Can3Tok: Canonical 3D Tokenization and Latent Modeling of Scene-Level 3D Gaussians

Quankai Gao[1] Iliyan Georgiev[2] Tuanfeng Y. Wang[2] Krishna Kumar Singh[2]
Ulrich Neumann[1,†] Jae Shin Yoon[2,†]
[1]University of Southern California [2]Adobe Research

Abstract

3D generation has made significant progress, however, it still largely remains at the object-level. Feedforward 3D scene-level generation has been rarely explored due to the lack of models capable of scaling-up latent representation learning on 3D scene-level data. Unlike object-level generative models, which are trained on well-labeled 3D data in a bounded canonical space, scene-level generations with 3D scenes represented by 3D Gaussian Splatting (3DGS) are unbounded and exhibit scale inconsistency across different scenes, making unified latent representation learning for generative purposes extremely challenging. In this paper, we introduce Can3Tok, the first 3D scene-level variational autoencoder (VAE) capable of encoding a large number of Gaussian primitives into a low-dimensional latent embedding, which effectively captures both semantic and spatial information of the inputs. Beyond model design, we propose a general pipeline for 3D scene data processing to address scale inconsistency issue. We validate our method on the recent scene-level 3D dataset DL3DV-10K, where we found that only Can3Tok successfully generalizes to novel 3D scenes, while compared methods fail to converge on even a few hundred scene inputs during training and exhibit zero generalization ability during inference. Finally, we demonstrate image-to-3DGS and text-to-3DGS generation as our applications to demonstrate its ability to facilitate downstream generation tasks. Project page: https://github.com/Zerg-Overmind/Can3Tok

1. Introduction

Realistic 3D Scene-level generation enables immersive AR/VR applications. While tremendous progress has been made in 3D object-level generation using various 3D representations [16, 19, 45, 48, 65], with significant improvements ranging from per-scene optimization to feedforward approaches, less research has focused on 3D scene-level generation. Pioneering works such as WonderJourney [78] and LucidDreamer [9] generate 3D scene content without training on 3D scene-level data. However, their per-scene optimization is time-consuming, prone to to texture saturation, and lacks 3D consistency due to the using of 2D diffusion models. Our approach explores realistic feedforward 3D scene-level generation by directly training on real 3D data. Conceptually, it aligns with the idea of Stable Diffusion [55], which enables conditional generation by aligning different modalities (e.g., text and 2D images) within the same latent space for controllable generation. Since the diffusion process and architectures such as UNet or DiT [47] are well-known to be grounded, the key bottleneck we identified in feedforward 3D generation is the development of a 3D VAE for learning scene-level latent representations. However, we find this to be non-trivial, as scene-level content is not merely a combination of multiple object-level elements but also includes the background, scene layout, and the relative position, scale, and orientation of objects.

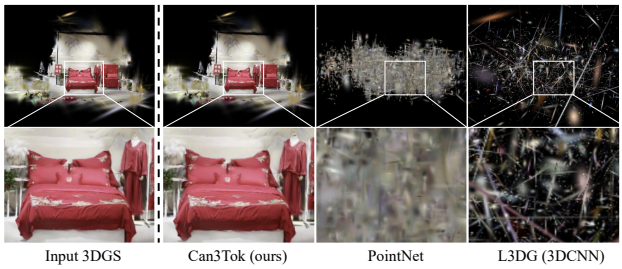


Figure 1. Reconstruction results from the latent space of 3D Gaussian splats (3DGS) of a general scene using our Can3Tok and other 3D-based VAE models [50, 53]. Can3Tok effectively preserves the global shape and local details while existing methods often fail to model the structured latent space from unstructured 3DGS.

In this paper, we explore to learn a new VAE that can project 3D data into a structured latent space. 3D Gaussian Splatting [25] (3DGS) is an emerging 3D representation that describes a scene as a set of Gaussians with a few parameters such as position, scaling values, and etc. One might wonder, is it still possible to build a latent space for many input 3DGS scene representations that are decodable with existing 3D-based VAE models, such as PointNet

[†]Equal advising.

VAE? Our experiments revealed that existing VAEs cannot reconstruct the input 3DGS as shown in Fig. 1, where the original scene structure is completely washed out as shown in Fig. 2.

The fundamental reasons include 1) the data structure of 3DGS is not compatible with existing VAE models: It is, in nature, highly unstructured due to its heterogeneous features (*i.e.*, representing geometry, appearance, and lighting) and irregularity like point clouds. Unlike object-level 3DGS representations that either have high quality multi-view rendered images from synthetic data [12] or well-captured 360-degree images from real world [79], the hallucination in 3DGS of a general scene is prominent due to insufficient multi-view observations during per-scene optimization, as shown in Fig. 2. Additionally, the large number of 3D Gaussians in a 3D scene makes it challenging to achieve a low-dimensional latent embedding. 2) Secondly, 3DGS representation of each scene has different global scene scales and also the individual scaling values of each 3D Gaussian primitive, making scaling up representation learning over a large number of 3DGS scene representations difficult.

We address the first problem by introducing a new 3DGS VAE module called Can3Tok, which tokenizes 3DGS inputs into canonical 3D tokens in a transformer-based VAE framework. To embed diverse and unstructured 3DGS representations with a large number of Gaussian primitives into a compact latent space, we first employ cross-attention with a low-dimensional learnable query, encoding 3DGS representations into a small-sized tensor to enhance the efficiency of subsequent self-attention computations. The learnable query is initialized with the coordinates of a regular volume in canonical space, introducing a structured geometry prior that facilitates representation learning on irregular data.

Due to the inconsistent scale over scenes introduced by the structure-from-motion initialization, we found that even a powerful VAE model alone fails to converge on more than a thousand input scene representations. In contrast, previous works on representation learning and generation for 2D images or 3D object-level tasks do not encounter this issue, as their data either naturally has a fixed size or resolution or is consistently bounded within a uniform scale [5, 12]. Therefore, we propose 3DGS normalization to unify both global scene scale and three-dimensional scaling values of all 3D Gaussian primitives. To further enhance quality and mitigate the impact of 3DGS tokens from hallucinated scene regions, we filter out noisy areas in the 3DGS reconstruction caused by insufficient supervision during per-scene optimization. This is achieved through semantic segmentation guidance [28] and K-nearest propagation among 3D Gaussian primitives, allowing us to extract only the cleanest and most salient scene partitions for training the VAE.

