Drive-Cascade: Autoregressive Occupancy to LiDAR and Video Synthesis

Shuangming Lei[1,*], Yuehao Huang[1,*], Yi Yao[1], Yijia Xie[1], Jingke Wang[1], Ruoyu Wang[1], Jiajun Lv[1], Guanglin Xu[2], AiXue Ye[2], Bingbing Liu[2], Siyuan Cheng[2], Hongbo Zhang[2], Yukai Ma[1,†], Yong Liu[1,†]

[1]Zhejiang University [2]Huawei Noah's Ark Lab

{shuangminglei, yuehaohuang, yukaima}@zju.edu.cn, yongliu@iipc.zju.edu.cn

Abstract

The generation of realistic, consistent, and controllable multi-modal data for dynamic driving scenes remains a crucial challenge in autonomous vehicle simulation. Current methods often struggle to maintain geometric and temporal coherence, particularly when synthesizing complex interactions across disparate modalities, such as LiDAR and video. In this paper, we propose a novel cascaded autoregressive framework to generate highly realistic and multi-modally aligned driving scenes. The key innovation of this work is the utilization of dynamic occupancy as a unified and explicit intermediate representation. The proposed framework operates in two stages: first, the system generates a coherent sequence of controllable dynamic occupancy grids that capture the spatiotemporal geometry of the scene. Second, conditioned on the generated occupancy prior, two specialized diffusion models autoregressively synthesize the corresponding LiDAR point clouds and camera videos. By anchoring the generation of all modalities to a shared geometric foundation, the proposed model inherently ensures cross-modal consistency and temporal stability. Extensive experiments demonstrate that the proposed approach significantly outperforms state-of-the-art methods in terms of generation fidelity, geometric accuracy, and long-term temporal coherence for both LiDAR and video synthesis, paving the way for high-fidelity and multi-modal simulation.

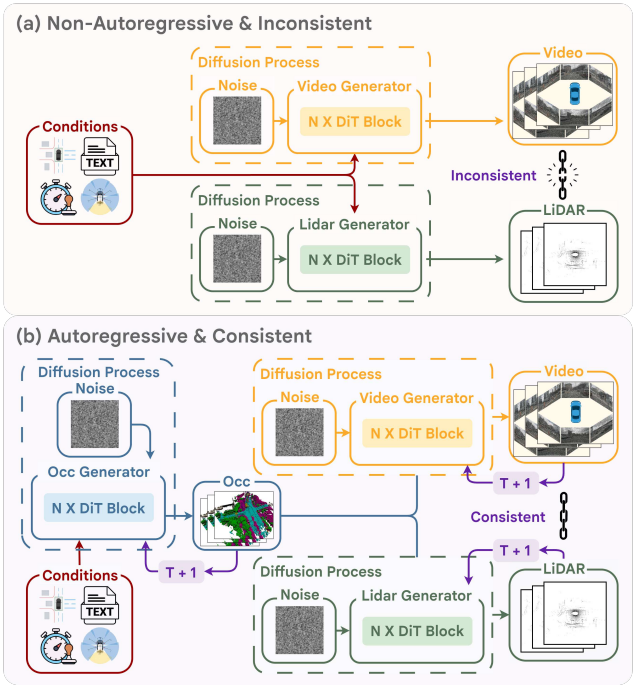

Figure 1. **Comparison of Generation Frameworks.** (a) Prior non-autoregressive methods generate short and disjoint segments, resulting in temporal incoherence and inter-modal inconsistencies. (b) The proposed autoregressive framework, centered on a unified 4D occupancy sequence, ensures long-horizon temporal coherence and strict cross-modal consistency by utilizing the 4D state as a shared spatiotemporal scaffold.

1. Introduction

Data-driven simulation is indispensable for the entire development lifecycle of autonomous driving systems [6]. High-fidelity, diverse, and controllable virtual environments allow for rigorous evaluation prior to real-world deployment. However, the acquisition of real-world data is prohibitively costly and time-consuming, and it fails to capture the full spectrum of rare and safety-critical "long-tail" scenarios [4]. Consequently, generative world models have emerged as a compelling approach for the construction of scalable world simulators capable of synthesizing diverse, novel, and adversarial driving scenarios [12, 15, 21].

The field is rapidly advancing toward unified frameworks that jointly generate multi-modal sensor data [19] to establish a holistic perceptual environment. Within this paradigm, 3D semantic occupancy has emerged as an effective intermediate representation, providing fine-grained geometric and semantic details. Despite this progress,

*Equal contribution.
†Corresponding authors.

Table 1. Feature comparison of generative world models for autonomous driving. The table highlights the key limitations of prior studies regarding temporal coherence and multi-modal consistency, which the proposed method successfully addresses.

Method	Control	Temporal	Cross-View	Multi-modal	Length
Vista [10]	×	×	×	×	1-10s
LidarDM [42]	✓	✓	×	×	10s+
MagicDrive [7]	✓	×	✓	×	5s (12FPS)
DriveArena [36]	✓	×	✓	×	10s+
X-Drive [33]	✓	×	✓	✓	5s (12FPS)
Ours	✓	✓	✓	✓	**10s+ (12FPS)**

occupancy-centric generative models continue to face two critical and interrelated challenges that severely limit the utility of these models for long-term simulation: temporal coherence in long-horizon generation and inter-modality consistency.

