ScenDi: 3D-to-2D Scene Diffusion Cascades for Urban Generation

Hanlei Guo[1], Jiahao Shao[1], Xinya Chen[1], Xiyang Tan[3], Sheng Miao[1], Yujun Shen[2], Yiyi Liao[1]✉
[1] Zhejiang University [2] Ant Group [3] The University of British Columbia

Project Page: https://xdimlab.github.io/ScenDi_website/

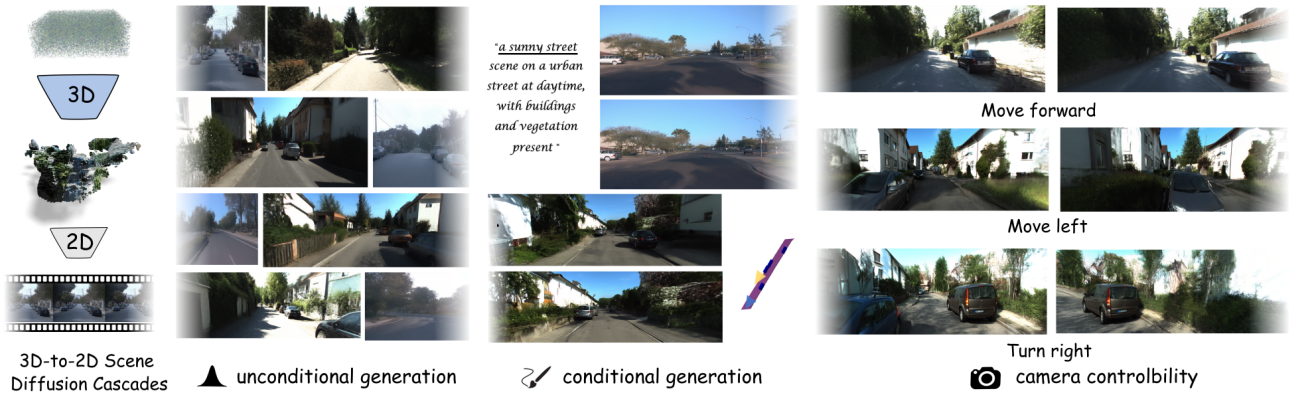

Figure 1. **ScenDi** generates high-quality urban scenes using a 3D-to-2D Scene Diffusion cascade, with optional condition signals like text and layout for controllable 3D space generation. Our method provides flexible camera control, even though our training data primarily consists of forward-moving trajectories.

Abstract

Recent advancements in 3D object generation using diffusion models have achieved remarkable success, but generating realistic 3D urban scenes remains challenging. Existing methods relying solely on 3D diffusion models tend to suffer a degradation in appearance details, while those utilizing only 2D diffusion models typically compromise camera controllability. To overcome this limitation, we propose ScenDi, a method for urban scene generation that integrates both 3D and 2D diffusion models. We first train a 3D latent diffusion model to generate 3D Gaussians, enabling the rendering of images at a relatively low resolution. To enable controllable synthesis, this 3DGS generation process can be optionally conditioned by specifying inputs such as 3d bounding boxes, road maps, or text prompts. Then, we train a 2D video diffusion model to enhance appearance details conditioned on rendered images from the 3D Gaussians. By leveraging the coarse 3D scene as guidance for 2D video diffusion, ScenDi generates desired scenes based on input conditions and successfully adheres to accurate camera trajectories. Experiments on two challenging real-world datasets, Waymo and KITTI-360, demonstrate the effectiveness of our approach.

✉ Corresponding author.

1. Introduction

Generating 3D urban scenes from scratch, either unconditionally or with coarse guidance such as text prompts or layout maps, is an important step toward building open-world virtual environments for gaming and driving simulation. Unlike image-to-video synthesis [18, 53], which extends an input view given viewpoints, 3D scene generation aims to create entire environments with consistent geometry and realistic appearance that can be rendered from free viewpoints. Such capability is crucial for enhancing data diversity, supporting scene composition, and enabling scalable scene generation.

A majority line of work addresses this task by directly performing generation in the 3D space [3, 26, 33, 79]. While such approaches offer explicit spatial modeling, their performance is constrained by the low resolution of 3D representations. Besides, generating 3D urban scenes that directly render to high-fidelity images is challenging due to the scarcity of such 3D GT data. More recently, another line of methods [14, 28, 39, 41, 87] attempt to enhance the performance by first generating 3D semantic voxels, and then render 2D conditions (e.g., depth or semantic maps) for video generation, allowing for producing high-fidelity details thanks to the powerful pre-trained 2D models. While this approach benefits from 3D structure, the final appearance is still entirely generated in 2D, requiring the model to

learn a complex mapping from depth or semantics to RGB images, which can make training less efficient and lack consistency when the same place is revisited. This raises a fundamental question: How much of the generation process should occur in 3D space, and how much should be delegated to 2D?

In this work, we propose ScenDi[1], a cascaded 3D-to-2D *Scene Di*ffusion framework that leverages the complementary strengths of both modalities. Our key insight is that the 3D stage should produce not only geometric structure but also coarse RGB appearance, a distinction from prior methods [14, 28, 41] that use 3D to solely provide geometric cues, forcing the 2D model to synthesize all appearance from scratch. Refining this coarse 3D appearance with a 2D model improves both training efficiency and loop consistency. Consequently, our cascaded architecture achieves high-fidelity urban scene synthesis alongside accurate camera controllability.

Our method begins by training a 3D latent diffusion model to generate coarse 3D scenes, which comprises a novel Voxel-to-3DGS VQ-VAE learned from 2D supervision and a 3D diffusion model, leveraging an off-the-shelf depth estimator to form the input voxel grids. This enables sampling 3D Gaussian Splatting (3DGS) scenes that renders to multi-view consistent images. However, rendered images from the 3D LDM often lack high-frequency details due to limited 3D resolution and fail to capture distant regions. To address this, we condition a 2D video diffusion model on the rendered RGB images, leveraging its ability to refine details and synthesize regions beyond predefined ranges. Experimental results on real-world autonomous driving datasets KITTI-360 and Waymo demonstrate that our 3D-to-2D diffusion cascades allow for generating high-quality urban scenes while preserving precise camera control.

2. Related Work

Direct 3D Content Generaion: One line of research focus on generating 3D scenes relying solely on 3D backbones. Early works [3, 7, 73, 75, 79] that adopt GANs [19] suffer from inherent issues such as training instability and model corruption. A growing trend to leverage 3D diffusion models in image on tasks emerged due to their superb training stability. Methods lying on this line enables the availability of explicit 3D representation, thus leads to the camera controllable viewpoints over the generated scenes. Some works [30, 54, 72, 84, 86] construct 3D ground-truth datasets and perform training directly in 3D space. However, the natural scarcity of high-quality 3D scene data, such as 3D Gaussian Splatting [25, 71] that require per-scene optimization, undermines these works' scalability and limits their ability to generalize to diverse real-world scenarios. In contrast, we propose to use a VQ-VAE that directly maps voxel grids obtained from off-the-shelf depth priors to 3D Gaussians. Similar to us, several methods [1, 4, 9, 26, 76] also train 3D diffusion models that predict 3D representations with 2D multi-view image supervisions. Although they can avoid heavy dataset pre-processing, their results often lose fine appearance details. Our method addresses this challenge by using 3D-to-2D diffusion cascades. The last line of methods [11, 32, 61, 69, 77] distills information from pretrained 2D diffusion models to optimize 3D representation based on Score Distillation Sampling (SDS) [49]. These methods require a long training time to generate a single scene and usually suffer from over-saturation.