We validate our design using the same training and testing splits from DL3DV-10K [35], an open-source scene-level dataset. We found that only our Can3Tok successfully converges on 3D scenes in training and generalizes well to unseen input scenes in inference, whereas other convolutional-based and transformer-based methods, with more parameters and network layers than Can3Tok, fail to converge on even a few hundred scenes in training and exhibit no generalization ability to unseen inputs in inference. Our ablation study highlights the benefits from each proposed component for reconstruct local details. We show that our latent space could serve as a prototype for future 3DGS generative tasks such as text-to-3DGS or image-to-3DGS generation. In summary, our main contributions include:

- We propose Can3Tok, the first VAE model that tokenizes the scene-level 3DGS data into the canonical tokens by cross attention, enabling an unified latent representation learning that significantly outperforms existing VAEs;
- We propose a comprehensive data processing framework for 3DGS representation, including normalization to address scale inconsistency for large-scale training, as well as semantic-aware filtering and data augmentation to enhance output quality;
- We showcase feedforward image-guided and text-guided scene-level 3DGS generation as applications of our Can3Tok.

Figure 2. Example of unstructured 3DGS representation of a scene (left) and its decoding results using PointNet-based VAE [50, 51] (right). 3DGS is highly unstructured in continuous and irregular 3D continuum where it often includes many floaters for the region with insufficient multi-view observations.

2. Related Work

3D Gaussian Splatting. 3D Gaussian splatting (3DGS) [25] has become one of the most popular 3D representations due to its flexibility and faster rendering than neural radiance fields (NeRF) [44]. Its discrete nature can be greatly beneficial for tokenization and feed-forward 3D reconstruction with only a few or even single image inputs [61, 62, 82]. 3DGS is akin to a point cloud and enables per-pixel parameter prediction and lifting [14, 82] for novel-view synthesis like SynSin [74]. Advancements have been specifically designed for large-scale and complex scene reconstruction [7, 26, 34, 38], real-time rendering [46] and anti-aliasing [32, 80].

However, since 3DGS encompasses richer geometric and

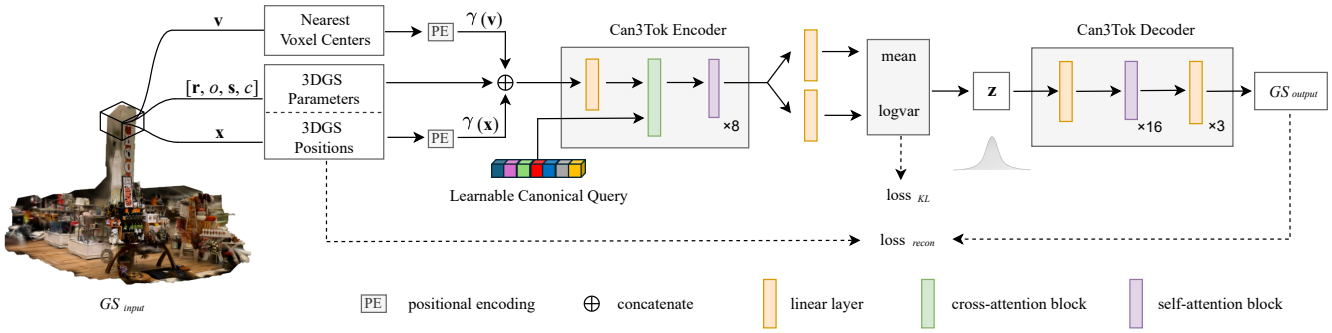


Figure 3. Can3Tok processes a batch of per-scene 3D Gaussians, with a batch size of B, where each scene contains the same number of Gaussians N. The encoder encodes input Gaussians into a low-dimensional latent space followed by a VAE reparametrization. And the decoder reconstructs the embeddings back into 3D space, corresponding to the original input 3D Gaussians.

appearance features and offers greater parameter flexibility than conventional point clouds, directly integrating diffusion modules for generation is non-trivial. Recent explorations in 3D Gaussian embedding [40, 53] and rearrangement [81] have focused on converting raw 3D Gaussian representations into more structured forms through voxelization or low-level encoding since diffusion models such as UNet and DiT cannot learn a denoising process. These approaches are currently limited to small-scale or object-level content.

Latent-Space Modeling. Structured latent space modeling across various data modalities, including images, videos, audio, and 3D representations, has been a long-standing problem for its advantages in compression, efficiency, and generalizability. Especially in generative tasks, the mapping between different input modalities is typically achieved through alignment in latent space, as seen in models like CLIP [52] for text-image alignment. Principal component analysis (PCA) [59] is widely used to compress the data in any modality into low-dimensional features by linear dimensionality reduction techniques. To improve the efficiency and compactness, researchers have explored a non-linear approach by learning a neural network for each specific data modality: Variational autoencoder (VAE) [27] is now a prototype neural architecture to project 2D visual data such as image [68] or video [84] into the latent space under an encoder-decoder framework supervised by reconstruction loss and KL-divergence loss [2]. Such VAE framework is also applicable to 1D audio [39] or 3D volume data [3] realized by multi-layer perceptrons (MLP) [1] or 3D convolutional neural networks (3DCNN) [67], respectively. To model the latent space for the unstructured 3D point clouds, PointNet [49, 50] and PointTransformer [83] were two representative architectures leveraging MLP layers and attention mechanisms, respectively. In the recent generative AI paradigm, many efforts have been made to combine all those modality in the sharable latent space, considering them as unique tokens to augment the generation quality and generation spectrum by multimodal learning [72, 73]. While 3DGS is emerging 3D representation, no prototype model for its latent space modeling has yet been actively explored where the application of existing 3D-based VAE highly suffers from the significant unstructuredness of 3DGS mentioned above.

