficiency remains a challenge for generative modeling in raw data space.

To enhance both quality and efficiency, recent studies have resorted to generation in a more compact latent space [64]. Some methods [34, 63, 78, 83, 89, 93, 95, 97] mainly focused on shape modeling, often requiring an additional texturing phase for complete 3D asset generation. Among them, a few approaches [21, 85] incorporated appearance information, but faced difficulties to model highly detailed appearance due to their surface representations. Other works [28, 31, 32, 55, 87] built latent representations for Radiance Fields or 3D Gaussians, which may pose challenges for accurate surface modeling. [10] encoded both geometry and appearance using latent primitives, but its prefitting process is both costly and lossy. In this work, we aim to build a versatile latent space that supports decoding into various 3D representations of high quality.

3D creation with 2D generative models. Instead of directly training 3D generative models, some recent methods leveraged 2D generative models to create 3D assets due to their superior generalization abilities. A pivotal work, DreamFusion [59], optimized 3D assets by distilling from pre-trained image diffusion models [64], followed by a large group of successors [36, 37, 72, 74, 81] with more advanced distillation techniques. Another group of works [26, 33, 40–42, 45, 66, 73, 84, 92, 99] involves generating multiview images via 2D diffusions and reconstructing 3D assets from them. However, these 2D-assisted approaches often yield lower geometry quality compared to native 3D models learned from 3D data collections, due to inherent multiview inconsistency in 2D generative models.

Rectified flow models. Rectified flow models [3, 38, 43] have recently emerged as a novel generative paradigm that challenges the dominance of diffusions [25, 69]. Recent works [16, 76] have demonstrated the effectiveness of them for large-scale image and video generation. In this paper, we also apply rectified flow models and demonstrate their abilities for 3D generation at scale.

3. Methodology

We aim to generate high-quality 3D assets in various 3D representation formats given text or image conditions. Figure 2 shows an overview, with details described below.

3.1. Structured Latent Representation

For a 3D asset $\mathcal{O}$, we encode its geometry and appearance information using a unified structured latent representation $\boldsymbol{z}$, which defines a set of local latents on a 3D grid:

$$\boldsymbol{z} = \{(\boldsymbol{z}_i, \boldsymbol{p}_i)\}_{i=1}^{L}, \quad \boldsymbol{z}_i \in \mathbb{R}^C, \ \boldsymbol{p}_i \in \{0, 1, \dots, N-1\}^3, \quad (1)$$

where $\boldsymbol{p}_i$ is the positional index of an active voxel in the 3D grid intersecting with the surface of $\mathcal{O}$, $\boldsymbol{z}_i$ denotes a local latent attached to the corresponding voxel, the derivation of which will be described later, N is the spatial length of the 3D grid, and L is the total number of active voxels. Intuitively, the active voxels $\boldsymbol{p}_i$ outline the coarse structure of the 3D asset, while the latents $\boldsymbol{z}_i$ capture finer details of appearance and shape. Together, these structured latents encompass the entire surface of $\mathcal{O}$, effectively capturing both the overall form and intricate details.

Due to the sparsity of 3D data, the number of active voxels is significantly smaller than the total size of the grid, *i.e.*, $L \ll N^3$, allowing to be constructed at a relatively high resolution. By default, we set $N = 64$ which leads to an average value of $L = 20$K.

3.2. Structured Latents Encoding and Decoding

With the structured latent representation, we develop an effective encoding scheme to encode 3D assets to it, and introduce different decoders for reconstruction across various 3D representations. The details are outlined below.

Visual feature aggregation. We first convert each 3D asset $\mathcal{O}$ into a voxelized feature $\boldsymbol{f} = \{(\boldsymbol{f}_i, \boldsymbol{p}_i)\}_{i=1}^{L}$. Here, $\boldsymbol{p}_i$ is the active voxels as defined in Eq. (1), and $\boldsymbol{f}_i$ is a visual feature recording detailed structure and appearance information of the local region.

To derive $\boldsymbol{f}_i$ for each active voxel, we aggregate features extracted from dense multiview images of $\mathcal{O}$. We render images from randomly sampled camera views on a sphere and extract feature maps using a pre-trained DINOv2 encoder [56]. Each voxel is projected onto the multiview feature maps to retrieve features at corresponding locations, and their average is used as $\boldsymbol{f}_i$, as shown in Fig. 2 (left-top). We set $\boldsymbol{f}$ to match the resolution of the structured latents $\boldsymbol{z}$ (*i.e.*, 64^3). Empirically, this is sufficient to reconstruct the original 3D asset at high fidelity, thanks to the strong representation capabilities of DINOv2 features together with the coarse structure provided by the active voxels.

Sparse VAE for structured latents. With the voxelized feature $\boldsymbol{f}$, we introduce a transformer-based VAE architecture for 3D assets encoding.

Specifically, an encoder $\boldsymbol{\mathcal{E}}$ first encodes $\boldsymbol{f}$ to structured latents $\boldsymbol{z}$, followed by a decoder $\boldsymbol{\mathcal{D}}$ that converts $\boldsymbol{z}$ into a 3D asset represented by certain 3D representation. Reconstruction losses are then applied between the decoded 3D assets and the ground truth to train the encoder and decoder in an end-to-end manner, along with a KL-penalty on $\boldsymbol{z}_i$ to encourage normal distribution regularization following [64].