Multi-view Image Generation: In contrast to direct 3D generation, some works leverage 2D diffusion models for multi-view generation, aiming to exploit the powerful generative prior of 2D diffusion models trained on internet-scale data. Despite the impressive progress in this direction, many methods target only on object-level generation [29, 34, 35, 37, 58, 63]. There are also attempts for street scene image/video generation [17, 18, 67, 70]. While most methods show superior visual fidelity compared with 3D generative methods, they struggle to provide flexible camera control due to the lack of explicit 3D information. Recently, there has been a growing trend towards gradually reconstructing 3D scenes while generating novel views [13, 82, 83]. However, these methods have not yet demonstrated the ability to handle large forward camera motions typical in driving scenarios.

Urban Scene Generation: A line of methods [10, 18, 20, 53, 68, 70, 78] tackles image-to-video generation for urban scenes. Despite promising results, noticeable visual degradation often occurs as the methods extend far from the initial viewpoint. Another set of approaches [16, 67] uses coarse guidance, such as HD maps or bounding boxes, to improve control over the generated scenes. However, these lack an explicit 3D backbone, leading to inter-frame inconsistencies. In contrast, our approach integrates diverse control signals for flexible scene generation while maintaining an explicit 3DGS backbone. The most related methods [14, 28, 39, 41, 47, 87] generate a semantic voxel grid and rely on 2D renderings like depth or semantic maps for appearance synthesis. In comparison, our method directly generates coarse 3DGS and refines them with a 2D model.

3. Method

ScenDi aims to generate high-quality and controllable urban scenes by incorporating both 3D and 2D diffusion models. An overview of our method is presented in Fig. 2, consisting of 3D generation and 2D refinement stages. Our 3D

[1] ScenDi means "descend" in Italian, reflecting our cascaded generation process moving from 3D to 2D.

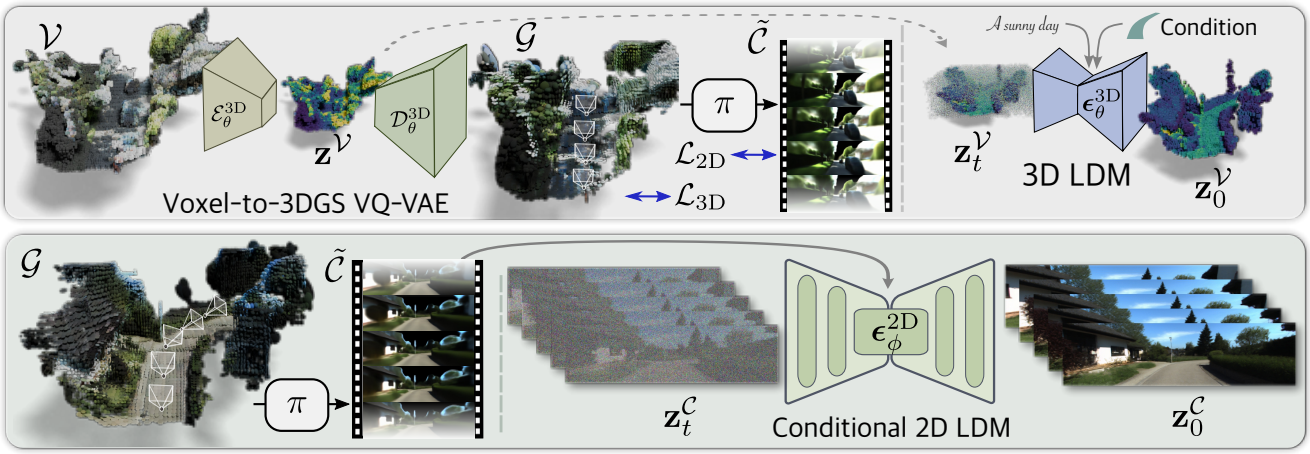

Figure 2. **Method Overview.** `ScenDi` leverages 3D and 2D diffusion cascades to generate high-quality urban scenes. *Top:* We first build a Voxel-to-3DGS VQ-VAE to reconstruct scenes in a feed-forward manner. The input is a colored voxel grid $\mathcal{V}$ constructed based on off-the-shelf metric depth estimator, whereas the output is a set of 3D Gaussian primitives $\mathcal{G}$. Then, we train a 3D diffusion model $\boldsymbol{\epsilon}_\theta^{\text{3D}}$ on the latent space $\mathbf{z}^{\mathcal{V}}$ to generate coarse 3D scenes, optionally conditioning on signals such as road maps or text prompts to enable explicit control over the content. *Bottom:* Based on the coarse 3D scene, we train a 2D video diffusion model to refine foreground appearance details as well as generate distant areas. We achieve this by adopting video clip $\tilde{\mathcal{C}}$ rendered from generated 3DGS as 3D conditional signals to fine-tune a conditional 2D latent diffusion model $\boldsymbol{\epsilon}_\phi^{\text{2D}}$.

generator is a 3D Latent Diffusion Model, aiming to generate coarse geometry and appearance within a pre-defined volume as detailed in Sec. 3.1. Here, we adopt 3D Gaussian primitives [25] as the output 3D representation for its high rendering efficiency. Additionally, this generation process can be optionally controlled by 3D bounding boxes, road maps, and text instructions. Next, we adopt a 2D video diffusion to fill in the details of rendered images from the 3D generation stage and synthesize background contents outside the volume as in Sec. 3.2. The training process is elaborated in Sec. 3.3.

3.1. Scene Generation by 3D Diffusion

Following the standard LDM [55], we design a 3D LDM for generating 3D Gaussian primitives, consisting of a 3D VQ-VAE and a latent-space diffusion model. While there are existing works in this direction for generating occupancy or semantic voxel grids [27, 36, 43, 51, 72], these methods require accurate 3D GT data, i.e. the GT semantic voxel grids are used as input and output of the VQ-VAE. However, it is extremely time-consuming to obtain a large amount of 3D Gaussians for urban scenes using per-scene optimization. Therefore, we propose to learn a voxel-to-3DGS VQ-VAE directly from 2D supervision and pseudo 3D GT obtained from metric monocular depth estimations.