Many existing generative world models [7, 8, 20] operate non-autoregressively, producing scenes in short and disjoint segments. When extended to long-horizon generation, this approach introduces severe visual artifacts, such as flickering, inconsistent object appearances, and the spontaneous disappearance and reappearance of dynamic agents. These failures violate the fundamental physical principles of motion continuity and object permanence, rendering the models unreliable for the simulation of complex and safety-critical driving maneuvers that require a stable and extended temporal context. Furthermore, the assurance of precise geometric alignment across modalities remains highly challenging. Even subtle deviations in the underlying 3D structure can be magnified in sensor-specific outputs (*e.g.*, LiDAR point clouds and camera images), leading to severe inconsistencies or cross-modal hallucinations, as depicted in Figure 1(a).

To address these limitations, we propose Drive-Cascade, an autoregressive world model illustrated in Figure 1(b). This model establishes 4D dynamic semantic occupancy sequences [2] as the core dynamic world state. The architecture is designed to enable temporally consistent long-horizon generation while preserving fine-grained controllability. By autoregressively modeling the dynamic 4D state, Drive-Cascade enforces temporal consistency and object permanence. The main contributions of this work are summarized as follows:

- We introduce Drive-Cascade, a unified and autoregressive framework based on 4D dynamic semantic occupancy sequences. This design achieves long-horizon temporal coherence and fine-grained control for multi-modal scene generation.
- We design a multi-modal generation mechanism conditioned on the dynamic occupancy sequence as a spatiotemporal scaffold. This mechanism ensures geometric consistency in the autoregressively generated video and LiDAR data, thereby mitigating cross-modal hallucinations.
- We conduct extensive experiments on standard benchmarks. The results demonstrate that Drive-Cascade significantly outperforms baseline methods in terms of generation fidelity, controllability, and temporal stability.

2. Related Work

Generative Models in Autonomous Driving. Generative models serve as a cornerstone for the simulation of complex driving dynamics and the facilitation of robust decision-making. The existing literature predominantly focuses on three representations: video, occupancy grids, and LiDAR. In video synthesis, GAIA-1 [15] and DriveArena [36] employ autoregressive modeling, while Epona [37] and MagicDrive [7] introduce DiT architectures and BEV constraints to improve temporal consistency. For occupancy generation, OccWorld [39] and DynamicCity [2] provide detailed 3D spatial priors. Meanwhile, LiDAR generation has transitioned from early GAN-based models [41] to recent diffusion-based approaches, such as RangeLDM [16] and PVD [14]. To unify these modalities, models such as GENESIS [11] attempt latent-space alignment, whereas UniScene [19] adopts explicit occupancy. Our Drive-Cascade distinguishes itself by utilizing a unified 4D occupancy representation. By incorporating the temporal dimension directly into the occupancy backbone, we provide a more robust geometric anchor that ensures high-fidelity and spatiotemporally consistent generation across both LiDAR and video.

Controllability in Scene Synthesis. A fundamental challenge in scene synthesis is the balance between control precision and input cost. High-precision methods [25, 40] rely on expensive annotations, such as 3D bounding boxes or HD maps, which restrict the scalability of these methods. Conversely, low-cost approaches, such as Drive-GPT4 [35], leverage text-based interaction but often fail to guarantee fine-grained spatial fidelity. To bridge this gap, recent studies, such as SceneDiffuser++ [28], utilize intermediate scene graphs. Our approach advances this line of research by adopting spatiotemporal occupancy as the control signal. This representation acts as a strong structural prior, offering a more intuitive and editable interface for the simulation of the precise evolution of dynamic traffic agents compared with abstract latent codes or sparse text.

Long-term Generation Models. The construction of robust world models necessitates the ability to simulate consistent and long-horizon environmental evolutions. Current paradigms are split between diffusion-based models (*e.g.*, SVD [3]), which are often constrained by fixed temporal windows, and autoregressive models (*e.g.*, DrivingWorld [17]), which naturally support variable-length synthesis but face the risk of compounding errors and distribution drift.