The encoder and decoder share the same transformer structure, as shown in Fig. 3a. To handle sparse voxels, we serialize input features from active voxels and add sinusoidal positional encodings based on their voxel positions, creating tokens with variable context length L, which are subsequently processed through transformer blocks. Con-

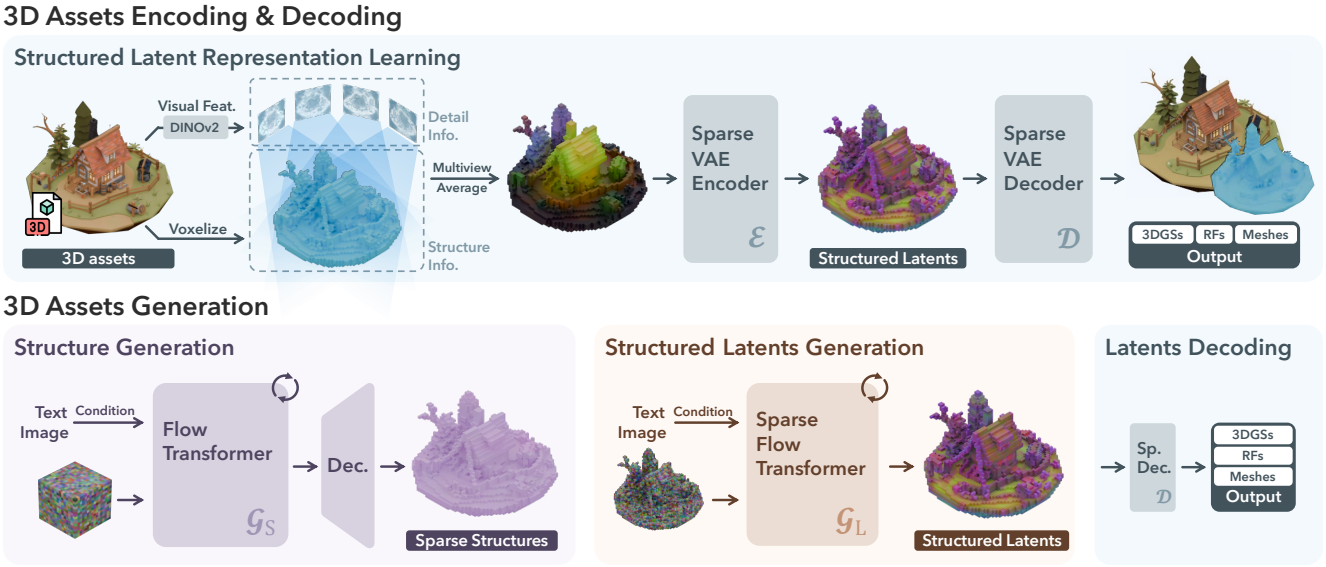

Figure 2. Overview of our method. **Encoding & Decoding:** We adopt a structured latent representation (SLAT) for 3D assets encoding, which defines local latents on a sparse 3D grid to represent both geometry and appearance information. It is encoded from the 3D assets by fusing and processing dense multiview visual features extracted from a DINOv2 encoder, and can be decoded into versatile output representations with different decoders. **Generation:** Two specialized rectified flow transformers are utilized to generate SLAT, one for the sparse structure and the other for local latents attached to it.

sidering the locality characteristic of the latents, we incorporate shifted window attention [44, 88] in 3D space to enhance local information interaction, which also improves efficiency compared to a full attention implementation.

Decoding into versatile formats. Our structured latents support decoding into diverse 3D representations, such as 3D Gaussians, Radiance Fields, and meshes, via respective decoders: $\mathcal{D}_{\text{GS}}$, $\mathcal{D}_{\text{RF}}$, and $\mathcal{D}_{\text{M}}$. These decoders share the same architecture except for their output layers, and can be trained using specific reconstruction losses tailored to their representations:

(a) 3D Gaussians. The decoding process is formulated as:

$$\mathcal{D}_{\text{GS}}: \{(\boldsymbol{z}_i, \boldsymbol{p}_i)\}_{i=1}^{L} \rightarrow \{\{(\boldsymbol{o}_i^k, \boldsymbol{c}_i^k, \boldsymbol{s}_i^k, \alpha_i^k, \boldsymbol{r}_i^k)\}_{k=1}^{K}\}_{i=1}^{L}, \quad (2)$$

where each $\boldsymbol{z}_i$ is decoded into K Gaussians with position offsets $\boldsymbol{o}$, colors $\boldsymbol{c}$, scales $\boldsymbol{s}$, opacities α, and rotations $\boldsymbol{r}$. To maintain locality of $\boldsymbol{z_i}$, we constrain the final positions $\boldsymbol{x}$ of the Gaussians to the vicinity of their active voxel: $\boldsymbol{x}_i^k = \boldsymbol{p}_i + \tanh(\boldsymbol{o}_i^k)$. The reconstruction losses consist of $\mathcal{L}_1$, D-SSIM and LPIPS [94] between rendered Gaussians and the ground truth images.

(b) Radiance Fields. The decoding process is defined as:

$$\mathcal{D}_{\text{RF}}: \{(\boldsymbol{z}_i, \boldsymbol{p}_i)\}_{i=1}^{L} \rightarrow \{(\boldsymbol{v}_i^{\text{x}}, \boldsymbol{v}_i^{\text{y}}, \boldsymbol{v}_i^{\text{z}}, \boldsymbol{v}_i^{\text{c}})\}_{i=1}^{L}, \quad (3)$$

where $\boldsymbol{v}_i^{\text{x}}, \boldsymbol{v}_i^{\text{y}}, \boldsymbol{v}_i^{\text{z}} \in \mathbb{R}^{16\times 8}$ and $\boldsymbol{v}_i^{\text{c}} \in \mathbb{R}^{16\times 4}$ are the CP-decomposition of a local radiance volume at 8^3 following Strivec [19], while the reconstruction losses are similar to those for Gaussians.