Object-Level 3D Generation. Object-level 3D content generation has achieved significant success in many applications, *e.g.*, 3D mesh generation [16] and texturing [42], and point-cloud generation [45]. With the tremendous progress made by 2D diffusion models, more diverse and high-fidelity 3D generation emerges by combining implicit 3D representations [44] with 2D- or 3D-aware diffusion priors [17, 20, 37, 48, 55, 58]. Several follow-up works [65, 66, 76] use 3DGS as an alternative to NeRF for faster rendering. More impressively, transformer-based architectures [13, 19, 21–24, 70] significantly boost the speed, scalability, and quality of 3D generation [30, 75, 85] and token prediction [6, 60] without any online optimization. While impressive, all those methods are specially designed for a complete object whose application to a general 3D scene often fail due to the complexity gap between objects and scenes e.g. scale-inconsistency and partial observations, which lead researchers to explore the generative framework uniquely designed for a general scene.

Scene-Level 3D Generation. Scene-level 3D generation usually refers to large-scale or even unbounded 3D content creation. One exceptional example is perpetual view generation [31, 36], which allows for view-consistent video or 3D generation with an arbitrary long camera trajectory. Besides these auto-regressive approaches, some efforts [9, 15, 18, 33, 77, 78], enhance global consistency by directly incorporating various 2.5D or 3D cues such as depth, normal, and point cloud into 3D representations such as meshes, NeRFs, or 3DGS for rendering. Though impressive, these methods mostly inherit cumbersome optimization in an iterative manner, Moreover, all recent advancements in scene-level 3D generation rely on 2D supervision, either from images or 2D diffusion models, making feed-forward generation unfeasible without a low-dimensional

embedding for large-scale 3D representations. In this work, we propose Can3Tok for scalable 3DGS embedding that be directly combined with widely-used diffusion architectures for feed-forward 3D generation without image-space supervision.

3. Our Approach

3.1. 3DGS Preliminaries

Given multi-view images with associated camera parameters, a 3D scene depicted in the images can be represented by a set of elliptical 3D primitives, each with an internal radiance field that follows a Gaussian distribution. An individual 3D Gaussian primitive has several parameters, including its 3D center $\mathbf{x} \in \mathbb{R}^3$, rotation represented by a quaternion $\mathbf{r} \in \mathbb{R}^4$, opacity $o \in \mathbb{R}^1$, scaling $\mathbf{s} \in \mathbb{R}^3$, view-independent RGB color $c \in \mathbb{R}^3$, and view-dependent color with high-dimensional spherical harmonics $c_h \in \mathbb{R}^h$.

3.2. Can3Tok Design

Can3Tok is a transformer-based VAE architecture comprising an encoder and a decoder. The encoder maps the tokenized 3DGS to a latent space, and the the decoder reconstructs the original input 3DGS data. The design of our VAE model is illustrated in Fig. 3.

Encoder. The inputs to the encoder is a set of 3DGS data. Similarly to the positional encoding in NeRF [43], we apply Fourier positional encoding on 3D Gaussian centers $\gamma(\mathbf{x}) : \mathbb{R}^{N\times 3} \rightarrow \mathbb{R}^{N\times L_B}$ with a pre-set maximum band L_B to better capture high-frequency components in a low-dimensional embedding. While theory and experiments [44, 64] have validated the effectiveness of this positional encoding approach on MLP-based networks, it has also proven effective across a broader range of architectures including transformers [6, 21, 22, 60, 85]. Since the 3DGS representation is unstructured, we append more structured "anchors" as representative locations to each 3D Gaussian, to reduce the burden of the encoder representation learning: We build a volume on the space of 3D Gaussians with resolution V^3 and apply the same Fourier positional encoding on the voxel center nearest to each 3DGS position $\gamma(\mathbf{v}) : \mathbb{R}^{N\times 3} \rightarrow \mathbb{R}^{N\times L_B}$. Please also refers to our supplementary materials for the effect of nearest voxel coordinates appending. While appending voxel coordinates is not crucial to Can3Tok's success, we find that it enhances output quality, particularly in level-of-detail.

The encoder takes per-scene 3DGS information $\mathcal{G} \in \mathbb{R}^{N\times(2\times L_B+C)}$ including $\gamma(\mathbf{x})$, $\gamma(\mathbf{v})$, and other 3D Gaussian parameters, where N is the number of Gaussians per each scene and C is the size of 3DGS feature dimension. The encoder starts with a linear layer that maps the inputs into key and $value$, and they are tokenized by a cross-attention that takes in the key, $value$, and a learnable canonical $query$, inspired by PerceiverIO [21]. This is because an input scene has more than 10k 3D Gaussians, making the naive self-attention computationally expensive. Importantly, the canonical query is initialized with regular voxel grids and associated descriptors, *i.e.,* $query \in \mathbb{R}^{M\times(P+Q)}$ where M is the number of canonical voxels, P denotes voxel's position, and Q is the size of the descriptors, and further optimized during the training of Can3Tok. The subsequent 8 blocks of self-attention are applied to the tokens to explore the affinities or any other relationships among latents while preserving its dimension unchanged.

Latent space. Following the common VAE design [27], the outputs from our encoder are projected into two latent vectors representing mean μ and log-variance $\log \boldsymbol{\sigma}^2$. The corresponding embedding $\mathbf{z}$ is sampled with the VAE reparameterization trick:

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\epsilon} * \exp(0.5 * \log \boldsymbol{\sigma}^2) \quad (1)$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is sampled from a normal distribution.