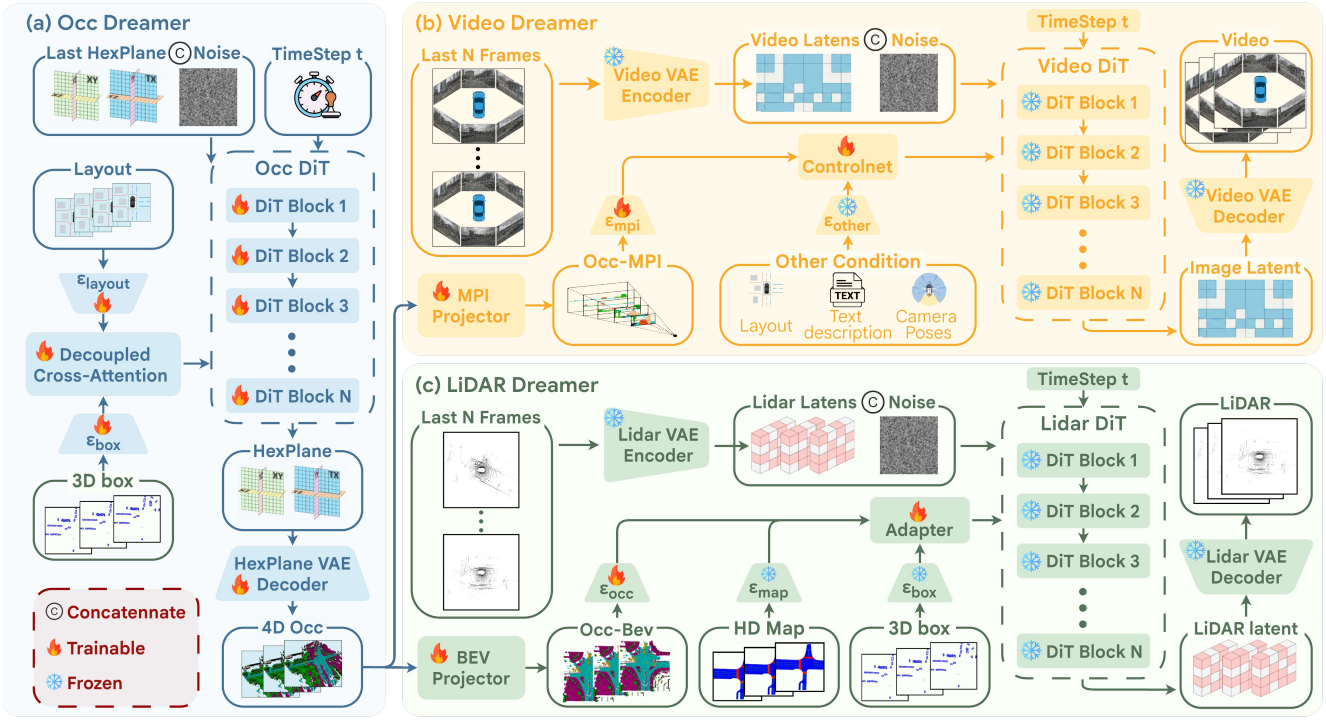

Figure 2. Overview of the **Drive-Cascade** architecture. The proposed autoregressive and occupancy-centric pipeline consists of three core modules: **(a) Occ Dreamer**, which acts as the state transition model to synthesize the 4D occupancy Occ_t by conditioning an Occ DiT on multi-granularity inputs; **(b) Video Dreamer**, which serves as the video emission model to transform Occ_t into a semantic MPI, providing fine-grained geometric guidance via a ControlNet to generate temporally coherent multi-view videos; and **(c) LiDAR Dreamer**, which functions as the LiDAR emission model to project Occ_t and layout priors into BEV features. These features are subsequently fused by an Adapter to condition a LiDAR DiT for the synthesis of LiDAR point clouds.

Recent hybrid architectures, such as ART-V [32], attempt to combine the fidelity of diffusion models with autoregressive continuity. Diverging from these single-modality efforts, Drive-Cascade is built upon a multi-modal autoregressive framework. Our key insight lies in leveraging the cascading relationship from occupancy to multi-sensor outputs, which effectively mitigates drift by anchoring the generation in a stable and geometrically consistent occupancy sequence.

3. Methodology

Overview. The overall architecture of Drive-Cascade, as illustrated in Fig. 2, constitutes a unified autoregressive world model designed to enforce temporal coherence and cross-modal consistency. The architecture features a novel system-level algorithmic cascade built around a central dynamic world state: a shared 4D occupancy scaffold. Compared to directly coupled generation, this design structurally constrains geometry and significantly reduces drift. The recursive generation process begins with the Occ Dreamer (Section 3.1), which acts as the core state transition model. This module generates the current 4D semantic occupancy grid Occ_t conditioned on external control signals C_t (Layout and 3D boxes) and the 4D occupancy from the previous timestep $t-1$. The generated Occ_t serves as a unified geometric scaffold that is broadcast to two parallel emission models to ensure strict cross-modal alignment. The Video Dreamer (Section 3.2) renders Vid_t conditioned on Occ_t and the previous N video frames from $t-1$, while the LiDAR Dreamer (Section 3.3) produces $LiDAR_t$ conditioned on the identical Occ_t and the previous N LiDAR frames from $t-1$. Finally, the complete multi-modal world state $S_t = \{Occ_t, Vid_t, LiDAR_t\}$ is cached and fed back as historical input for the generation at timestep $t+1$, thereby achieving stable long-horizon rollouts without abrupt collapse.

3.1. Occ Dreamer: Dynamic Occupancy Generation

As shown in Fig. 2(a), the Occ Dreamer aims to generate high-fidelity and highly dynamic 4D scenes. The core architecture of this module combines a HexPlane VAE and a trainable Diffusion Transformer (Occ DiT). The primary task of the VAE is to process high-dimensional and sparse 4D occupancy sequences, represented as $Z \times X \times Y \times T \times C_{sem}$, where Z denotes the vertical bins, X and Y form the BEV grid, T represents the time steps, and C_{sem} denotes the semantic classes. The VAE encoder significantly

reduces the dimensionality of the data, transforming it into a compact and informative HexPlane latent representation z_{occ}, which can subsequently be reconstructed accurately via the HexPlane VAE Decoder.

The generative module operates within this low-dimensional and efficient latent space of z_{occ}. The module functions as a denoising network, tasked with learning to reconstruct the original latent representation z_0 from pure Gaussian noise z_T through a defined step-by-step denoising process. To achieve stable and efficient training, we employ the Flow Matching [24] strategy to guide and optimize the process. Flow Matching is chosen over traditional discrete diffusion models (DDIM [27]) because it optimizes for continuous probability flows, which is better suited for learning the smooth and continuous state transition dynamics inherent in the $z_{occ(t-1)} \rightarrow z_{occ(t)}$ autoregressive loop.