Input Voxel Grid Construction: We first construct a colored voxel grid $\mathcal{V} \in \mathbb{R}^{H\times W\times D\times 3}$ as input to our 3D VQ-VAE. This is achieved by leveraging an metric monocular depth estimator. Specifically, given several posed input images $\left\{\mathbf{I}_i \in \mathbb{R}^{h\times w\times 3}\right\}_{i=1}^{N}$ within a volume of interest, we use a pre-trained depth estimator [23] to infer their depth maps $\left\{\mathbf{D}_i \in \mathbb{R}^{h\times w}\right\}_{i=1}^{N}$. Next, we unproject the depth maps and merge them into a unified coordinate system based on their camera poses. We follow [12] to create a consistency check module to examine inaccurate depth predictions and adopt filters to remove outliers. By merging information from different camera views, we obtain a global RGB point cloud which provides a coarse 3D scene context. This colored point cloud is then discretized into the RGB voxel grid $\mathcal{V}$, where points outside of the voxel grid are discarded. We refer to $\mathcal{V}$ as the foreground, which only covers a limited range of the unbounded urban scene.We project the merged point cloud into each input view to obtain corresponding foreground masks $\{\mathbf{M}_i\}_{i=1}^{N}$, such that our VQ-VAE only needs to reconstruct scenes within the foreground region.

Voxel-to-3DGS VQ-VAE: Taking the voxel grid $\mathcal{V}$ as input, we train a VQ-VAE which maps it into a lower dimensional latent code $\mathbf{z} \in \mathbb{R}^{H'\times W'\times D'\times F'}$ and then decodes it into a set of 3D Gaussian primitives $\mathcal{G}$. More formally, let $\mathcal{E}_\theta^{3D}$ denotes the encoder, $\mathcal{Q}_\theta^{3D}$ the vector quantizer, and $\mathcal{D}_\theta^{3D}$ the decoder of the 3D VQ-VAE, we map the input $\mathcal{V}$ to the Gaussian primitives $\mathcal{G}$ as follows:

$$\mathbf{z}^{\mathcal{V}} = \mathcal{E}_\theta^{\text{3D}}(\mathcal{V}),\ \mathbf{z}_q^{\mathcal{V}} = \mathcal{Q}_\theta^{\text{3D}}(\mathbf{z}^{\mathcal{V}}),\ \mathcal{G} = \mathcal{D}_\theta^{\text{3D}}(\mathbf{z}_q^{\mathcal{V}}) \quad (1)$$

In practice, our decoder $\mathcal{D}_\theta^{3D}$ predicts scene occupancy $\mathbf{O} \in R^{H\times W\times D}$ through an occupancy branch and a feature volume $\mathbf{F} \in R^{H\times W\times D\times F}$ at the same spatial resolution of $\mathcal{V}$ through another feature branch. We denote one Gaussian primitive through MLP heads when a voxel is predicted as

occupied. Each 3D Gaussian primitive is parameterized by the following attributes: color $\mathbf{c}$, opacity α, scale $\mathbf{s}$, rotation $\mathbf{R}$, and an offset to the voxel center $\Delta\mathbf{o}$. Let f_θ^{color}, f_θ^{opa}, f_θ^{geo}, and f_θ^{offset} denote the MLP heads for predicting the Gaussian attributes, and $\mathbf{f} \in \mathbb{R}^F$ denotes a feature vector responding to one voxel of $\mathbf{F}$, we obtain the attributes of one 3D Gaussian primitive from an occupied 3D voxel as follows:

$$\mathbf{c} = f_\theta^{\text{color}}(\mathbf{f}), \quad \alpha = f_\theta^{\text{opa}}(\mathbf{f}), \tag{2}$$

$$\mathbf{s}, \mathbf{R} = f_\theta^{\text{geo}}(\mathbf{f}), \quad \Delta\mathbf{o} = f_\theta^{\text{offset}}(\mathbf{f}) \tag{3}$$

The 3D Gaussian primitives can then be rendered into images through the rendering function π.

Latent 3D Diffusion Model: After training a generalizable feed-forward reconstruction network, we train a 3D diffusion model that operates on the 3D latent space of the VQ-VAE, which allows for generating a 3D latent code from pure noise.

More specifically, we follow the standard forward process of the diffusion model by adding noise to the original clean latent $\mathbf{z}_0$.

$$\mathbf{z}_t^{\mathcal{V}} = \sqrt{\overline{\alpha}_t}\mathbf{z}_0^{\mathcal{V}} + \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}, \ \boldsymbol{\epsilon} \in \mathcal{N}(0,1) \tag{4}$$

where $\overline{\alpha}_t$ controls the level of noise added to the original latent and $\mathbf{z}_t$ is the noisy version of $\mathbf{z}_0$ at time step t. For the reverse process, we train a noise estimator $\boldsymbol{\epsilon}_\theta(\mathbf{z}_t; t)$ for denoising, which follows the standard DiT [48] network architecture for image generation [15]. In practice, we train our diffusion model to perform v-prediction [56]. The predicted clean latent $\hat{\mathbf{z}}_0$ can be derived from the direct output of the network by:

$$\hat{\mathbf{z}}_0^{\mathcal{V}} = \sqrt{\overline{\alpha}_t}\mathbf{z}_t^{\mathcal{V}} - \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}_\theta^{3D}(\mathbf{z}_t^{\mathcal{V}}; t, c) \tag{5}$$

where c indicates optional condition signals described below. During sampling, the model denoises a pure Gaussian noise ϵ over multiple iterations to a clean latent using a standard DDIM sampler [59].

Controllable 3D Scene Generation: To improve controllability, we incorporate optional conditional signals during training of the 3D latent diffusion model. We explore two settings: coarse 3D layout conditioning and text conditioning. For layout conditioning, our 3D generative model directly supports 3D constraints without rendering them into 2D space [47, 67]. Specifically, we build a 3D conditional volume from primitive labels [31], where each voxel contains a one-hot semantic label (e.g., vehicle, road, or empty). This volume is downsampled to the latent resolution and concatenated with the initial latent $\mathbf{z}_0$ for diffusion training. For text-conditioned generation, we follow [46] to annotate each scene with attributes such as weather, road type, and background, forming a descriptive sentence. Text embeddings extracted via [50] are then injected into the model through cross-attention.

3.2. Scene Augmentation by 2D Diffusion

Although images synthesized by 3D LDM excel in maintaining cross-view consistency and disentangling camera poses from contents, they may suffer from blurriness due to the resolution limitation. Additionally, far areas are difficult to model in 3D. Therefore, we use the prior information of the pre-trained 2D model [6, 64] to augment appearance details while completing distant areas.