Decoder. As shown in Fig. 3, the decoder takes latent samples $\mathbf{z}$ and recovers 3DGS parameters GS_{output} through a linear layers and 16 blocks of self-attention. Unlike PerceiverIO [22], which is designed for the prediction on a discrete target domain, our decoder does not have a pre-defined output query as we aim for the reconstruction in continuous 3D space. To this end, the decoder's tail includes multiple linear layers with non-linear activation function for mapping a latent space into 3D continuum. Although the multi-layer perceptron at the end of the Can3Tok decoder has only a limited number of learnable parameters, both its inputs and outputs are within a bounded space, making the latent-to-3D mapping feasible and less computationally burdensome.

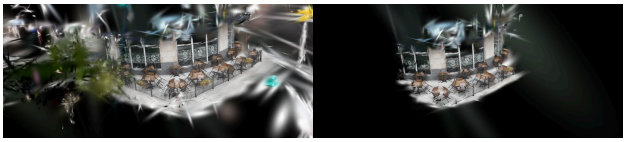
Training Objective. We follow the common setting of training a VAE [27]. The objective for our model optimization is to minimize the following loss:

$$\mathcal{L} = \text{Dist}(GS_{\text{output}}, GS_{\text{input}}) + \lambda \mathcal{L}_{\text{KL}}(\mathbf{z}, \mathcal{N}(\mathbf{0}, \mathbf{I})) \quad (2)$$

where GS_{input} are input 3D Gaussian representations and the scalar λ balances two losses: Dist measures the $L2$ distance between the recovered 3DGS and the ground truth 3DGS across all different feature channels, and $\mathcal{L}_{\text{KL}}$ is the KL divergence between the latent space $\mathbf{z}$ and a normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$ so as to have a structured distribution.

3.3. 3DGS Processing

A fundamental challenge in scaling up VAE models lies in the scale inconsistency across different 3DGS scene representations as we found that none of the existing methods, including our Can3Tok, can generalize well to a large number



Raw 3DGS inputs After Semantics-aware filtering

Figure 4. Before and after our semantic-aware 3DGS filtering.

of scene-level 3DGS representations. Since neither global scene scales or scaling factors of each 3D Gaussian primitive are not metric as the using of COLMAP [56] for camera pose estimation and 3D SfM point triangulation, we propose to unify 3DGS representations into a bounded scale for scaling up training and semantic-aware filtering for improving the reconstruction quality of model outputs.

Normalization. Since there are no established techniques for scalable 3D Gaussian representation learning, we take inspiration from 2D image representation learning [41, 71], where the size of all input images are the same and their RGB channels are normalized into a bounded coherent scale (*e.g.*, [-1,1]) has proven effective for accelerating model convergence and generalization. Specifically, to apply normalization to 3DGS data, we mean-shift the 3DGS centers $\mathbf{x}$ to the origin of world space, and bound all 3DGS into a sphere with radius r while we also re-scale the scaling factor $\mathbf{s}$ of each 3DGS to become $\hat{\mathbf{s}}$:

$$\begin{aligned} \text{translate} &= -\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i, \\ \text{scale} &= \frac{r}{\max|\mathbf{x}+\text{translate}|_2 * 1.1}, \\ \hat{\mathbf{x}} &= (\mathbf{x}+\text{translate}) * \text{scale}, \\ \hat{\mathbf{s}} &= \mathbf{s} * \text{scale}, \\ \hat{\mathbf{T}}_i &= (\mathbf{T}_i+\text{translate}) * \text{scale}, \end{aligned} \tag{3}$$

where $\mathbf{T}_i$ and $\hat{\mathbf{T}}_i$ are center locations of cameras that associate to the training views of 3DGS scene representations. Unlike the RGB channels of 2D images, we preserve all other 3DGS attributes during normalization, as they are heterogeneous and retain their physical meaning only within their original numerical range. The intuition behind Eq. 3 is that: If the same transformation is applied to the camera centers associated with the same 3DGS representations, transforming $\mathbf{T}_i$ into $\hat{\mathbf{T}}_i$ while keeping the camera orientation unchanged, then images rendered from the 3DGS before normalization, using the original camera view with center position $\mathbf{T}_i$, will be identical to those rendered from the 3DGS after normalization, using the same camera view but with the re-scaled position $\hat{\mathbf{T}}_i$. Therefore, another benefit of the proposed 3DGS normalization is that we can recover the metric scales of scenes by estimating metric depth on images rendered from our generated outputs using depth foundation models.

3DGS training data without filtering Can3Tok outputs

Figure 5. Noise-dominated 3DGS training data hurts the latent-space modeling of Can3Tok where meaningful local details largely collapse, which motivate us to have clean scene-level 3DGS data with semantics-aware filtering.

Semantic-aware Filtering. 3DGS reconstructions from general scenes often contain noise artifacts like floaters due to the lack of visual observations (unlike objects which are normally captured with sufficient views). Though Can3Tok can effectively compress 3DGS inputs as latent representations, we experimentally found that such noise deteriorates the latent representation where high-frequency details are washed out in the decoded 3DGS output as shown in Fig. 5. To address this issue, we apply semantic-guided filtering to the raw 3DGS input to subsample as-clean-as-possible 3DGS primitives. Specifically, we apply LangSam [28], a text-guided variant of segment anything model, on the middle frame of each scene video with the text prompt "the most salient region". LangSam crops out the most semantically meaningful region depicted in the image. We pick one Gaussian within the segmentation mask and incrementally include more Gaussians based on a K-NN algorithm in 3D space until reaching a pre-set number N. As shown in Fig. 4, such semantic filtering can preserve the most semantically meaningful contents while removing the less salient and noisy Gaussians.

4. Experiments

4.1. Implementation Details

Dataset: We run 3DGS on all videos from DL3DV-10K dataset [35] (with 6:1 training/testing split), where camera positions and SfM points are obtained by COLMAP [57] for 3DGS initialization. We set $N = 100K$ for each scene representation by applying an upper bound on Gaussian densification and pruning during per-scene optimization.
Data Augmentation: We apply random SO(3) rotations to input 3DGS representations to get more GS_{input} as a way of data augmentation, similarly to the common random-rotation augmentation on 2D images.
Architecture: We implement our encoder with 1 linear layer, 1 cross-attention block, 8 self-attention blocks, and 2 linear layers for mapping latents at the bottleneck into mean and log-variance. Our decoder starts with one linear layer followed by 16 self-attention blocks and ends with 3 linear layers. Self- and cross-attention blocks are of multi-head with 12 heads and 64 dimensions each, implemented using Flash-Attention [11]. Layer normalization is appended

Table 1. Quantitative comparison on DL3DV-10K testing set (with filtering). L_2 error measures the distance between each pair of GS_{output} and GS_{input} over the test set. Failure rate is the percentage of cases where the model completely fails to reconstruct the input 3D Gaussians.