Multi-granularity Conditioning. A key innovation of the proposed approach lies in the ability to handle multi-granularity conditions. To generate scenes that comply with real-world rules and contain specific dynamic events, the model must interpret and fuse two distinct types of control signals, $C_t = \{Layout_t, Boxes_t\}$. On one hand, the 2D semantic map $Layout_t$ ($H \times W \times C_{layout}$), which represents the static environment (*e.g.*, roads, buildings, and vegetation), is encoded by a 2D CNN into a spatial feature map f_{layout} that is rich in contextual information. On line with this, the list of N agent bounding boxes $Boxes_t$, which represents dynamic elements (defined by position, dimensions, orientation, and class: $[x, y, z, l, w, h, \theta, \text{class}]$), is processed by a Transformer encoder and converted into a set of $N \times D$ tokens f_{boxes} to capture the precise state of each agent. The dual-path encoder design matches the characteristics of the two condition types. Inside each Transformer module of the DiT, the flattened f_{layout} and f_{boxes} are injected as conditions via decoupled cross-attention. The DiT backbone utilizes a cross-attention mechanism to query and fuse the heterogeneous information. This mechanism enables the model to selectively focus on relevant static environmental constraints (from f_{layout}) and immediate dynamic agent constraints (from f_{boxes}) at every generation step. The final result is the generation of a HexPlane z_0 that is geometrically precise, semantically reasonable, and strictly adherent to all input conditions.

Autoregressive Dynamics. To generate a complete dynamic scene, achieving precision in a single frame is insufficient; the sequence must also guarantee coherence and logical consistency over time. This challenge is amplified by the risk of cascading error accumulation inherent in long-horizon autoregression. To address this issue, we design an autoregressive dynamics mechanism. Specifically, when generating the t-th frame, the DiT is conditioned not only on the current control signals C_t but also on the latent representation from the previous timestep, $z_{occ(t-1)}$, as a historical reference. This conditioning is implemented by concatenating $z_{occ(t-1)}$ with the input noise z_T for the current timestep along the channel dimension. This design provides the model with an explicit reference to the state of the previous timestep, compelling the network to generate frames based on the learned state transition rules from $t-1$ to t rather than in isolation.

3.2. Video Dreamer: Occupancy-based Autoregressive Video Generation

We utilize a video diffusion model, DreamForge [26], as the backbone. To achieve controllable generation, we adopt a local control strategy that aligns pixel-level spatial features from the condition to the generated image. Following methods such as ControlNet[38] and SyntheOcc [20], this approach is more effective than the use of global 1D embeddings, as the strategy preserves the fine-grained spatiotemporal geometry of the 3D occupancy input.

Occupancy-to-MPI Conditioning Bridge. To bridge the 3D occupancy grid (Occ_t) with the 2D diffusion backbone, we employ the 3D Semantic Multiplane Image (MPI) representation [20], which allows for the efficient storage of semantic and geometric information, including occluded elements. Upon receiving Occ_t from Section 3.1, we transform the grid into Occ-MPI (semantic MPIs for the N cameras) via the MPI Projector. For each camera n, we define D discrete depth planes at depths d_l. For each pixel (u, v) on the l-th plane, the point is unprojected to the 3D world coordinates P_l using the camera intrinsics K_n and extrinsics T_n:

$$P_l = T_n \cdot (K_n^{-1} \cdot [u \cdot d_l, v \cdot d_l, d_l]^T) \tag{1}$$

We then query the corresponding semantic label from the Occ_t voxel grid at P_l via trilinear interpolation: $MPI_{n,l}(u, v) = \text{Interpolate}(Occ_t, P_l)$. This process yields a $D \times H \times W$ semantic MPI for each camera. To align the MPIs with the diffusion latent space, we adopt the 1×1 convolutional encoder design from SyntheOcc [20]. This lightweight encoder extracts spatially aligned features without downsampling. These features are subsequently added directly to the ControlNet, providing the model with precise and view-specific geometric guidance that is strictly faithful to the input Occ_t.

Autoregressive Dynamics. A key feature of the Video Dreamer is the explicit modeling of temporal dynamics. In addition to the static geometric prior from Occ_t, we explicitly condition the module on the sequence of the previous N frames generated at timestep $t-1$ (denoted as $Vid_{t-1}^{f-N \cdots f}$, where f is the frame index). This temporal context is processed by the native motion-aware attention mechanisms of the backbone. The dual-conditioning strategy, which combines the static geometric prior (Occ_t) with the dynamic temporal prior, ensures that the generated frame Vid_t is

geometrically accurate and follows a coherent motion trajectory.

3.3. LiDAR Dreamer: Occupancy-based Autoregressive LiDAR Generation

In this section, we introduce the LiDAR Dreamer, a framework for the autoregressive generation of high-fidelity LiDAR point cloud sequences. To achieve efficient generation, the sparse 3D point cloud is initially compressed into a dense 2D latent representation via a VAE. Specifically, the input point cloud x is fed into a voxelizer V and converted into a BEV format [34] feature map. The LiDAR VAE Encoder compresses the BEV feature map into a discrete 2D latent token map z, while the decoder is responsible for reconstructing the original point cloud from z. The subsequent diffusion model operates within the low-dimensional and dense latent space of z. The core generative model is a Diffusion Transformer (DiT).

Occupancy-to-BEV Conditioning. A key feature of the LiDAR Dreamer is the ability to fuse information from multiple sources to guide the generation process, particularly through the introduction of explicit geometric priors. For multimodal condition fusion, we introduce Layout Conditioning: static and dynamic layout information (such as 3D bounding boxes and HD maps) is projected into the BEV space. Each instance (*e.g.*, vehicles, pedestrians, and lane lines) is mapped into the color space [5] to form conditional images. These conditional images are concatenated and processed through a lightweight image adapter, generating multi-scale features that are subsequently added to the corresponding Transformer layers of the DiT via an Adapter. This process provides the model with semantic guidance regarding the scene layout. Furthermore, the 4D occupancy Occ_t generated in Section 3.1 serves as a strong geometric prior. Through the BEV Projector, Occ_t is converted into Occ-BEV. This projector consists of a 3D sparse convolutional network that extracts 3D features and flattens them along the Z-axis to form an informative and multi-channel BEV feature map $f_{Occ_{bev}}$. The feature map $f_{Occ_{bev}}$ is injected as a geometric condition into each layer of the DiT denoising network via the Adapter, which compels the diffusion model to generate points exclusively in regions explicitly marked as occupied by Occ_t, effectively suppressing generation in free regions. This mechanism acts as a strict geometric mask and density prior.