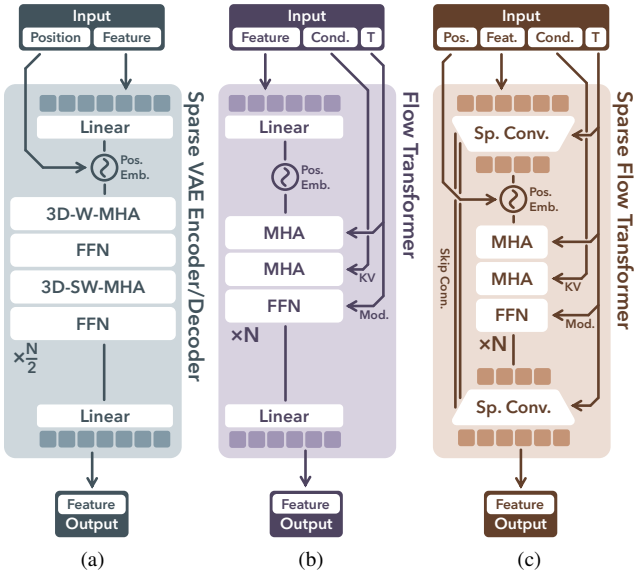

Figure 3. The network structures for encoding, decoding, and generation.

(c) Meshes. The decoding process is as follows:

$$\mathcal{D}_{\text{M}}: \{(\boldsymbol{z}_i, \boldsymbol{p}_i)\}_{i=1}^{L} \rightarrow \{\{(\boldsymbol{w}_i^j, d_i^j)\}_{j=1}^{64}\}_{i=1}^{L}, \quad (4)$$

where $\boldsymbol{w}_i^j \in \mathbb{R}^{24}$ are the flexible parameters in FlexiCubes [65] and $d_i^j \in \mathbb{R}$ is a signed distance. We append two convolutional upsampling blocks after the transformer backbone to increase the final output resolution to 256^3 (*i.e.*, each $\boldsymbol{z_i}$ for a grid of 4^3), extract meshes from 0-level iso-

surfaces, and compute $\mathcal{L}_1$ between rendered depth (normal) maps and their ground truth as the reconstruction losses.

In practice, we adopt Gaussians to learn the encoder and decoder end-to-end due to their high fidelity and efficiency. For other output formats, we simply freeze the learned encoder and train their decoders from scratches as described above. Despite trained with Gaussians, the learned structured latents can faithfully reconstruct other formats, demonstrating strong extensibility (See Tab. 1).

3.3. Structured Latents Generation

We introduce a two-stage generation pipeline to generate the structured latents, which first generates the sparse structure, followed by the local latents attached to it. For modeling the latent distribution, we employ rectified flow models [38]. We will first provide a brief introduction to these models before detailing our generation pipeline.

Rectified flow models. Rectified flow models use a linear interpolation forward process, $\boldsymbol{x}(t) = (1-t)\boldsymbol{x}_0 + t\boldsymbol{\epsilon}$, which interpolates between data samples $\boldsymbol{x}_0$ and noises $\boldsymbol{\epsilon}$ with a timestep t. The backward process is represented as a time-dependent vector field, $\boldsymbol{v}(\boldsymbol{x}, t) = \nabla_t \boldsymbol{x}$, moving noisy samples toward the data distribution, and can be approximated with a neural network $\boldsymbol{v}_\theta$ by minimizing the conditional flow matching (CFM) objective [38]:

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t,\boldsymbol{x}_0,\boldsymbol{\epsilon}} \|\boldsymbol{v}_\theta(\boldsymbol{x}, t) - (\boldsymbol{\epsilon} - \boldsymbol{x}_0)\|_2^2. \quad (5)$$

Sparse structure generation. In the first stage, we aim to generate the sparse structure $\{\boldsymbol{p}_i\}_{i=1}^L$. To enable this with a tensorized neural network, we convert the sparse active voxels into a dense binary 3D grid $\boldsymbol{O} \in \{0,1\}^{N\times N\times N}$, setting voxel values to 1 if active, and 0 otherwise.

Directly generating the dense grid $\boldsymbol{O}$ is computationally expensive. We introduce a simple VAE with 3D convolutional blocks to compress it into a low-resolution feature grid $\boldsymbol{S} \in \mathbb{R}^{D\times D\times D\times C_S}$. Since $\boldsymbol{O}$ represents only coarse geometry, this compression is nearly lossless, enhancing efficiency significantly. It also converts the discrete values in $\boldsymbol{O}$ into continuous features suited for rectified flow training.

We introduce a simple transformer backbone $\boldsymbol{\mathcal{G}}_S$ for generating $\boldsymbol{S}$, as shown in Fig. 3b. An input dense noisy grid is serialized, combined with positional encodings (as in Sec. 3.2), and fed into the transformer for denoising. Timestep information is incorporated using adaptive layer normalization (adaLN) and a gating mechanism [58]. Conditions are injected through cross attention layers as keys and values. For text conditions, we use features from a pre-trained CLIP [62] model. For image conditions, we adopt visual features from DINOv2. The denoised feature grid $\boldsymbol{S}$ is decoded into the discrete grid $\boldsymbol{O}$, and further converted back to active voxels $\{\boldsymbol{p}_i\}_{i=1}^L$ as the final sparse structure.

Table 1. Reconstruction fidelity of different latent representations. (†: evaluated using albedo color; ‡: evaluated via Radiance Fields)

Method	Appearance		Geometry			
	PSNR↑	LPIPS↓	CD↓	F-score↑	PSNR-N↑	LPIPS-N↓
LN3Diff	26.44	0.076	0.0299	0.9649	27.10	0.094
3DTopia-XL	25.34†	0.074†	0.0128	0.9939	31.87	0.080
CLAY	–	–	0.0124	0.9976	35.35	0.035
Ours	**32.74**/32.19‡	**0.025**/0.029‡	**0.0083**	**0.9999**	**36.11**	**0.024**

Structured latents generation. In the second stage, we generate latents $\{\boldsymbol{z}_i\}_{i=1}^L$ given the structure $\{\boldsymbol{p}_i\}_{i=1}^L$ using a transformer $\boldsymbol{\mathcal{G}}_L$ designed for sparse structures (Fig. 3c).