	L_2 error ↓	Failure rate↓
L3DG [53]	1200.4	100%
PointNet VAE [50]	1823.0	100%
PointTtransformer [83]	230.7	70%
Ours	**30.1**	**2.5%**

Table 2. Quantitative comparisons and ablation studies on DL3DV-10K testing set. L_2 error measures the distance between each pair of GS_{output} and GS_{input} over the test set. Failure rate is the percentage of cases where the model completely fails to reconstruct the input 3D Gaussians.

	L_2 error ↓	Failure rate↓
Ours (w/o Learnable Query)	10^{25}	100%
Ours (w/o normalization)	1889.7	100%
Ours (w/o voxel appending)	50.5	4.3%
Ours (w/o data filtering)	73.3	6.1%
Ours (w/o data augmentation)	53.3	4.6%
Ours (full)	**30.1**	**2.5%**

to each linear layer and attention block. Latent query has size $\mathbf{Q} \in \mathbb{R}^{256\times768}$. Mean μ, log-variance $\log\sigma^2$, and $\mathbf{z}$ are in $\in \mathbb{R}^{64\times64\times4}$, which has exactly the same size as the latent space of Stable Diffusion [54]. The output from Fourier positional encoding has size $L_B = 51$, specifically $\gamma(\mathbf{x}) : \mathbb{R}^{N\times3} \rightarrow \mathbb{R}^{N\times51}$. Input volume has resolution $V = 40$. We set the loss hyper-parameters to $\lambda = 1\times10^{-6}$. Each scene has the same number $N = 40K$ of Gaussians after semantics-aware filtering. We train our model on 8 A100 GPUs for 5 days. A single forward pass for encoding and decoding an input 3D scene with our model takes only $\sim$0.06 sec, making it compatible with a diffusion module for feedforward generation, as we demonstrate later.

4.2. Baselines and Metrics

We compare our VAE model to existing 3D-based VAE models. L3DG [53] is a recent method for object-level 3DGS encoding and decoding. We implement L3DG and other convolution-based architectures with Minkowski Engine [8] and spconv [10] following L3DG paper. We also compare with a PointNet-based autoencoder [50] with more network layers to increase its model capacity. Another baseline method is PointTransformer [83], which is a popular transformer-based architecture. All compared methods are trained and tested on the same train and test sets from DL3DV-10K [35] with the same data processing as training our Can3Tok, to evaluate if they can accurately generalize to unseen input 3DGS. Since 3D Gaussians have more parameters than simple location coordinates, we use average L_2-norm across all 3DGS features between each pair of GS_{output} and GS_{input} on the test set as an evaluation metric. Moreover, we measure the failure rate, defined as the percentage of cases in which the model completely fails to reconstruct the input 3D Gaussians, as shown in Fig. 6 and more qualitative results in the supplementary material.

4.3. Results

In our experiments, PointNet and L3DG cannot even converge with more than 500 3D scenes while PointTransformer performs slightly better but still produces poor visual quality. As shown in Tab. 1, these methods fail to generalize to unseen 3D scenes with 100% failure rate, while our design shows great potential for its scalability with low L_2 error and failure rate. The failure rate is defined as the percentage of scenes whose L_2 error of reconstruction exceeds 1000.0, which are completely not recognizable as shown in Fig. 6. Our model successfully reconstructs original input 3DGS, and neither convolution- or transformer-based methods are able to succeed in decoding the 3DGS scenes by losing the original global shape and local details. While PointTransformer [83] outperforms L3DG and PointNet in terms of L_2 error and failure rate, the reconstructions show stretched and distorted patterns on all test samples. Qualitative results shown in Fig. 6 and supplementary materials are general and not cherry picked. This failure mode observed in convolution-based VAEs is also highlighted in the concurrent work Bolt3D [63].

4.4. Latent-Space Analysis

It remains inconclusive why our proposed method genuinely captures structured and meaningful 3D geometric patterns within the representation instead of merely memorizes the input 3DGS. This question cannot be fully answered by quantitative metrics and qualitative comparisons in spatial domain.

Spatial Encoding: Therefore, we highlight our structured latent space by exploring the spatial relationship between inputs and the associated latent embeddings via t-SNE [69] visualization shown in Fig. 7. We encode a 3DGS scene with different SO(3) rotations into latent embeddings in inference stage. Even though no explicit constraints on latent space were applied during training, our method automatically discovers the spatial correlation between inputs and latent embeddings. While other baseline methods mix up the same scene under different 3D orientations and other scenes in latent space. Therefore the decoder is fail to correctly decode latents back to different 3DGS representations.

Semantic Encoding: We also highlight the ability of our latent representation to abstract semantic information of inputs instead of merely memorizing all 3D Gaussians. In Fig. 8, subfigures are two different 3D Gaussian filterings

Figure 6. Qualitative comparisons between ours and other VAE outputs. Results are not cherry picked as all compared methods show zero generalization ability on novel scene inputs. PointNet and L3DG do not converge on training set.