Conditional Latent 2D Diffusion Model: Inspired by recent diffusion-based super-resolution methods [65, 81], we use a 2D video diffusion model to perform conditional generation. Let $\mathcal{C} = [\mathbf{I}_1, \cdots, \mathbf{I}_K] \in \mathbb{R}^{K\times h\times w\times 3}$ denotes a target RGB video clip consisting of K frames, and $\tilde{\mathcal{C}} = [\tilde{\mathbf{I}}_1, \cdots, \tilde{\mathbf{I}}_K] \in \mathbb{R}^{K\times h\times w\times 3}$ denotes the corresponding video produced by our Voxel-to-3DGS VQ-VAE. We train a 2D video diffusion model for generating $\mathcal{C}$ conditioned on $\tilde{\mathcal{C}}$. Here, we also adapt a latent video diffusion model, using a frozen VAE from [6, 64] to map the original videos to the latent space. Let $\mathbf{z}^{\tilde{\mathcal{C}}}$ denotes the latent of rendered video $\tilde{\mathcal{C}}$, and $\mathbf{z}^{\mathcal{C}}$ denotes to latent of final refined video $\mathcal{C}$. We perform the standard forward process by adding noise to $\mathbf{z}^{\mathcal{C}}$, same as in Eq. (4). For the reverse process, we learn a conditional noise estimator $\boldsymbol{\epsilon}_\phi(\mathbf{z}_t^{\mathcal{C}}; t, \mathbf{z}^{\tilde{\mathcal{C}}})$.

Training: During training, we use images synthesized from VQ-VAE reconstruction and their ground-truth images as paired data and project both type of images into latent space via frozen VAE encoder $\mathcal{E}_\phi^{2D}$. Inspired by [24, 57], We adopt channel-concatenation as our condition mechanism. Specifically, we only apply random noises to $\mathbf{z}^{\mathbf{I}_{gt}}$ following equation Eq. 4 to obtain its noisy version $\mathbf{z}_t^{\mathcal{C}}$ and concatenate it with $\mathbf{z}^{\tilde{\mathcal{C}}}$ to formulate the final input to U-Net. After fine-tuning the original model, we can denoise noisy latent into a high-quality video clip under the guidance of coarse 3D prior $\tilde{\mathcal{C}}$.

Inference: To address the computational demands of training a 2D video diffusion model, we limit training to a small subset of frames. This yields abrupt transitions between clips when applied to longer video sequences with a simple replacement trick [42]. Inspired by [8, 57], we use Diffusion Forcing training and inference strategy. Specifically, after training with aforementioned setup, we finetune the network by independently sampling distinct noise levels for each individual frame within the clip instead of only one noise level for the whole clip. During sampling time, we condition previously generated W frames to the later $F - W$ frames via concatenating along the time dimension. The time embedding for the whole clip is $\mathbf{t} = [\underbrace{t_\epsilon, t_\epsilon, ..., t_\epsilon}_{W}, \underbrace{t, t, ..., t}_{F-W}]$, where time embedding t_ϵ is a small timestep and t is sampled from a fixed common diffusion scheduler. In this way, we can do sampling conditioned on

previously generated frames.

3.3. Training Losses

The whole training process can be divided into three stages. First, we train a generalizable VQ-VAE to reconstruct multiple scenes. For geometry reconstruction, we use a simple BCE loss to regularize the predicted occupancy $\mathbf{O}$ to be consistent with the occupancy of input voxel grid $\mathcal{V}$. For appearance reconstruction, we utilize L1 loss and SSIM loss following 3DGS [25], with a mask loss to separate foreground and background. We sample M frames to render for each single scene to apply the 2D image loss, yielding the full loss for one scene as:

$$\mathcal{L}_{recon} = \mathcal{L}_{\text{3D}} + \sum_{m=1}^{M} \mathcal{L}_{\text{2D}}^{m} \tag{6}$$

$$\mathcal{L}_{\text{3D}} = \lambda_{bce}\mathcal{L}_{bce} + \lambda_{vq}\mathcal{L}_{VQ} \tag{7}$$

$$\mathcal{L}_{\text{2D}} = \lambda_{rgb}\mathcal{L}_{1} + \lambda_{ssim}\mathcal{L}_{ssim} + \lambda_{fg}\mathcal{L}_{fg} \tag{8}$$

Secondly, we train the 3D diffusion model to generate coarse 3D scene latent. We compute Mean Squared Error between the clean sample $\mathbf{z}_0^{\mathcal{V}}$ and predicted clean sample $\hat{\mathbf{z}}_0^{\mathcal{V}}$ derived from $Eq.$ (5) to supervise our model:

$$\mathcal{L}_{diff}^{3D} = ||\mathbf{z}_0^{\mathcal{V}} - \hat{\mathbf{z}}_0^{\mathcal{V}}||^2 \tag{9}$$

Finally, for 2D diffusion model fine-tuning, we following the training procedure of pre-trained models [6, 64] as our training loss:

$$\mathcal{L}_{diff}^{2D} = ||\mathbf{z}_0^{\mathcal{C}} - \hat{\mathbf{z}}_0^{\mathcal{C}}||^2 \tag{10}$$

where $\hat{\mathbf{z}}_0^{\mathcal{C}}$ denotes the predicted original latent.

4. Experiments

4.1. Implementation Details

Datasets: We conduct our experiments on two outdoor autonomous driving datasets KITTI-360 [31] and Waymo [60]. For both datasets, we set $H = 32$, $W = 128$, $D = 192$ for the voxel grid $\mathcal{V}$, corresponding to $[-3\text{m}, 9.8\text{m}]$ for height, $[-25.6\text{m}, 25.6\text{m}]$ for width, and $[-20\text{m}, 56.8\text{m}]$ for the car forwarding direction. We filter out scenes with large turn, slow ego-motion, or significant dynamic objects. For our 2D loss $\mathcal{L}_{\text{2D}}$ to train VQ-VAE, we render RGB images at the resolution of 128×256 and resize the GT to the same resolution. For the 2D diffusion fine-tuning stage, we scale up image resolution to 640×960 for both datasets. Due to dataset scale limitation, we introduce spatial overlap in voxel grids, resulting in $\sim$20k samples for Waymo and $\sim$35k for KITTI-360. Further preprocessing details are provided in supplementary.

3D Diffusion: For our Voxel-to-3DGS VQ-VAE, we replace standard 3D convolutions with Asymmetrical Residual Blocks from Cylinder3D [88] for efficiency. The encoder downsamples the input by a factor of 4. During training, we use $M = 4$ 2D supervision images per scene for both datasets. For the reconstruction loss, we set $\lambda_{bce} = 1$, $\lambda_{vq} = 0.25$, $\lambda_{rgb} = 0.8$, $\lambda_{ssim} = 0.2$, $\lambda_{fg} = 0.5$. Our 3D diffusion model is a 3D variant adapted from [44], following the original transformer architecture of [48]. Note that we train on KITTI-360 and Waymo jointly and inject a dataset ID embedding for dataset awareness. VQ-VAE is trained from scratch for 300k steps using a batch size of 8 and latent 3D diffusion model is trained from scratch for 100k steps using a batch size of 32 separately with an initial learning rate of 1e-4. We use 8 A100 GPUS for training, our model requires approximately 2 days for VQ-VAE training and 3 days for 3D Diffusion training.