Instead of directly serializing input noisy latents as in the sparse VAE encoder in Sec. 3.2, we improve efficiency by packing them into a shorter sequence before serialization, similarly as done by DiT [58]. Due to our sparse structure, we apply a downsampling block with sparse convolutions [79] to pack latents within a 2^3 local region, followed by multiple time-modulated transformer blocks. A convolutional upsampling block is appended at the end of the transformer, with skip connections to the downsampling block that facilitates spatial information flow. Like in $\boldsymbol{\mathcal{G}}_S$, timesteps are integrated via adaLN layers, and text/image conditions are injected through cross-attentions.

We train $\boldsymbol{\mathcal{G}}_S$ and $\boldsymbol{\mathcal{G}}_L$ separately using the CFM objective in Eq. (5). After training, structured latents $\boldsymbol{z} = \{(\boldsymbol{z}_i, \boldsymbol{p}_i)\}_{i=1}^L$ can be sequentially generated by the two models and converted into high-quality 3D assets in various formats by different decoders: $\boldsymbol{\mathcal{D}}_{\text{GS}}$, $\boldsymbol{\mathcal{D}}_{\text{RF}}$, and $\boldsymbol{\mathcal{D}}_{\text{M}}$.

3.4. 3D Editing with Structured Latents

Our method supports flexible 3D editing and we present two simple *tuning-free* editing strategies.

Detail variation. The separation between the structure and latents enables detail variation of 3D assets without affecting the overall coarse geometry. This can be easily accomplished by preserving the asset's structure and executing the second generation stage with different text prompts.

Region-specific editing. The locality of SLAT allows for region-specific editing by altering voxels and latents in targeted areas while leaving others unchanged. To this end, we adapt Repaint [48] to our two-stage generation pipeline. Given a bounding box for the voxels to be edited, we modify our flow models' sampling processes to create new content in that region, conditioned on the unchanged areas and any provided text or image prompts. Consequently, the first stage generates new structures within the specified region, and the second stage produces coherent details.

4. Experiments

Implementation details. For training, we carefully collect approximately 500K high-quality 3D assets from 4

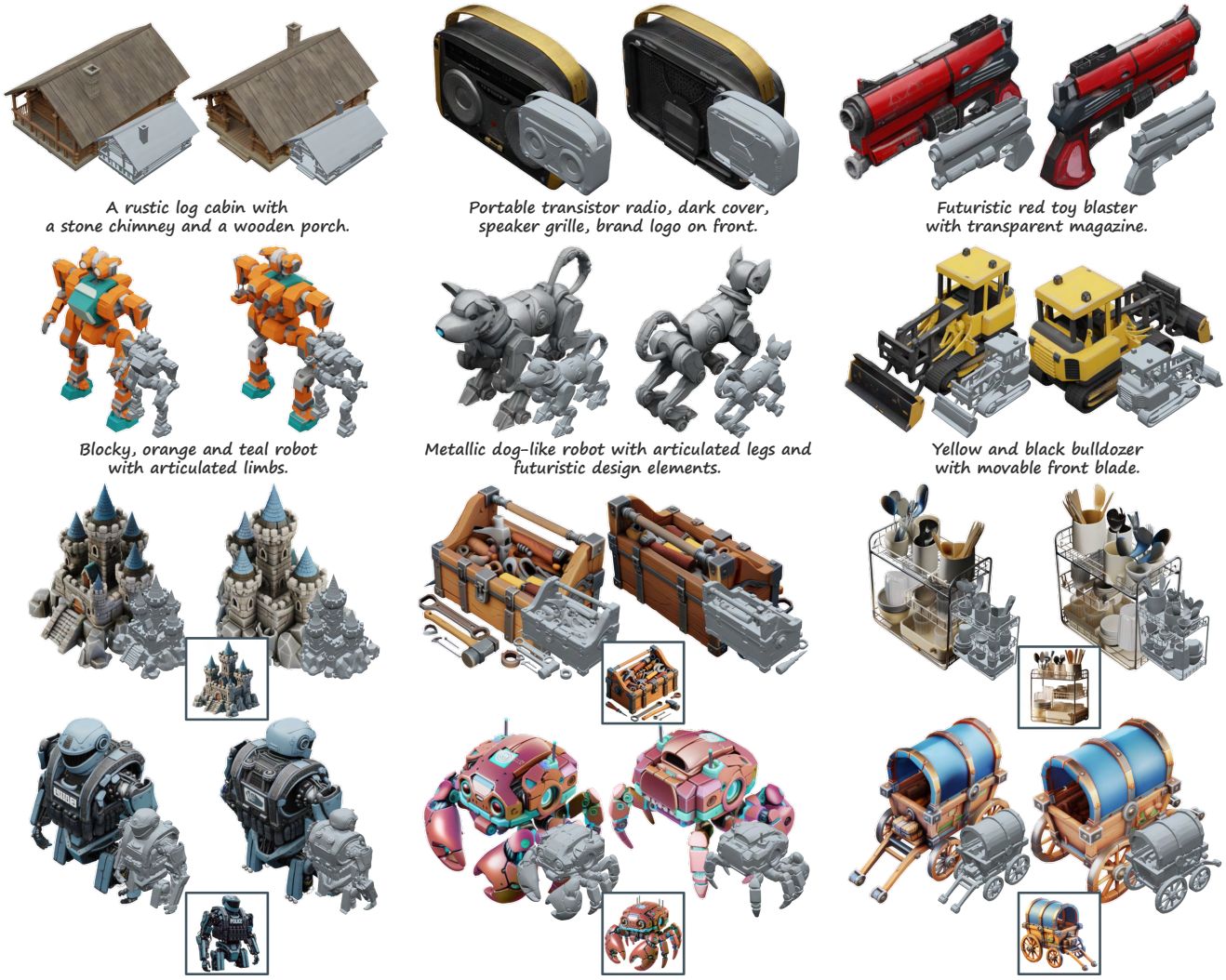

Figure 4. High-quality 3D assets created by our method, represented in Gaussians and meshes, given AI-generated text or image prompts.