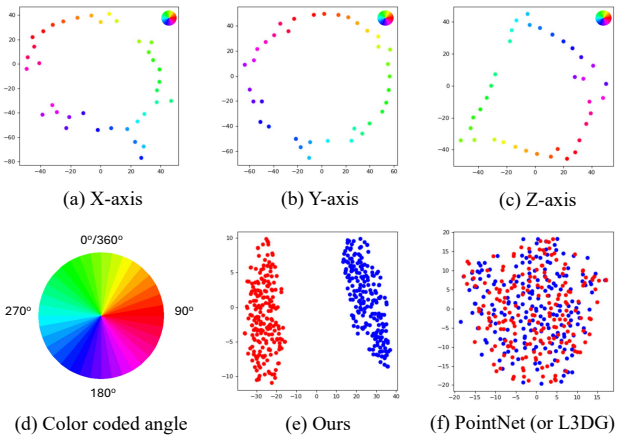


Figure 7. t-SNE visualizations of the latent space of 3DGS for the same scene with 36 linearly interpolated SO(3) rotations from 0 to 360 degrees. All three rotations exhibit patterns of closed loops, demonstrating that our model preserves spatial information in the latent representations. In (e) and (f), red dots are latent embeddings of the same scene but with 200 random SO(3) rotations and blue dots are latent embeddings of different scenes.

or croppings from raw 3D Gaussians but covering the same 3D contents (some outdoor chairs and desks on the ground together with a wall), while two samples are from a completely different scene. In latent space, latent embeddings from the same scenes are close to each other (black dots), and along with other different scenes (gray dots) are far away. Additionally, the different SO(3) rotations of the same scene can be close to each other in latent space as shown in Fig. 7 as they are semantically similar.

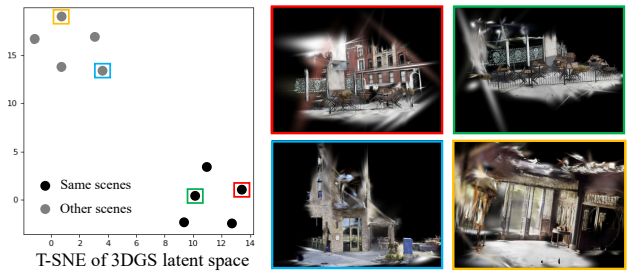


Figure 8. Given 3DGS reconstruction for various scenes, we randomly subsample 3DGS similar to the semantics-aware filtering in Sec. 3.3; and demonstrate the t-SNE of their latent space. The latent spaces from same scenes, visualized in red and green images, are closer each other, and otherwise for other scenes.

4.5. Ablation Study

We study the effectiveness of each proposed module and technique. Please refer to Tab. 2 for the quantitative ablations. Also, we describe the qualitative results and the detailed settings of ablation study in the supplementary materials, which include 1) without learnable query, 2) no 3DGS normalization, 3) no 3DGS data filtering, 4) no voxel coordinates appending, 5) no data augmentation.

In Fig. 5, We demonstrate that semantic-aware data filtering is essential for capturing high-frequency details of 3DGS in the latent space, as it mitigates the negative impact of training a VAE model with imperfect 3DGS reconstruction results. Since learning raw 3DGS with floaters and noise deteriorates the high-frequency correlation between the input 3DGS and the latent space. Consequently,

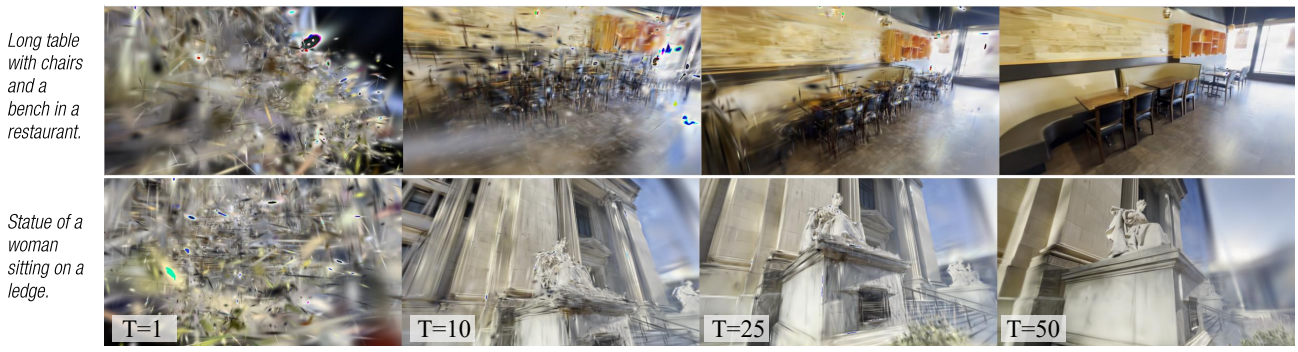


Figure 9. Two examples of text-conditioned generation with a latent diffusion UNet, which is trained on our 3DGS embeddings of scenes and corresponding text prompts. The scene labels are shown in the captions, and the images are rendered with camera positions around the generated 3DGS by denoising the latents $\mathbf{z}$ passing through our Can3Tok decoder. T denotes the denoising time step. Please note that, while the images are rendered from a fixed camera viewpoint, the global 3DG structures are shifting during the denoising process, which lead to the viewpoint-shifting sensation.

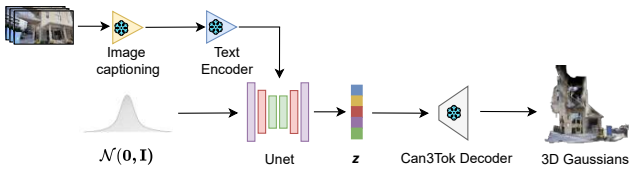


Figure 10. A pipeline for text-conditioned latent diffusion model for 3D Gaussian generation using Can3Tok latent embeddings. The pretrained image-captioning, text-encoder and Can3Tok decoder modules are frozen during training the denoising UNet model. Snow labels denote the pre-trained models.

without the proposed data filtering, L_2 increases, as shown in Tab. 2. More importantly, while Can3Tok is the only method capable of converging on thousands of training samples, it fails to generalize to unseen test data without 3DGS normalization (w/o normalization). This reveals that a well-designed model alone cannot address the unique scale-inconsistency issue in scene-level 3D representations.

4.6. Application

We showcase that our 3DGS latent embedding could be useful for various generative tasks including text-guided and image-guided 3DGS generations.