2D Diffusion: We trained two variants of the 2D diffusion model, each based on a different pretrained video diffusion backbone: Wan2.1-1.3B-i2v [64] and SVD [6]. This setup allows us to ensure a fair comparison with previous SVD-based methods [18], while also exploring the potential of our approach with the stronger WAN backbone. We initialize our models from the corresponding pretrained weights and reuse their VAEs, which remain frozen during training. For the SVD variant, following [57], we sequentially fine-tune the spatial and temporal layers. First, we fine-tune spatial layers on single-frame images for 12k steps (batch size 32), then temporal layers on video clips for 25k steps (batch size 8), with randomly sampled clip length up to 5. For WAN-variant, we fine-tune transformer modules for 25k steps, with randomly sampled clip length up to 17. To enhance temporal consistency, we apply the diffusion forcing strategy [8] on both models for an additional 15k steps. We use a learning rate of 3×10^{-5} with AdamW on a cluster of 8 NVIDIA Tesla A100 GPUs for training. The SVD and Wan2.1-1.3B fine-tuning required approximately 5 days.

4.2. Baselines and Metrics

Metrics: We evaluate the image quality of generated scenes by using FID [22], KID [5]. For FID and KID evaluation, we use 5k samples to measure the distribution similarity. We further evaluate FVD [62] on 1k clips to quantify the visual quality and temporal consistency of the generated videos. We use DUSt3R [66] to estimate camera poses of the generated scenes and calculate the rotation error (RotErr) and translation error (TransErr) following [21]. We additionally include Met3R [2] to measure the 3D consistency. We highlight the best, second-best, and third-best scores achieved on any metrics.

Baselines: We adopt the current state-of-the-art 3D urban generation methods as our baselines [3, 38, 74, 75, 80]. Recent image-to-video scene generation methods [18, 53] are also included as a reference despite their inherent differ-

Method	Setting	Conditions	Backbone	FID↓	KID↓	FVD↓	Met3R↓	TransErr↓	RotErr↓
DiscoScene [75]	3D Gen	3D Bbox	3D GAN	135.3	0.093	2025.9	0.5436	5.45	2.07
CC3D [3]	3D Gen	BEV Layout	3D GAN	90.8	0.091	706.1	0.2480	1.82	1.76
UrbanGen [80]	3D Gen	3D Bbox+Semantic Voxel	3D GAN	33.0	0.017	300.1	0.2196	0.21	0.25
Ours	3D Gen	3D Bbox+Road Map	SVD [6]	36.9	0.026	400.3	0.2141	0.23	0.13
Ours	3D Gen	3D Bbox+Road Map	WAN2.1-1.3B [64]	22.9	0.016	262.6	0.1247	0.06	0.23
Vista [18]	I2V	Ref Images	SVD [6]	25.6	0.016	234.0	0.0907	2.33	0.74
Gen3C [53]	I2V	Ref Images	Cosmos [45]	24.1	0.012	426.1	0.1476	0.04	0.11

Table 1. **Quantitative Comparison** about video quality and camera controllability on KITTI-360 .

Figure 3. **Qualitative Comparison on KITTI-360.** We show results under forward moving situations. Our method achieve superior visual quality over other baselines.

ences in target tasks. DiscoScene [75], CC3D [3], UrbanGen [80] and GaussianCity [74] are GAN-based methods with different layout conditions: DiscoScene [75] relies on foreground object layouts, CC3D [3] uses BEV semantic maps, UrbanGen [80] uses semantic voxel grids, and GaussianCity [74] uses BEV semantic and height maps. While DiscoScene, CC3D and UrbanGen use neural radiance fields as 3D representations, GaussianCity is a 3DGS-based method similar to us. Different from these GAN-based methods, Urban architect [38] is based on a pre-trained diffusion model that optimizes an urban scene using VSD loss [69] given provided 3D layout priors. As for image-to-video baselines, we adopt Vista [18] and Gen3C [53], where the former is fine-tuned from SVD [6] conditioned on various control signals, and the latter warps monocular depth to target views and inpaints it with a Cosmos [45] model.

4.3. Quantitative and Qualitative Comparison

Comparison with 3D Generation baselines: We compare visual quality and camera controllability with 3D generative baselines [3, 38, 74, 75, 80] in Tab. 1 and show qualitative comparisons in Fig. 3. DiscoScene achieves reasonable results for foreground objects, but background generation may occasionally fail. CC3D [3] and UrbanGen [80] improve the background based on BEV or semantic voxel conditions, but their 3D feature volume representations have resolution constraints. GaussianCity [74] provides a more compact scene representation by adopting BEV-Point repre-

Figure 4. **Ablation on Conditional Signal** for 2D augmentation. We show samples generated by different conditional signals on KITTI-360 after same training steps.

sentation but is trained from scratch on scene datasets, thus posing challenges in capturing high-frequency textures. Urban Architect [38] leverages diffusion model distillation with LG-VSD loss, which requires an extremely long processing time. In contrast, our method leverages the rich prior knowledge in the video diffusion model to synthesize more photorealistic appearance while maintaining good consistency, yielding the best FID, KID, FVD and Met3R. We also design various out-of-distribution (OOD) trajectories involving translation and rotation to evaluate camera controllability. Although all evaluated methods are trained primarily on forward-facing views, our approach leverages a 2D diffusion model to hallucinate unobserved regions, thereby maintaining robust control under large viewpoint shifts. Conversely, pure 3D generative baselines (e.g., DiscoScene, CC3D) lack the robust priors necessary to extrapolate beyond their training distribution. As a result, large camera movements induce severe image degradation in these methods, causing pose extractors like DUSt3R to fail and yielding significantly lower camera accuracy scores. Since the layout data and training code are unavailable, we only present qualitative results on GaussianCity and Urban Architect based on their paper or provided dataset samples.

Comparison with Image-to-video Generative baselines: We also compare with image-to-video methods [18, 53] and quantitative results are shown in Tab. 1. Since the generated frames from these methods often overlap with the GT frames, comparing FID/FVD is not entirely fair. However, our method exhibits a comparable performance. Notably, when the input and current viewpoints exhibit minimal overlap, Gen3C frequently suffers from significant quality degradation and inconsistency. This observation explains both its relatively high FVD score, which is sensitive to intra-clip performance drops, and its correspondingly higher Met3R score for evaluating 3D consistency. To test camera controllability, we utilized the identical camera trajectory as the 3D generation baselines. We observe that pure 2D methods [18] usually maintain a constant orientation on the straight road when conditioned on a small-angle curve signal while methods incorporating an explicit 3D representation achieve a more accurate camera viewpoint. We provide further details and qualitative results on this experiment in the supplementary.

	PSNR ↑	SSIM ↑	LPIPS ↓
w/o $\mathcal{L}_{bce}$, $G = 1$	21.47	0.750	0.327
w/ $\mathcal{L}_{bce}$, $G = 6$	21.53	**0.753**	**0.317**
w/ $\mathcal{L}_{bce}$, $G = 1$ (Ours)	**21.56**	**0.753**	0.321

Table 2. **Ablation on Voxel-to-3DGS VQ-VAE** on Waymo dataset. G denotes to number of Gaussians per voxel.