public datasets: Objaverse (XL) [13], ABO [12], 3D-FUTURE [17], and HSSD [30]. We render 150 images per asset, and employ GPT-4o [1] for captioning. Data augmentation is applied to both text and image prompts: texts are summarized to varying lengths, and images are rendered with different FoVs. We use classifier-free guidance (CFG) [24] with a drop rate of 0.1 and AdamW [46] optimizer with a learning rate of $1e-4$. We train three models with total parameters of 342M (Basic), 1.1B (Large), and 2B (X-Large). The XL model is trained with 64 A100 GPUs (40G) for 400K steps with a batchsize of 256. At inference, CFG strength is set to 3 and sampling steps to 50.

For quantitative evaluations, we use Toys4k [70], which is not part of our training set or those of the compared methods. For visual results, comparisons, and user studies, we use text generated by GPT-4 [2] and images by DALL-E 3 [4]. Our method uses decoded *Gaussians for appearance* evaluation and *meshes for geometry*, unless specified otherwise. Refer to the *suppl. material* for more details.

4.1. Reconstruction Results

We first assess the reconstruction fidelity of different latent representations. We compare SLAT with alternatives also learned from large-scale data: latent point clouds from 3DTopia-XL [10], latent vector sets from CLAY [93], and latent triplanes from LN3Diff [31].

For appearance fidelity, we report PSNR and LPIPS between rendered reconstruction results and ground truth. For geometry quality, we use Chamfer Distance (CD) and F-score to assess overall shape accuracy, and PSNR and LPIPS for rendered normal maps to evaluate surface details.

As shown in Tab. 1, our method outperforms all baselines across all evaluated metrics. For geometry, it even surpasses CLAY which focuses solely on shape encoding. The high-fidelity reconstruction results under diverse output formats demonstrates strong versatility of SLAT.

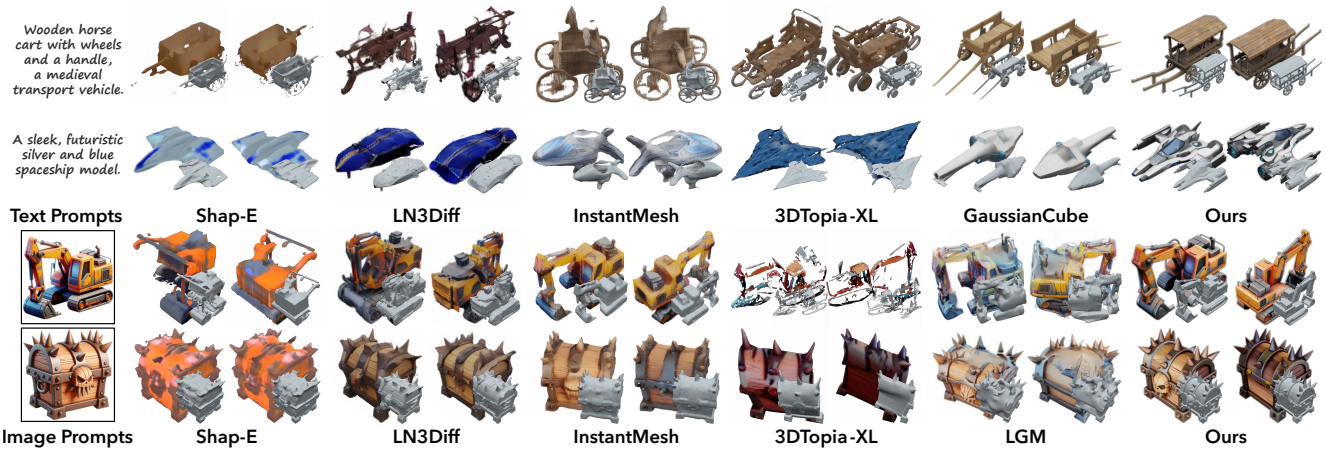

Figure 5. Visual comparisons of generated 3D assets between our method and previous approaches, given AI-generated prompts.

Table 2. Quantitative comparisons using Toys4k [70]. (KD is reported $\times 100$. †: evaluated using shaded images of PBR meshes.)

Method	Text-to-3D						Image-to-3D					
	CLIP↑	FD_{incep}↓	KD_{incep}↓	FD_{dinov2}↓	KD_{dinov2}↓	FD_{point}↓	CLIP↑	FD_{incep}↓	KD_{incep}↓	FD_{dinov2}↓	KD_{dinov2}↓	FD_{point}↓
Shap-E	25.04	37.93	0.78	497.17	49.96	6.58	82.11	34.72	0.87	465.74	62.72	8.20
LGM	24.83	36.18	0.77	507.47	61.89	24.73	83.97	26.31	0.48	322.71	38.27	15.90
InstantMesh	25.56	36.73	0.62	478.92	49.77	10.79	84.43	20.22	0.30	264.36	25.99	9.63
3DTopia-XL	22.48†	53.46†	1.39†	756.37†	87.40†	13.72	78.45†	37.68†	1.20†	437.37†	53.24†	18.21
Ln3Diff	18.69	71.79	2.85	976.40	154.18	19.40	82.74	26.61	0.68	357.93	50.72	7.86
GaussianCube	24.91	27.35	0.30	460.07	39.01	29.95	–	–	–	–	–	–
Ours L	26.60	20.54	**0.08**	238.60	4.24	5.24	**85.77**	**9.35**	**0.02**	**67.21**	**0.72**	**2.03**
Ours XL	**26.70**	**20.48**	**0.08**	**237.48**	**4.10**	**5.21**	–	–	–	–	–	–

4.2. Generation Results

In this section, we evaluate our generation quality. We first present various 3D generation results of our method, and then compare with other baseline methods.