Autoregressive Dynamics. To generate temporally coherent LiDAR sequences, the LiDAR Dreamer employs an autoregressive dynamics mechanism. When generating the t-th frame, the module is conditioned on the previous k LiDAR frames generated at timestep $t-1$ (denoted as $LiDAR_{t-1}^{f-k\cdots f}$, where f is the frame index) as an additional temporal prior. This prior is fed into the DiT alongside the layout and occupancy conditions, ensuring that the generated $LiDAR_t$ is spatially plausible and consistent with historical observations regarding temporal dynamics.

4. Experiment

To comprehensively evaluate the performance of our model, all experiments are conducted on the highly challenging nuScenes autonomous driving benchmark [4]. Regarding data processing, the semantic occupancy annotations provided by the nuScenes-Occupancy dataset [29] consist of only 2Hz keyframes. Because this frequency is insufficient to support high-frequency dynamic generation, we employ the NKSR [18] interpolation technique to upsample the annotations to a 12Hz frame rate [19]. In our experimental design, we focus the evaluation on multi-modal generation capabilities, conducting detailed experimental analyses on the generation quality and consistency of three key modalities: occupancy, video, and LiDAR.

Multimodal Autoregressive Generation. As shown in Figure 3, our Drive-Cascade framework demonstrates robust multimodal autoregressive generation capabilities, offering both flexibility and controllability. Specifically, when processing large-scale scenes, the Occupancy Dreamer initially distills and generates dynamic occupancy from HD maps and 3D boxes. This occupancy information encodes the geometric structure of the scene and captures the dynamic changes of objects, thereby serving as a highly robust spatiotemporal scaffold. Both the LiDAR Dreamer and the Video Dreamer utilize this scaffold as core guidance. The former incorporates additional conditions to autoregressively generate long-sequence LiDAR. The latter generates temporally consistent and content-rich long-form surround-view videos based on the scaffold and provided captions.

Occupancy Forecasting and Generation. We quantitatively evaluate the occupancy forecasting and generation capabilities on the nuScenes-Occupancy validation set [29], comparing our model with state-of-the-art methods, such as OccWorld [39], OccSora [30], and UniScene [19]. As shown in Table 2, our method achieves the highest performance across all evaluation metrics. On the mIoU metric, which measures semantic accuracy, our model reaches 24.17, representing a significant improvement of 24.33% compared with the best baseline, UniScene. For completeness, we also evaluate against GaussianWorld under the same forecasting metrics. Although GaussianWorld requires input videos while our method relies solely on high-level conditions, our approach still achieves a superior mIoU (24.17 versus 22.13). Furthermore, our model demonstrates a significant advantage on metrics that evaluate the fidelity and temporal consistency of the generated sequences, reducing F3D by 22.12% and MMD by 22.5% compared with UniScene. These results demonstrate that our model generates high-precision semantic occupancy

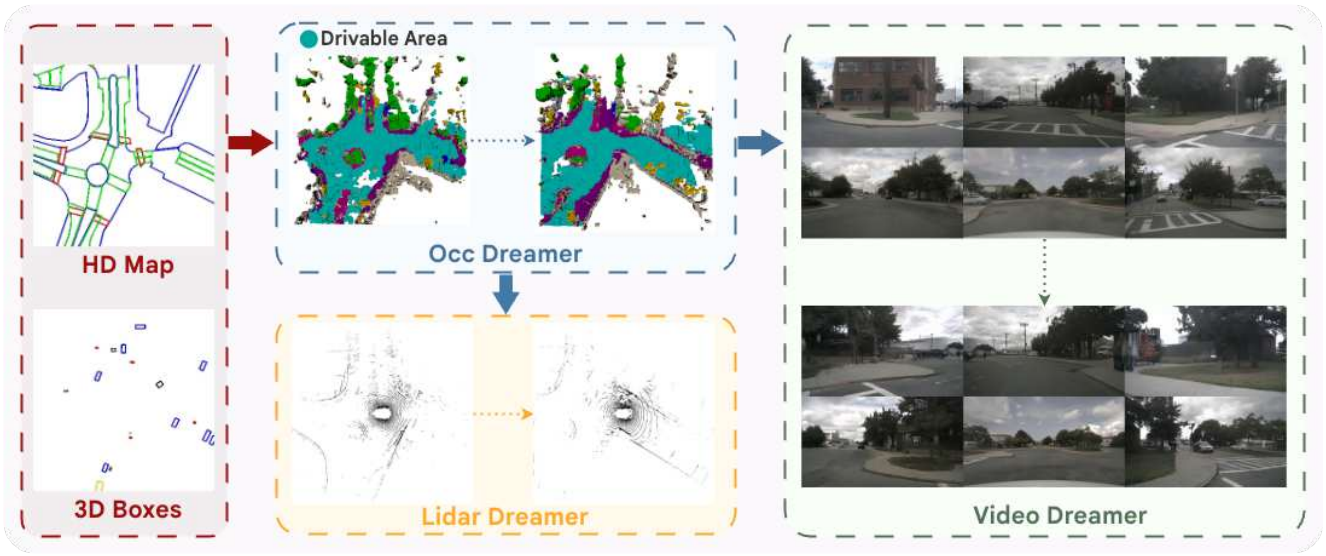

Figure 3. Our **Drive-Cascade** model demonstrates robust generative performance. First, the Occupancy Dreamer synthesizes dynamic occupancy sequences from HD maps and 3D bounding boxes. Subsequently, the LiDAR Dreamer autoregressively generates long-sequence LiDAR point clouds, guided by the synthesized occupancy prior and additional conditions. Finally, the Video Dreamer renders controllable long-form videos, conditioned on the identical occupancy scaffold and auxiliary inputs.