	KITTI-360		Waymo	
	FID↓	KID↓	FID↓	KID↓
Depth Cond.	78.6	0.070	77.4	0.071
RGB Cond. (Ours)	**36.9**	**0.026**	**41.3**	**0.030**

Table 3. **Ablation on Conditional Signals**. Both models are trained in the same way for the same number of steps.

4.4. Ablation Study

Ablation on Voxel-to-3DGS VQ-VAE: As shown in Tab. 2, we validate the effects of two components we use for training the Voxel-to-3DGS VQ-VAE. We compute PSNR, SSIM, LPIPS [85] on the Waymo [60] validation dataset. First, we ablate the effect of the BCE loss (w/o $\mathcal{L}_{bce}$), which provides direct supervision to help our model learn scene geometry. We find that while the VQ-VAE can still converge with only 2D supervision, it struggles to predict accurate scene occupancy, resulting in slightly degraded performance. Next, we explore the effect of increasing the number of Gaussians per voxel from 1 to 6 ($G = 6$) and find only modest improvements. This matches observations in [40, 52]. A possible explanation is that predicting diverse Gaussians from a shared voxel feature is difficult in our generalizable model. To balance GPU memory usage and performance, we set $G = 1$ in our approach.

Condition Signal for 2D Augmentation: For the ablation study of the 2D diffusion model, we utilize our SVD version for all experiments. We ablate our key design of generating coarse geometry and appearance in the 3D space. Following another line of work [14, 28, 39, 41, 87] that purely consider appearance generation in the 2D space, we replace rendered RGB image as a rendered geometry buffer (depth map in our case) as condition and train our 2D video diffusion in the same manner. We evaluate final FID/KID [22] in both cases and the results are shown in Tab. 3. Model conditions on the foreground RGB can synthesize images with higher quality at the same training steps since they provide direct texture information, hence improving training efficiency compared with using the indirect geometry conditions. We visualize different conditional signals and their synthesized full images in Fig. 4. Under the guidance

Figure 5. **Controllable Scene Synthesis on Waymo and KITTI360.** The visualization of conditional signals and corresponding synthesized images confirms the adherence of the generated content to diverse conditional guidance.

Figure 6. **Ablation on Inference Strategy**. We visualize neighboring frames obtained from two clips. Using the diffusion forcing strategy (w/ DF) significantly improves the consistency of background regions compared with the repaint strategy (w/o DF).

Figure 7. **Inpaint Samples**. We use Repaint strategy [42] to inpaint original scenes, demonstrating our model's capability to generate diverse scenes.

of the depth map, the surface of surrounding houses is much rougher and there are more noisy artifacts. Moreover, using depth as a conditioning signal struggles to preserve appearance consistency when revisiting the same place, whereas our RGB guidance enhances consistency by providing a coarse appearance guidance. More details about this experiment can be found in supplementary.

Inference Strategy: As we only model the background with our video diffusion model and fine-tune it on clips with a maximum frame 5/17, we encounter sudden changes between clips for distant areas. To solve this problem, we use previously generated frames as conditions for the later frames, inspired by [8, 57]. We show the impact of using the Diffusion Forcing strategy in Fig. 6. When using this strategy, frames between clips become consistent, otherwise the background will change drastically.

4.5. More Results

Scene Controllability: To demonstrate the fine-grained controllability of our method during the generation process, Fig. 5 showcases scenes synthesized under different conditional signals. Conditioned on the textual instructions provide control over global attributes such as weather and background composition, for example specifying whether the scene should contain only vegetation or include buildings. In addition, given bounding boxes and road maps, our method can also generate scenes with diverse vehicle layouts and distinct road structures, such as straight roads and curved turns.

Scene Inpainting: To showcase the generative capability of our model, we use the Repaint strategy [42] to regenerate certain parts of the original scene. Specifically, we fixed the first half of the scene latent, and inpaint another half part. It can be noticed in Fig. 7 that the shape of the house in the latter part has been changed.

5. Conclusion

We propose `ScenDi`, a novel urban scene generator that leverages 3D-to-2D diffusion cascades. By introducing a novel 3D latent diffusion model and a refinement stage with a 2D diffusion model, our method improves image quality while maintaining strong camera and content controllability through the pure 3D generation stage, aided by diverse conditional signals. Our method highlights a promising direction for complex scene generation by integrating both diffusion paradigms to fully unlock their potential.

Limitations and Future Work: One of our main challenges is that the quality of video diffusion depends on the preceding 3D LDM. Although the 2D refinement stage can mitigate some artifacts inherited from the 3D generation step, unsatisfactory 3D generation may cause corruption. Scaling up the training data and model size could further improve the visual quality of native 3D scene generation.

Acknowledgements

This work is supported by NSFC under grant 62441223 and Ant Group Research Fund.

References

[1] Titas Anciukevicius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J. Mitra, and Paul Guerrero. RenderDiffusion: Image diffusion for 3D reconstruction, inpainting and generation. *arXiv*, 2022. 2

[2] Mohammad Asim, Christopher Wewer, Thomas Wimmer, Bernt Schiele, and Jan Eric Lenssen. Met3r: Measuring multi-view consistency in generated images, 2024. 5

[3] Sherwin Bahmani, Jeong Joon Park, Despoina Paschalidou, Xingguang Yan, Gordon Wetzstein, Leonidas Guibas, and Andrea Tagliasacchi. Cc3d: Layout-conditioned generation of compositional 3d scenes, 2023. 1, 2, 5, 6

[4] Miguel Angel Bautista, Pengsheng Guo, Samira Abnar, Walter Talbott, Alexander Toshev, Zhuoyuan Chen, Laurent Dinh, Shuangfei Zhai, Hanlin Goh, Daniel Ulbricht, Afshin Dehghan, and Josh Susskind. Gaudi: A neural architect for immersive 3d scene generation. *arXiv*, 2022. 2

[5] Mikołaj Bińkowski, Danica J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans, 2021. 5

[6] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, Varun Jampani, and Robin Rombach. Stable video diffusion: Scaling latent video diffusion models to large datasets, 2023. 4, 5, 6

[7] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *arXiv*, 2021. 2

[8] Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2025. 4, 5, 8

[9] 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 *ICCV*, 2023. 2

[10] Jiacheng Chen, Ziyu Jiang, Mingfu Liang, Bingbing Zhuang, Jong-Chyi Su, Sparsh Garg, Ying Wu, and Manmohan Chandraker. Autoscape: Geometry-consistent long-horizon scene generation. *arXiv preprint arXiv:2510.20726*, 2025. 2

[11] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22246–22256, 2023. 2

[12] Kai Cheng, Xiaoxiao Long, Wei Yin, Jin Wang, Zhiqiang Wu, Yuexin Ma, Kaixuan Wang, Xiaozhi Chen, and Xuejin Chen. Uc-nerf: Neural radiance field for under-calibrated multi-view cameras in autonomous driving. *arXiv preprint arXiv:2311.16945*, 2023. 3