Text/image-to-3D generation. Figure 4 showcases 3D assets generated by our method, where the text and image prompts are given below. We present two views for each asset: front-left and back-right.

Upon visual inspection, our method produces 3D assets with an unprecedented level of quality. The generated appearances possess vibrant colors and vivid details, such as the radio speaker's grille and the toy blaster's scratches. The geometries reveal complex structures and fine shape details, with superior surface properties like flat faces and sharp edges (*e.g.*, the bulldozer's hollow driving cab and the equipment on the police robot). It can even handle *translucent objects* such as the drinking glasses on the kitchen rack. Additionally, the generated contents closely match the elements from the provided text (*e.g.*, the log cabin with a stone chimney and wooden porch) and faithfully adhere to details from input images (*e.g.*, the castle with brick walls). More results can be found in Fig. 1 and the *suppl. material*.

Qualitative comparisons. We compare our approach with existing 3D generation methods that utilize different generative paradigms, latent representations, and output formats, including 2D-assisted methods: InstantMesh [86] and LGM [73]; and 3D generative approaches: GaussianCube [91], Shap-E [28], 3DTopia-XL, and LN3Diff. We do not compare with CLAY in this phase, as their generation models are currently unavailable to us.

Figure 6. User study for text/image-to-3D generation.

We begin by presenting visual comparisons in Fig. 5. Our method outperforms all previous approaches, offering not only more vivid appearances and finer geometries but also more precise alignment with the provided text and image prompts. It excels at producing intricate and coherent details, whereas alternatives experience varying degrees of quality degradation: The 2D-assisted methods suffer from structural distortion due to multiview inconsistencies inherent in the 2D generative models they rely on; other 3D generative approaches encounter featureless appearances and geometries, constrained by the limited reconstruction fidelity of their latent representations. GaussianCube and LGM do not provide plausible geometries, which is an inherent issue with their 3D Gaussian representations.

Quantitative comparisons. Furthermore, we perform quantitative comparisons using text and image prompts in Toys4k and present the results in Tab. 2. We utlize Fréchet distance (FD) [23] and kernel distance (KD) [5] with various feature extractors (*i.e.*, Inception-v3 [71], DINOv2, and

Table 3. Ablation study on the size of SLAT.

Resolution	Channel	PSNR↑	LPIPS↓
32	16	31.64	0.0297
32	32	31.80	0.0289
32	64	31.85	0.0283
64	8	**32.74**	**0.0250**

Table 4. Ablation study on different generation paradigms.

	Method	Training set		Toys4k	
		CLIP↑	FD_{dinov2}↓	CLIP↑	FD_{dinov2}↓
Stage 1	Diffusion	25.09	132.71	25.86	295.90
	Rectified flow	**25.40**	**113.42**	**26.37**	**269.56**
Stage 2	Diffusion	25.58	100.88	26.45	244.08
	Rectified flow	**25.65**	**95.97**	**26.61**	**240.20**

Table 5. Ablation study on model size.

Method	Training set		Toys4k	
	CLIP↑	FD_{dinov2}↓	CLIP↑	FD_{dinov2}↓
B	25.41	121.45	26.47	265.26
L	25.62	99.92	26.60	238.60
XL	**25.71**	**93.96**	**26.70**	**237.48**

PointNet++ [60]) to assess overall quality of the generated outputs, and use CLIP score [62] to evaluate the consistency between the generated results and the input prompts. As demonstrated, our method significantly surpasses previous methods across all evaluated metrics.

User study. In addition, we conduct a user study with over 100 participants to compare different methods based on human preferences. We leverage 68 AI-generated text prompts and 67 image prompts, and create 3D assets from them via each method without any curation. As illustrated in Fig. 6, our method is strongly preferred by users due to its significant improvements in generation quality.

4.3. Ablation Study

We conduct ablation studies to validate the design choices of our method under the text-to-3D configuration.

Size of structured latents. To determine the size for SLAT, we train sparse VAEs with varying latent resolutions and channels. As shown in Tab. 3, while the performance under 32^3 is quite good, it tends to plateau as the number of latent channels increases. Switching to 64^3 provides a significant boost. We prioritize quality over efficiency and adopt 64^3 as our default setting for SLAT.

Rectified flow *v.s.* diffusion. We compare rectified flow models with a widely used diffusion baseline [58] in Tab. 4. We independently alter the generation method at each stage using the large model size, while maintaining the XL model unchanged for the other stages. As shown, replacing diffusion models with rectified flow models at any stage improves both generation quality and prompt alignment.

Model size. We examine the model's performance with varying numbers of parameters. Table 5 shows that increasing the model size consistently improves the generation performance on both training distribution and Toys4k.

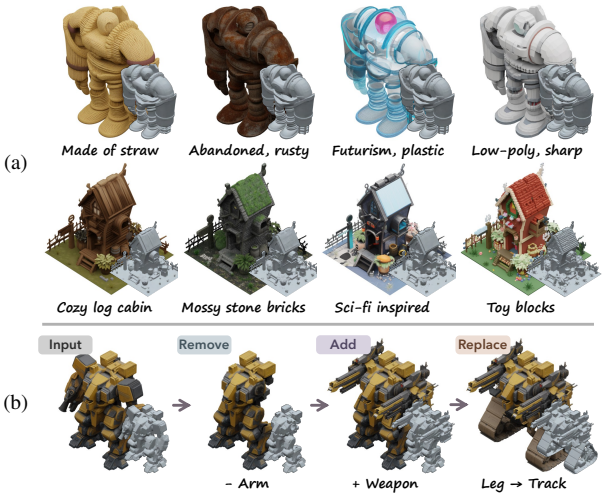

Figure 7. **Top:** Given coarse structures, our method generates 3D asset variations coherent with the text prompts. **Bottom:** Tuning-free region-specific editing results of our method, guided by text or image prompts. More results in Fig.1.