Text-to-3DGS Generation. As shown in Fig. 10, we train a diffusion UNet for mapping a noise vector sampled from a normal distribution into a meaningful 3D Gaussian embedding $\mathbf{z}$, conditioned on text prompts. Since the DL3DV-10K dataset lacks labels or scene text descriptions for scenes, we caption the middle frame of each video to label each scene where we use a pre-trained BLIP model [29] whose output text prompts are consistently concise and capture salient semantic information. After this labeling, we train the diffusion model on $(\text{text label}, \mathbf{z})$ pairs. At inference, the UNet samples a $\mathbf{z}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ from normal distribution and tries to denoise it with T steps to approach $\mathbf{z}_0 \equiv \mathbf{z}$, conditioned on a text scene label. We showcase the denoising process of two inference examples in Fig. 9.

Image-to-3DGS Generation. In the supplementary document, we also showcase that our 3DGS latent embedding can be combined with existing image regression modules [4] to enable image-guided 3DGS generation. To this end, we train the encoder that regresses the image to our 3DGS latent space, and we use Can3Tok decoder to construct the associated 3DGS outputs. Please refer to the supplementary for the visual results.

4.7. Limitations

As shown in 2, our method did not achieve a 100% success rate. This is due to some low-quality 3DGS reconstructions in the training set. We observed that some videos used for 3DGS reconstruction suffer from severe motion blur and an imbalanced distribution of close-up and distant views during data capture. As a result, the corresponding latent representations become less discriminative and tend to mix with those of other 3D scenes. Besides, our method is limited to the 3DGS representation, as it is more discrete and suited for tokenization than other neural representations.

5. Conclusion

We introduce Can3Tok, the first method for scene-level 3DGS latent representations, demonstrating that all existing approaches fail without a model design and proper 3D data normalization specifically tailored to the 3DGS representation. Through latent space analysis, qualitative and quantitative comparisons, we show that our method significantly outperforms existing 3D VAE models. Additionally, we propose a 3DGS data processing approach to address the open problem of scale inconsistency in 3D representations. To further enhance quality and scalability, we introduce semantic-aware filtering and data augmentation. Finally, we showcase the practical utility of Can3Tok in 3D generative applications, including text-guided and image-guided 3D generation with 3DGS representation.

6. Acknowledgment

We thank Kai Zhang and Zexiang Xu for the insightful discussion.

References

[1] Luis B Almeida. Multilayer perceptrons. In *Handbook of Neural Computation*, pages C1–2. CRC Press, 2020. 3

[2] Dmitry I Belov and Ronald D Armstrong. Distributions of the kullback–leibler divergence with applications. *British Journal of Mathematical and Statistical Psychology*, 64(2): 291–309, 2011. 3

[3] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Generative and discriminative voxel modeling with convolutional neural networks. *arXiv preprint arXiv:1608.04236*, 2016. 3

[4] Clément Chadebec, Louis Vincent, and Stephanie Allassonniere. Pythae: Unifying generative autoencoders in python - a benchmarking use case. In *Advances in Neural Information Processing Systems*, pages 21575–21589. Curran Associates, Inc., 2022. 8

[5] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[6] 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. 3, 4

[7] Kai Cheng, Xiaoxiao Long, Kaizhi Yang, Yao Yao, Wei Yin, Yuexin Ma, Wenping Wang, and Xuejin Chen. Gaussianpro: 3d gaussian splatting with progressive propagation. In *Forty-first International Conference on Machine Learning*, 2024. 2

[8] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3075–3084, 2019. 6

[9] Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023. 1, 3

[10] Spconv Contributors. Spconv: Spatially sparse convolution library. https://github.com/traveller59/spconv, 2022. 6

[11] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 5

[12] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 2

[13] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3

[14] Zhiwen Fan, Wenyan Cong, Kairun Wen, Kevin Wang, Jian Zhang, Xinghao Ding, Danfei Xu, Boris Ivanovic, Marco Pavone, Georgios Pavlakos, et al. Instantsplat: Unbounded sparse-view pose-free gaussian splatting in 40 seconds. *arXiv preprint arXiv:2403.20309*, 2024. 2

[15] Rafail Fridman, Amit Abecasis, Yoni Kasten, and Tali Dekel. Scenescape: Text-driven consistent scene generation. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[16] 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. 1, 3

[17] Junlin Han, Filippos Kokkinos, and Philip Torr. Vfusion3d: Learning scalable 3d generative models from video diffusion models. In *European Conference on Computer Vision*, pages 333–350. Springer, 2025. 3

[18] Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7909–7920, 2023. 3

[19] 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. 1, 3

[20] Hanzhe Hu, Zhizhuo Zhou, Varun Jampani, and Shubham Tulsiani. Mvd-fusion: Single-view 3d via depth-consistent multi-view generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9698–9707, 2024. 3

[21] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. *arXiv preprint arXiv:2107.14795*, 2021. 3, 4

[22] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021. 4

[23] Hanwen Jiang, Zhenyu Jiang, Yue Zhao, and Qixing Huang. Leap: Liberate sparse-view 3d modeling from camera poses. *arXiv preprint arXiv:2310.01410*, 2023.

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

[25] 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. 1, 2

[26] Bernhard Kerbl, Andreas Meuleman, Georgios Kopanas, Michael Wimmer, Alexandre Lanvin, and George Drettakis.