[13] 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. 2

[14] Boyang Deng, Richard Tucker, Zhengqi Li, Leonidas Guibas, Noah Snavely, and Gordon Wetzstein. Streetscapes: Large-scale consistent street view generation using autoregressive video diffusion. In *SIGGRAPH 2024 Conference Papers*, 2024. 1, 2, 7

[15] Prafulla Dhariwal and Alex Nichol. Diffusion models beat gans on image synthesis, 2021. 4

[16] Ruiyuan Gao, Kai Chen, Zhihao Li, Lanqing Hong, Zhenguo Li, and Qiang Xu. Magicdrive3d: Controllable 3d generation for any-view rendering in street scenes. *arXiv preprint arXiv:2405.14475*, 2024. 2

[17] Ruiyuan Gao, Kai Chen, Enze Xie, Lanqing Hong, Zhenguo Li, Dit-Yan Yeung, and Qiang Xu. MagicDrive: Street view generation with diverse 3d geometry control. In *International Conference on Learning Representations*, 2024. 2

[18] Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 1, 2, 5, 6, 7

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 2

[20] Mariam Hassan, Sebastian Stapf, Ahmad Rahimi, Pedro M. B. Rezende, Yasaman Haghighi, David Brüggemann, Isinsu Katircioglu, Lin Zhang, Xiaoran Chen, Suman Saha, Marco Cannici, Elie Aljalbout, Botao Ye, Xi Wang, Aram Davtyan, Mathieu Salzmann, Davide Scaramuzza, Marc Pollefeys, Paolo Favaro, and Alexandre Alahi. Gem: A generalizable ego-vision multimodal world model for fine-grained ego-motion, object dynamics, and scene composition control, 2024. 2

[21] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024. 5

[22] 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, 2018. 5, 7

[23] Mu Hu, Wei Yin, Chi Zhang, Zhipeng Cai, Xiaoxiao Long, Hao Chen, Kaixuan Wang, Gang Yu, Chunhua Shen, and Shaojie Shen. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation, 2024. 3

[24] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 4

[25] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time

radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 2, 3, 5

[26] Seung Wook Kim, Bradley Brown, Kangxue Yin, Karsten Kreis, Katja Schwarz, Daiqing Li, Robin Rombach, Antonio Torralba, and Sanja Fidler. Neuralfield-ldm: Scene generation with hierarchical latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 2

[27] Jumin Lee, Sebin Lee, Changho Jo, Woobin Im, Juhyeong Seon, and Sung-Eui Yoon. Semcity: Semantic scene generation with triplane diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024. 3

[28] Bohan Li, Jiazhe Guo, Hongsi Liu, Yingshuang Zou, Yikang Ding, Xiwu Chen, Hu Zhu, Feiyang Tan, Chi Zhang, Tiancai Wang, et al. Uniscene: Unified occupancy-centric driving scene generation. *arXiv preprint arXiv:2412.05435*, 2024. 1, 2, 7

[29] 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, 2023. 2

[30] Zuoyue Li, Zhenqiang Li, Zhaopeng Cui, Marc Pollefeys, and Martin R. Oswald. Sat2scene: 3d urban scene generation from satellite images with diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7141–7150, 2024. 2

[31] Yiyi Liao, Jun Xie, and Andreas Geiger. KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *arXiv preprint arXiv:2109.13410*, 2021. 4, 5

[32] 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. 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*, 2023. 1

[34] 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, 2023. 2

[35] Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023. 2

[36] Yuheng Liu, Xinke Li, Xueting Li, Lu Qi, Chongshou Li, and Ming-Hsuan Yang. Pyramid diffusion for fine 3d large scene generation, 2024. 3

[37] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023. 2

[38] Fan Lu, Kwan-Yee Lin, Yan Xu, Hongsheng Li, Guang Chen, and Changjun Jiang. Urban architect: Steerable 3d urban scene generation with layout prior. *arXiv preprint arXiv:2404.06780*, 2024. 5, 6, 7

[39] Jiachen Lu, Ze Huang, Zeyu Yang, Jiahui Zhang, and Li Zhang. Wovogen: World volume-aware diffusion for controllable multi-camera driving scene generation. In *European Conference on Computer Vision*, pages 329–345. Springer, 2024. 1, 2, 7

[40] 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. 7

[41] Yifan Lu, Xuanchi Ren, Jiawei Yang, Tianchang Shen, Zhangjie Wu, Jun Gao, Yue Wang, Siheng Chen, Mike Chen, Sanja Fidler, and Jiahui Huang. Infinicube: Unbounded and controllable dynamic 3d driving scene generation with world-guided video models, 2024. 1, 2, 7

[42] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11451–11461, 2022. 4, 8

[43] Quan Meng, Lei Li, Matthias Nießner, and Angela Dai. Lt3sd: Latent trees for 3d scene diffusion, 2024. 3

[44] Shentong Mo, Enze Xie, Ruihang Chu, Lewei Yao, Lanqing Hong, Matthias Nießner, and Zhenguo Li. Dit-3d: Exploring plain diffusion transformers for 3d shape generation. *arXiv preprint arXiv: 2307.01831*, 2023. 5

[45] NVIDIA, Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, Daniel Dworakowski, Jiaojiao Fan, Michele Fenzi, Francesco Ferroni, Sanja Fidler, Dieter Fox, Songwei Ge, Yunhao Ge, Jinwei Gu, Siddharth Gururani, Ethan He, Jiahui Huang, Jacob Huffman, Pooya Jannaty, Jingyi Jin, Seung Wook Kim, Gergely Klár, Grace Lam, Shiyi Lan, Laura Leal-Taixe, Anqi Li, Zhaoshuo Li, Chen-Hsuan Lin, Tsung-Yi Lin, Huan Ling, Ming-Yu Liu, Xian Liu, Alice Luo, Qianli Ma, Hanzi Mao, Kaichun Mo, Arsalan Mousavian, Seungjun Nah, Sriharsha Niverty, David Page, Despoina Paschalidou, Zeeshan Patel, Lindsey Pavao, Morteza Ramezanali, Fitsum Reda, Xiaowei Ren, Vasanth Rao Naik Sabavat, Ed Schmerling, Stella Shi, Bartosz Stefaniak, Shitao Tang, Lyne Tchapmi, Przemek Tredak, Wei-Cheng Tseng, Jibin Varghese, Hao Wang, Haoxiang Wang, Heng Wang, Ting-Chun Wang, Fangyin Wei, Xinyue Wei, Jay Zhangjie Wu, Jiashu Xu, Wei Yang, Lin Yen-Chen, Xiaohui Zeng, Yu Zeng, Jing Zhang, Qinsheng Zhang, Yuxuan Zhang, Qingqing Zhao, and Artur Zolkowski. Cosmos world foundation model platform for physical ai, 2025. 6