4.4. Applications

We demonstrate tuning-free applications of our method by utilizing the editing strategies described in Sec. 3.4.

3D asset variations. Figure 1 and 7a show 3D asset variation results. Our method produces variants adhering to the overall shape of the given structures while exhibiting diverse appearance and geometry details guided by the text.

Region-specific editing of 3D assets. Figure 1 and 7b illustrate the editing sequences of two 3D assets, involving removal, addition, and replacement operations. Corresponding prompts (either text or image) for each step are provided. Our method enables detailed local region editing, such as adding a river and bridge in the island example.

5. Conclusion

We introduced a novel 3D generation method for versatile and high-quality 3D asset creation. At its core lies SLAT, a structured latent representation that allows decoding to versatile output formats by comprehensively encoding both geometry and appearance information into localized latents anchored on a sparse 3D grid, where the latents are fused and processed from dense multiview image features extracted by a powerful vision foundation model. We proposed a two-stage generation pipeline utilizing rectified flow transformers tailored for SLAT generation at scale. Extensive experiments demonstrated the superiority of our method in 3D generation, in terms of quality, versatility, and editability, highlighting its strong potential for a wide range of real-world applications in digital production.

References

[1] Gpt-4o system card. 2024. 6

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 6

[3] Michael Samuel Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *ICLR*, 2023. 3

[4] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023. 6

[5] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. In *International Conference on Learning Representations*, 2018. 7

[6] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *IEEE/CVF International Conference on Computer Vision*, 2022. 2

[7] Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2416–2425, 2023. 2

[8] Junsong Chen, YU Jincheng, GE Chongjian, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *ICLR*, 2024. 1

[9] Yiwen Chen, Tong He, Di Huang, Weicai Ye, Sijin Chen, Jiaxiang Tang, Xin Chen, Zhongang Cai, Lei Yang, Gang Yu, et al. Meshanything: Artist-created mesh generation with autoregressive transformers. *arXiv preprint arXiv:2406.10163*, 2024. 2

[10] Zhaoxi Chen, Jiaxiang Tang, Yuhao Dong, Ziang Cao, Fangzhou Hong, Yushi Lan, Tengfei Wang, Haozhe Xie, Tong Wu, Shunsuke Saito, et al. 3dtopia-xl: Scaling high-quality 3d asset generation via primitive diffusion. *arXiv preprint arXiv:2409.12957*, 2024. 3, 6

[11] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4456–4465, 2023. 2

[12] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21126–21136, 2022. 6

[13] 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*, 36, 2024. 6

[14] Yu Deng, Jiaolong Yang, Jianfeng Xiang, and Xin Tong. Gram: Generative radiance manifolds for 3d-aware image generation. In *IEEE/CVF International Conference on Computer Vision*, 2022. 2

[15] Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21795–21806, 2024. 2

[16] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 1, 2, 3

[17] 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*, pages 1–25, 2021. 6

[18] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *Advances In Neural Information Processing Systems*, 35:31841–31854, 2022. 2

[19] Quankai Gao, Qiangeng Xu, Hao Su, Ulrich Neumann, and Zexiang Xu. Strivec: Sparse tri-vector radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17569–17579, 2023. 2, 4

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27, 2014. 2

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

[22] Xianglong He, Junyi Chen, Sida Peng, Di Huang, Yangguang Li, Xiaoshui Huang, Chun Yuan, Wanli Ouyang, and Tong He. Gvgen: Text-to-3d generation with volumetric representation. In *ECCV*, 2024. 2

[23] 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. 7

[24] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 6

[25] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2, 3

[26] 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. In *ICLR*, 2024. 3

[27] Ka-Hei Hui, Ruihui Li, Jingyu Hu, and Chi-Wing Fu. Neural wavelet-domain diffusion for 3d shape generation. In *SIGGRAPH Asia 2022 Conference Papers*, pages 1–9, 2022. 2

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

[29] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2

[30] 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. 6

[31] Yushi Lan, Fangzhou Hong, Shuai Yang, Shangchen Zhou, Xuyi Meng, Bo Dai, Xingang Pan, and Chen Change Loy. Ln3diff: Scalable latent neural fields diffusion for speedy 3d generation. In *ECCV*, 2024. 2, 3, 6

[32] Yushi Lan, Shangchen Zhou, Zhaoyang Lyu, Fangzhou Hong, Shuai Yang, Bo Dai, Xingang Pan, and Chen Change Loy. Gaussiananything: Interactive point cloud latent diffusion for 3d generation. In *ICLR*, 2025. 3

[33] 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. In *ICLR*, 2024. 3

[34] 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. 3

[35] Yuhan Li, Yishun Dou, Xuanhong Chen, Bingbing Ni, Yilin Sun, Yutian Liu, and Fuzhen Wang. Generalized deep 3d shape prior via part-discretized diffusion process. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16784–16794, 2023. 2

[36] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6517–6526, 2024. 3

[37] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023. 3

[38] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *ICLR*, 2023. 3, 5

[39] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020. 2

[40] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10072–10083, 2024. 3

[41] Minghua Liu, Chong Zeng, Xinyue Wei, Ruoxi Shi, Linghao Chen, Chao Xu, Mengqi Zhang, Zhaoning Wang, Xiaoshuai Zhang, Isabella Liu, et al. Meshformer: High-quality mesh generation with 3d-guided reconstruction model. *arXiv preprint arXiv:2408.10198*, 2024.