Table 2. Quantitative comparison of occupancy forecasting and generation on the nuScenes-Occupancy validation set.

Method	mIoU ↑	F3D ↓	MMD ↓
OccWorld [39]	17.52	164.23	12.56
OccSora [30]	15.11	207.70	11.23
UniScene [19]	19.44	158.55	10.60
Ours	**24.17**	**123.47**	**8.21**

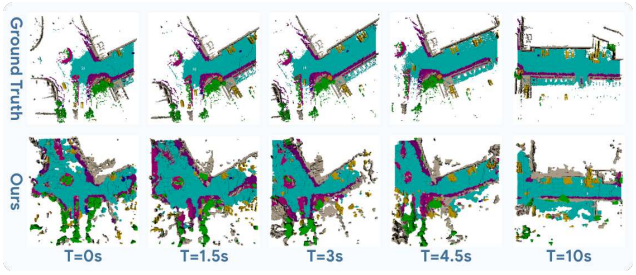

Figure 4. **Qualitative evaluation of occupancy generation.** Our method generates high-fidelity and spatiotemporally coherent occupancy sequences. Even during long-horizon generation (over 10 seconds), the model maintains strong temporal consistency and synthesizes plausible details, effectively reconstructing missing regions present in the ground truth.

and maintains a high degree of realism and temporal coherence in long-term predictions.

As shown in Figure 4, the model generates high-fidelity and spatiotemporally coherent occupancy sequences. Even in long-horizon generation (over 10 seconds), our method maintains strong temporal consistency and generates plausible, fine-grained scene details, achieving a richness that exceeds sparse regions in the ground truth.

Video Generation Results. We apply our proposed method to generate multi-view driving scenes using annotations from the nuScenes validation set. The objective is to evaluate the fidelity of the generated data and the efficacy of the data for downstream perception tasks.

We compare our model against several state-of-the-art baseline methods, including MagicDrive [7], MagicDrive3D [8], and DreamForge [26]. This selection is based on a critical criterion for fair comparison: similar to our method, these baselines can generate video sequences without conditioning on a ground-truth initial frame. For a comprehensive and fair comparison, we evaluate against models based on SD V1.5 and more advanced models based on 3DVAE and DiT. Following previous studies [7, 26], we utilize BevFormer [22] for 3D object detection (mAP) and BEV segmentation (mIoU). Generation fidelity is quantified using the Fréchet Video Distance (FVD) [13].

The results are presented in Table 3. In terms of generation fidelity, our model achieves an FVD score of 81.87, outperforming the strongest baseline, MagicDriveDiT, by 12.97 points. For downstream tasks, the scenes generated by our model achieve 36.66 mIoU in BEV segmentation. Notably, this performance is the highest among all generative models and is nearly on par with the original real nuScenes dataset. In 3D object detection, the achieved mAP of 18.35 is competitive with MagicDriveDiT (18.17). Although a DreamForge variant achieves a higher mAP, the variant suffers from significantly worse fidelity and segmen-

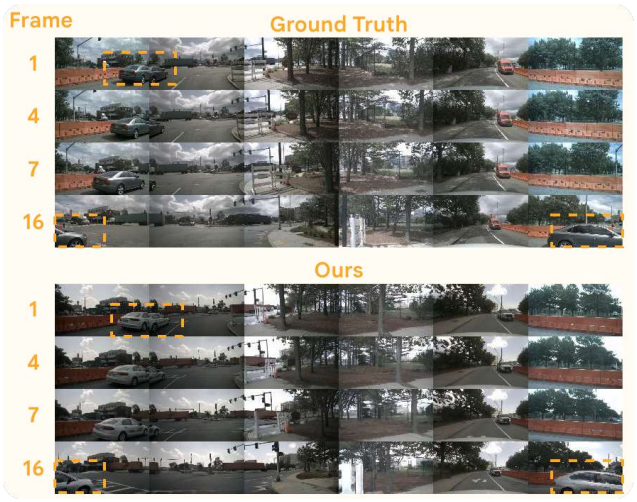

Figure 5. **Qualitative evaluation of video generation.** Our approach demonstrates controllable scene generation while maintaining robust cross-view and temporal consistency of objects.

Table 3. Evaluations of Video generation on the nuScenes dataset. Metrics are computed for 16-frame video clips.

Data Source	Resolution	Base Model	FVD ↓	mAP ↑	mIoU ↑
Ori nuScenes [4]	224×400	-	-	29.69	36.70
MagicDrive [7]	224×400	SD V1.5	218.12	11.86	18.34
MagicDrive3D [8]	224×400	SD V1.5	210.40	12.05	18.24
DreamForge [26]	224×400	SD V1.5	209.90	14.37	29.07
DreamForge	448×800	SD V1.5	233.20	**22.52**	32.98
DreamForge	448×800	3DVAE, DiT	103.61	19.17	34.36
MagicDriveDiT [9]	848×1600	3DVAE, DiT	94.84	18.17	20.40
Ours	448×800	3DVAE, DiT	**81.87**	18.35	**36.66**

Table 4. Evaluation of multi-view consistency for the video generation task, measured by the Key Points Matching (KPM) metric.