[46] NVIDIA, Alisson Azzolini, Hannah Brandon, Prithvijit Chattopadhyay, Huayu Chen, Jinju Chu, Yin Cui, Jenna Diamond, Yifan Ding, Francesco Ferroni, Rama Govindaraju, Jinwei Gu, Siddharth Gururani, Imad El Hanafi, Zekun Hao, Jacob Huffman, Jingyi Jin, Brendan Johnson, Rizwan Khan, George Kurian, Elena Lantz, Nayeon Lee, Zhaoshuo Li,

Xuan Li, Tsung-Yi Lin, Yen-Chen Lin, Ming-Yu Liu, Andrew Mathau, Yun Ni, Lindsey Pavao, Wei Ping, David W. Romero, Misha Smelyanskiy, Shuran Song, Lyne Tchapmi, Andrew Z. Wang, Boxin Wang, Haoxiang Wang, Fangyin Wei, Jiashu Xu, Yao Xu, Xiaodong Yang, Zhuolin Yang, Xiaohui Zeng, and Zhe Zhang. Cosmos-reason1: From physical common sense to embodied reasoning, 2025. 4

[47] Julian Ost, Andrea Ramazzina, Amogh Joshi, Maximilian Bömer, Mario Bijelic, and Felix Heide. Lsd-3d: Large-scale 3d driving scene generation with geometry grounding. *The Fortieth AAAI Conference on Artificial Intelligence*, 2026. 2, 4

[48] William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022. 4, 5

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

[50] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 4

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

[52] Xuanchi Ren, Yifan Lu, Hanxue Liang, Jay Zhangjie Wu, Huan Ling, Mike Chen, Francis Fidler, Sanja annd Williams, and Jiahui Huang. Scube: Instant large-scale scene reconstruction using voxsplats. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. 7

[53] Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 1, 2, 5, 6, 7

[54] Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulò, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3dg: Latent 3d gaussian diffusion. In *SIGGRAPH Asia 2024 Conference Papers*, 2024. 2

[55] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 3

[56] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models, 2022. 4

[57] Jiahao Shao, Yuanbo Yang, Hongyu Zhou, Youmin Zhang, Yujun Shen, Vitor Guizilini, Yue Wang, Matteo Poggi, and Yiyi Liao. Learning temporally consistent video depth from video diffusion priors, 2024. 4, 5, 8

[58] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model, 2023. 2

[59] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models, 2022. 4

[60] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5, 7

[61] 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. 2

[62] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges, 2019. 5

[63] Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitrii Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. SV3D: Novel multi-view synthesis and 3D generation from a single image using latent video diffusion. In *European Conference on Computer Vision (ECCV)*, 2024. 2

[64] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 4, 5, 6

[65] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin C.K. Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution, 2024. 4

[66] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 5

[67] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-driven world models for autonomous driving. *arXiv preprint arXiv:2309.09777*, 2023. 2, 4

[68] Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. *arXiv preprint arXiv:2311.17918*, 2023. 2

[69] 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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. 2, 6

[70] Yuqing Wen, Yucheng Zhao, Yingfei Liu, Fan Jia, Yanhui Wang, Chong Luo, Chi Zhang, Tiancai Wang, Xiaoyan Sun, and Xiangyu Zhang. Panacea: Panoramic and controllable video generation for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6902–6912, 2024. 2

[71] Tong Wu, Yu-Jie Yuan, Ling-Xiao Zhang, Jie Yang, Yan-Pei Cao, Ling-Qi Yan, and Lin Gao. Recent advances in 3d gaussian splatting. *Computational Visual Media*, 10(4):613–642, 2024. 2

[72] Zhennan Wu, Yang Li, Han Yan, Taizhang Shang, Weixuan Sun, Senbo Wang, Ruikai Cui, Weizhe Liu, Hiroyuki Sato, Hongdong Li, and Pan Ji. Blockfusion: Expandable 3d scene generation using latent tri-plane extrapolation. *ACM Transactions on Graphics*, 43(4), 2024. 2, 3

[73] Haozhe Xie, Zhaoxi Chen, Fangzhou Hong, and Ziwei Liu. CityDreamer: Compositional generative model of unbounded 3D cities. In *CVPR*, 2024. 2

[74] Haozhe Xie, Zhaoxi Chen, Fangzhou Hong, and Ziwei Liu. Generative gaussian splatting for unbounded 3D city generation. In *CVPR*, 2025. 5, 6

[75] Yinghao Xu, Menglei Chai, Zifan Shi, Sida Peng, Skorokhodov Ivan, Siarohin Aliaksandr, Ceyuan Yang, Yujun Shen, Hsin-Ying Lee, Bolei Zhou, and Tulyakov Sergy. Discoscene: Spatially disentangled generative radiance field for controllable 3d-aware scene synthesis. *arxiv: 2212.11984*, 2022. 2, 5, 6

[76] Yinghao Xu, Hao Tan, Fujun Luan, Sai Bi, Peng Wang, Jiahao Li, Zifan Shi, Kalyan Sunkavalli, Gordon Wetzstein, Zexiang Xu, and Kai Zhang. Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model, 2023. 2

[77] Xiuyu Yang, Yunze Man, Jun-Kun Chen, and Yu-Xiong Wang. Scenecraft: Layout-guided 3d scene generation. In *Advances in Neural Information Processing Systems*, 2024. 2

[78] Xuemeng Yang, Licheng Wen, Yukai Ma, Jianbiao Mei, Xin Li, Tiantian Wei, Wenjie Lei, Daocheng Fu, Pinlong Cai, Min Dou, Botian Shi, Liang He, Yong Liu, and Yu Qiao. Drivearena: A closed-loop generative simulation platform for autonomous driving. *arXiv preprint arXiv:2408.00415*, 2024. 2

[79] Yuanbo Yang, Yifei Yang, Hanlei Guo, Rong Xiong, Yue Wang, and Yiyi Liao. Urbangiraffe: Representing urban scenes as compositional generative neural feature fields. *ARXIV*, 2023. 1, 2

[80] Yuanbo Yang, Yujun Shen, Yue Wang, Andreas Geiger, and Yiyi Liao. Urbangen: Urban generation with compositional and controllable neural fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–17, 2025. 5, 6

[81] Fanghua Yu, Jinjin Gu, Zheyuan Li, Jinfan Hu, Xiangtao Kong, Xintao Wang, Jingwen He, Yu Qiao, and Chao Dong. Scaling up to excellence: Practicing model scaling for photo-realistic image restoration in the wild, 2024. 4

[82] Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T. Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. *arXiv:2406.09394*, 2024. 2

[83] Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T. Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, and Charles Herrmann. Wonderjourney: Going from anywhere to everywhere. In *CVPR*, 2024. 2

[84] 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

[85] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 7

[86] Junsheng Zhou, Weiqi Zhang, and Yu-Shen Liu. Diffgs: Functional gaussian splatting diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. 2

[87] Benjin Zhu, Xiaogang Wang, and Hongsheng Li. Consistentcity: Semantic flow-guided occupancy dit for temporally consistent driving scene synthesis. In *IEEE/CVF International Conference on Computer Vision*, 2025. 1, 2, 7

[88] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. *arXiv preprint arXiv:2011.10033*, 2020. 5