[42] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023. 1, 3

[43] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 3

[44] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 4

[45] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9970–9980, 2024. 3

[46] I Loshchilov. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[47] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20654–20664, 2024. 2

[48] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11461–11471, 2022. 5

[49] Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2837–2845, 2021. 2

[50] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 2

[51] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[52] Norman Müller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner. Diffrf: Rendering-guided 3d radiance field diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4328–4338, 2023. 2

[53] Charlie Nash, Yaroslav Ganin, SM Ali Eslami, and Peter Battaglia. Polygen: An autoregressive generative model of 3d meshes. In *International conference on machine learning*, pages 7220–7229. PMLR, 2020. 2

[54] 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. 2

[55] Evangelos Ntavelis, Aliaksandr Siarohin, Kyle Olszewski, Chaoyang Wang, Luc V Gool, and Sergey Tulyakov. Autodecoding latent 3d diffusion models. *Advances in Neural Information Processing Systems*, 36:67021–67047, 2023. 3

[56] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. 2, 3

[57] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 2

[58] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 5, 8

[59] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *ICLR*, 2023. 1, 3

[60] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 8

[61] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 2

[62] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 5, 8

[63] 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*, pages 4209–4219, 2024. 2, 3

[64] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2, 3

[65] Tianchang Shen, Jacob Munkberg, Jon Hasselgren, Kangxue Yin, Zian Wang, Wenzheng Chen, Zan Gojcic, Sanja Fidler, Nicholas Sharp, and Jun Gao. Flexible isosurface extraction for gradient-based mesh optimization. *ACM Trans. Graph.*, 42(4), 2023. 2, 4

[66] Yichun Shi, Peng Wang, Jianglong Ye, Long Mai, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. In *ICLR*, 2024. 3

[67] J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20875–20886, 2023. 2

[68] Ivan Skorokhodov, Aliaksandr Siarohin, Yinghao Xu, Jian Ren, Hsin-Ying Lee, Peter Wonka, and Sergey Tulyakov. 3d generation on imagenet. *arXiv preprint arXiv:2303.01416*, 2023. 2

[69] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 2, 3

[70] 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*, pages 1798–1808, 2021. 6, 7

[71] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 7

[72] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22819–22829, 2023. 3

[73] 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*, 2024. 3, 7

[74] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. In *ICLR*, 2024. 3

[75] Zhicong Tang, Shuyang Gu, Chunyu Wang, Ting Zhang, Jianmin Bao, Dong Chen, and Baining Guo. Volumediffusion: Flexible text-to-3d generation with efficient volumetric encoder. *arXiv preprint arXiv:2312.11459*, 2023. 2

[76] The Movie Gen team. Movie gen: A cast of media foundation model. https://ai.meta.com/research/movie-gen/, 2024. 3

[77] 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. 1

[78] Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, Karsten Kreis, et al. Lion: Latent point diffusion models for 3d shape generation. *Advances in Neural Information Processing Systems*, 35:10021–10039, 2022. 3

[79] Peng-Shuai Wang, Yang Liu, Yu-Xiao Guo, Chun-Yu Sun, and Xin Tong. O-cnn: octree-based convolutional neural networks for 3d shape analysis. *ACM Trans. Graph.*, 36(4), 2017. 5

[80] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4563–4573, 2023. 2

[81] 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. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[82] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and Josh Tenenbaum. Learning a probabilistic latent space of object shapes via 3d generative-adversarial modeling. *Advances in neural information processing systems*, 29, 2016. 2

[83] 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*, 2024. 3

[84] Jianfeng Xiang, Jiaolong Yang, Binbin Huang, and Xin Tong. 3d-aware image generation using 2d diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2383–2393, 2023. 3

[85] 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. *arXiv preprint arXiv:2408.14732*, 2024. 2, 3

[86] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. 7

[87] Haitao Yang, Yuan Dong, Hanwen Jiang, Dejia Xu, Georgios Pavlakos, and Qixing Huang. Atlas gaussians diffusion for 3d generation with infinite number of points. *arXiv preprint arXiv:2408.13055*, 2024. 3

[88] Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3d: A pretrained transformer backbone for 3d indoor scene understanding, 2023. 4

[89] 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. 2, 3

[90] Bowen Zhang, Yiji Cheng, Chunyu Wang, Ting Zhang, Jiaolong Yang, Yansong Tang, Feng Zhao, Dong Chen, and Baining Guo. Rodinhd: High-fidelity 3d avatar generation with diffusion models. *arXiv preprint arXiv:2407.06938*, 2024. 2

[91] Bowen Zhang, Yiji Cheng, Jiaolong Yang, Chunyu Wang, Feng Zhao, Yansong Tang, Dong Chen, and Baining Guo. Gaussiancube: Structuring gaussian splatting using optimal transport for 3d generative modeling. *arXiv preprint arXiv:2403.19655*, 2024. 2, 7

[92] 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*, 2024. 3

[93] 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, 2024. 2, 3, 6

[94] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 4

[95] Zibo Zhao, Wen Liu, Xin Chen, Xianfang Zeng, Rui Wang, Pei Cheng, Bin Fu, Tao Chen, Gang Yu, and Shenghua Gao. Michelangelo: Conditional 3d shape generation based on shape-image-text aligned latent representation. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[96] Xinyang Zheng, Yang Liu, Pengshuai Wang, and Xin Tong. Sdf-stylegan: Implicit sdf-based stylegan for 3d shape generation. In *Computer Graphics Forum*, pages 52–63, 2022. 2

[97] Xin-Yang Zheng, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Heung-Yeung Shum. Locally attentional sdf diffusion for controllable 3d shape generation. *ACM Transactions on Graphics (ToG)*, 42(4):1–13, 2023. 3

[98] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Josh Tenenbaum, and Bill Freeman. Visual object networks: Image generation with disentangled 3d representations. *Advances in neural information processing systems*, 31, 2018. 2

[99] 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*, pages 10324–10335, 2024. 2, 3