A hierarchical 3d gaussian representation for real-time rendering of very large datasets. *ACM Transactions on Graphics (TOG)*, 43(4):1–15, 2024. 2

[27] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3, 4

[28] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 2, 5

[29] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pages 12888–12900. PMLR, 2022. 8

[30] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*, 2023. 3

[31] Zhengqi Li, Qianqian Wang, Noah Snavely, and Angjoo Kanazawa. Infinitenature-zero: Learning perpetual view generation of natural scenes from single images. In *European Conference on Computer Vision*, pages 515–534. Springer, 2022. 3

[32] Zhihao Liang, Qi Zhang, Wenbo Hu, Ying Feng, Lei Zhu, and Kui Jia. Analytic-splatting: Anti-aliased 3d gaussian splatting via analytic integration. *arXiv preprint arXiv:2403.11056*, 2024. 2

[33] Chieh Hubert Lin, Hsin-Ying Lee, Willi Menapace, Menglei Chai, Aliaksandr Siarohin, Ming-Hsuan Yang, and Sergey Tulyakov. Infinicity: Infinite-scale city synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22808–22818, 2023. 3

[34] Jiaqi Lin, Zhihao Li, Xiao Tang, Jianzhuang Liu, Shiyong Liu, Jiayue Liu, Yangdi Lu, Xiaofei Wu, Songcen Xu, Youliang Yan, et al. Vastgaussian: Vast 3d gaussians for large scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5166–5175, 2024. 2

[35] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22160–22169, 2024. 2, 5, 6

[36] Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14458–14467, 2021. 3

[37] 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. 3

[38] 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

[39] Yin-Jyun Luo, Kat Agres, and Dorien Herremans. Learning disentangled representations of timbre and pitch for musical instrument sounds using gaussian mixture variational autoencoders. *arXiv preprint arXiv:1906.08152*, 2019. 3

[40] Qi Ma, Yue Li, Bin Ren, Nicu Sebe, Ender Konukoglu, Theo Gevers, Luc Van Gool, and Danda Pani Paudel. Shapesplat: A large-scale dataset of gaussian splats and their self-supervised pretraining. *arXiv preprint arXiv:2408.10906*, 2024. 3

[41] Jesús Malo. Normalized image representation for efficient coding. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, pages 1408–1412. IEEE, 2003. 5

[42] Oscar Michel, Roi Bar-On, Richard Liu, Sagie Benaim, and Rana Hanocka. Text2mesh: Text-driven neural stylization for meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13492–13502, 2022. 3

[43] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 4

[44] 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, 3, 4

[45] 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. 1, 3

[46] Michael Niemeyer, Fabian Manhardt, Marie-Julie Rakotosaona, Michael Oechsle, Daniel Duckworth, Rama Gosula, Keisuke Tateno, John Bates, Dominik Kaeser, and Federico Tombari. Radsplat: Radiance field-informed gaussian splatting for robust real-time rendering with 900+ fps. *arXiv preprint arXiv:2403.13806*, 2024. 2

[47] 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. 1

[48] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 1, 3

[49] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016. 3

[50] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1, 2, 3, 6

[51] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017. 2

[52] 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. 3

[53] Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulò, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3dg: Latent 3d gaussian diffusion. *arXiv preprint arXiv:2410.13530*, 2024. 1, 3, 6

[54] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 6

[55] 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. 1, 3

[56] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[57] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 5

[58] 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. 3

[59] Jonathon Shlens. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, 2014. 3

[60] Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19615–19625, 2024. 3, 4

[61] Stanislaw Szymanowicz, Eldar Insafutdinov, Chuanxia Zheng, Dylan Campbell, Joao F Henriques, Christian Rupprecht, and Andrea Vedaldi. Flash3d: Feed-forward generalisable 3d scene reconstruction from a single image. *arXiv preprint arXiv:2406.04343*, 2024. 2

[62] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10208–10217, 2024. 2

[63] Stanislaw Szymanowicz, Jason Y Zhang, Pratul Srinivasan, Ruiqi Gao, Arthur Brussee, Aleksander Holynski, Ricardo Martin-Brualla, Jonathan T Barron, and Philipp Henzler. Bolt3d: Generating 3d scenes in seconds. *arXiv preprint arXiv:2503.14445*, 2025. 6

[64] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020. 4

[65] 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. 1, 3

[66] 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 *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 3

[67] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 3

[68] Arash Vahdat and Jan Kautz. NVAE: A deep hierarchical variational autoencoder. In *Neural Information Processing Systems (NeurIPS)*, 2020. 3

[69] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9 (11), 2008. 6

[70] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 3

[71] Yi Wang, Ying-Cong Chen, Xiangyu Zhang, Jian Sun, and Jiaya Jia. Attentive normalization for conditional image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5094–5103, 2020. 5

[72] Yikai Wang, Xinghao Chen, Lele Cao, Wenbing Huang, Fuchun Sun, and Yunhe Wang. Multimodal token fusion for vision transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12186–12195, 2022. 3

[73] Zekun Wang, King Zhu, Chunpu Xu, Wangchunshu Zhou, Jiaheng Liu, Yibo Zhang, Jiashuo Wang, Ning Shi, Siyu Li, Yizhi Li, et al. Mio: A foundation model on multimodal tokens. *arXiv preprint arXiv:2409.17692*, 2024. 3

[74] Olivia Wiles, Georgia Gkioxari, Richard Szeliski, and Justin Johnson. Synsin: End-to-end view synthesis from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7467–7477, 2020. 2

[75] Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Peng Wang, Jiahao Li, Zifan Shi, Kalyan Sunkavalli, Gordon Wetzstein, Zexiang Xu, et al. Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model. *arXiv preprint arXiv:2311.09217*, 2023. 3

[76] 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. 3

[77] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. *arXiv preprint arXiv:2406.09394*, 2024. 3

[78] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6658–6667, 2024. 1, 3

[79] Xianggang Yu, Mutian Xu, Yidan Zhang, Haolin Liu, Chongjie Ye, Yushuang Wu, Zizheng Yan, Chenming Zhu, Zhangyang Xiong, Tianyou Liang, et al. Mvimgnet: A large-scale dataset of multi-view images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9150–9161, 2023. 2

[80] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19447–19456, 2024. 2

[81] 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. 3

[82] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19. Springer, 2024. 2

[83] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 16259–16268, 2021. 3, 6

[84] Sijie Zhao, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Muyao Niu, Xiaoyu Li, Wenbo Hu, and Ying Shan. Cv-vae: A compatible video vae for latent generative video models. *arXiv preprint arXiv:2405.20279*, 2024. 3

[85] 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, 4