Method	KPM(%) ↑
Drive-WM [31]	45.8
Ours	**87.5**

tation performance.

Furthermore, we evaluate the multi-view consistency of the visual outputs utilizing the Key Points Matching (KPM) [31] metric. Computed on overlapping view pairs and averaged over frames, our approach achieves 87.5% KPM, considerably surpassing Drive-WM [31] (45.8%) and confirming the efficacy of our framework in maintaining robust cross-view alignment, as shown in Table 4.

As shown in Figure 5, our approach maintains object consistency across multiple views and ensures high temporal coherence. This approach produces stable and realistic sequences, whereas baselines frequently struggle with flickering artifacts or inconsistent object appearances over time.

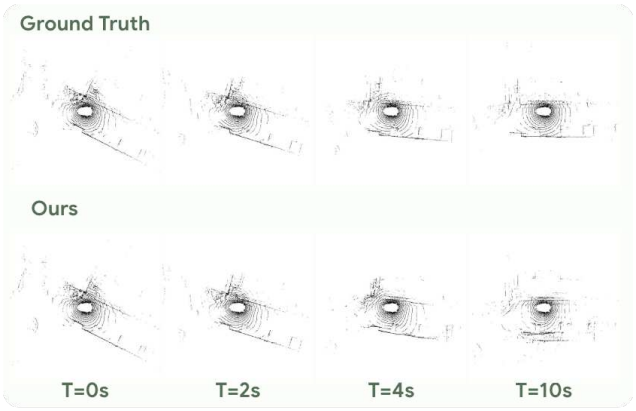

Figure 6. **Qualitative evaluation of LiDAR generation.** The proposed method enables the controllable generation of LiDAR point clouds, synthesizing high-fidelity scenes with exceptional temporal consistency.

Table 5. Evaluations of scene-level fidelity for LiDAR generation on the nuScenes dataset. Lower is better for all metrics (↓).

Method	JSD ($\times 10^{-4}$) ↓	MMD ($\times 10^{-2}$) ↓
UniScene [19]	31.55	13.61
OpenDWM [1]	20.17	5.61
OpenDWM-DiT [1]	19.90	5.73
Ours	**18.55**	**4.78**

LiDAR Generation Results. We further evaluate the scene-level fidelity of the LiDAR generation of our model on the nuScenes validation set, benchmarking against state-of-the-art methods, including UniScene [19], OpenDWM [1], and OpenDWM-DiT [1]. To ensure a fair comparison, all baseline results are cited from LiDARCrafter [23]. The quantitative comparison is presented in Table 5, where our model achieves the best scores across all metrics. Compared with the strongest baseline, OpenDWM-DiT, our model achieves a relative reduction of 6.8% on the JSD metric and a significant reduction of 16.6% on the MMD metric. These results demonstrate that the generated LiDAR point clouds possess a significant advantage in scene-level fidelity, exhibiting a data distribution that closely approximates real-world observations.

As shown in Figure 6, the qualitative results visually confirm the quantitative advantages. Our method controllably generates high-fidelity LiDAR scenes with superior temporal consistency.

Cross-modal Consistency. Beyond individual modalities, we evaluate the structural alignment across the generated multimodal outputs. To quantify video-LiDAR consistency, we employ the Depth Alignment Score (DAS) [33], which measures the mean absolute error (MAE) between projected LiDAR depths and video-estimated depths. As shown in

Table 6. Evaluation of Video-LiDAR cross-modal consistency, measured by the Depth Alignment Score (DAS).

Method	DAS ↓
X-Drive [33]	1.69
Ours	**1.16**

Table 7. Ablation study on the key components of the Occupancy Generation Model.

Method	mIoU ↑	F3D ↓	MMD ↓
Ours	**24.17**	**123.47**	**8.21**
w/. Vehicle Layout only	10.12	323.52	30.22
w/o. 3D Boxes	19.24	154.33	9.56
w/. DDIM	22.87	137.25	9.07

Table 6, our method achieves a DAS of 1.16, marking a significant improvement over the X-Drive [33] baseline (1.69). Notably, this reduced error indicates that our 4D occupancy scaffold effectively constrains the geometric scale across different sensors, mitigating cross-modal structural hallucinations.

4.1. Ablation Studies

Effect of Designs in Occupancy Generation Model. We conduct in-depth ablations on three key designs within the occupancy model (Table 7). First, we validate the HD map layout. The "w/ Vehicle Layout only" variant, replacing our full layout with a vehicle-only baseline [2], shows a severe performance drop: mIoU plummets by 58.1%, with F3D and MMD also deteriorating. This confirms that high-quality HD map layouts are essential for high-fidelity occupancy. Second, we assess 3D bounding box supervision. Removing this ("w/o. 3D Boxes") leads to a 20.4% mIoU drop and a 25.0% F3D increase, confirming 3D priors are crucial for learning dynamic structures. Finally, we compare generation paradigms. Replacing our Flow Matching [24] with DDIM [27] causes a comprehensive decline: mIoU decreases by 5.4% and F3D increases by 11.2%. This indicates Flow Matching better captures temporal dynamics for superior generation quality.

Effect of Designs in Video Generation Model. In our framework, the 4D semantic occupancy sequence acts as a critical spatiotemporal scaffold, providing explicit geometric and semantic guidance to enforce consistency in the generated video. To assess the impact of this occupancy guidance, we remove the 4D occupancy condition from the video generation module (denoted as "w/o. Occupancy MPI"). As shown in Table 8, removing the occupancy guidance leads to a significant degradation in generation quality. Specifically, the FVD score deteriorates by 7.65 points, reflecting a clear loss in video fidelity. Furthermore, the BEV segmentation performance drops by 1.84 points. This is a substantial drop, as the mIoU of our full model is nearly on par with real data. This result confirms that the 4D occupancy scaffold is crucial for the generation of geometrically accurate and semantically correct scene layouts. Although the "w/o. Occupancy MPI" variant shows a marginal increase in mAP, this increase comes at a significant cost to overall video fidelity and the semantic structural integrity of the scene.

Table 8. Ablation for video generation on the nuScenes dataset.

Method	FVD ↓	mAP ↑	mIoU ↑
Ours	**81.87**	18.35	**36.66**
w/o. Occupancy MPI	89.52	**19.17**	34.82

Table 9. Ablation for LiDAR generation on the nuScenes dataset. Lower is better for all metrics (↓).

Method	JSD ($\times 10^{-4}$) ↓	MMD ($\times 10^{-2}$) ↓
Ours	**18.55**	**4.78**
w/o. Occupancy BEV	19.90	5.73

Effect of Designs in LiDAR Generation Model. In our framework, the bird's-eye view (BEV) of the occupancy serves as a critical geometric condition to ensure that the generated point clouds align with the geometric structure of the scene. To evaluate the impact of this guidance, we remove this condition from the LiDAR generation module (denoted as "w/o. Occupancy BEV"). As shown in Table 9, removing the occupancy guidance leads to a noticeable decline in generation quality. Specifically, the JSD score deteriorates by 7.28%, and the MMD score increases by 5.02%. This performance degradation indicates that the point clouds generated by the "w/o. Occupancy BEV" variant exhibit a larger distributional divergence from the real data. This result confirms that the Occupancy BEV condition is crucial for guiding the LiDAR generator to produce geometrically accurate and realistic point clouds that are consistent with the predicted scene layout.

5. Conclusion and Future Work

We introduced Drive-Cascade, a unified autoregressive world model centered on 4D semantic occupancy as an explicit core dynamic state, decoupling generation into a state transition model for physical dynamics and parallel observation emission models for sensor outputs. Experimental results demonstrate high generation fidelity and fine-grained controllability.

Future Work. We will focus on two key directions: transitioning from open-loop control to a closed-loop interactive simulator accepting agent actions for policy training, and replacing our staged training pipeline with end-to-end joint fine-tuning to better optimize the joint latent space.

References

[1] Opendwm: Open driving world models, 2025. https://github.com/SenseTime-FVG/OpenDWM. 7

[2] Hengwei Bian, Lingdong Kong, Haozhe Xie, Liang Pan, Yu Qiao, and Ziwei Liu. Dynamiccity: Large-scale 4d occupancy generation from dynamic scenes. *arXiv preprint arXiv:2410.18084*, 2024. 2, 8

[3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2

[4] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1, 5, 7, 12

[5] Rui Chen, Zehuan Wu, Yichen Liu, Yuxin Guo, Jingcheng Ni, Haifeng Xia, and Siyu Xia. Unimlvg: Unified framework for multi-view long video generation with comprehensive control capabilities for autonomous driving. *arXiv preprint arXiv:2412.04842*, 2024. 5

[6] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017. 1

[7] 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. *arXiv preprint arXiv:2310.02601*, 2023. 2, 6, 7

[8] 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, 6, 7

[9] Ruiyuan Gao, Kai Chen, Bo Xiao, Lanqing Hong, Zhenguo Li, and Qiang Xu. Magicdrive-v2: High-resolution long video generation for autonomous driving with adaptive control. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28135–28144, 2025. 7

[10] 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. *Advances in Neural Information Processing Systems*, 37:91560–91596, 2024. 2

[11] Xiangyu Guo, Zhanqian Wu, Kaixin Xiong, Ziyang Xu, Lijun Zhou, Gangwei Xu, Shaoqing Xu, Haiyang Sun, Bing Wang, Guang Chen, et al. Genesis: Multimodal driving scene generation with spatio-temporal and cross-modal consistency. *arXiv preprint arXiv:2506.07497*, 2025. 2

[12] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3), 2018. 1

[13] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6, 14

[14] Yuenan Hou, Xinge Zhu, Yuexin Ma, Chen Change Loy, and Yikang Li. Point-to-voxel knowledge distillation for lidar semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8479–8488, 2022. 2

[15] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023. 1, 2

[16] Qianjiang Hu, Zhimin Zhang, and Wei Hu. Rangeldm: Fast realistic lidar point cloud generation. In *European Conference on Computer Vision*, pages 115–135. Springer, 2024. 2

[17] Xiaotao Hu, Wei Yin, Mingkai Jia, Junyuan Deng, Xiaoyang Guo, Qian Zhang, Xiaoxiao Long, and Ping Tan. Drivingworld: Constructing world model for autonomous driving via video gpt. *arXiv preprint arXiv:2412.19505*, 2024. 2

[18] Jiahui Huang, Zan Gojcic, Matan Atzmon, Or Litany, Sanja Fidler, and Francis Williams. Neural kernel surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4369–4379, 2023. 5, 12

[19] 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. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11971–11981, 2025. 1, 2, 5, 6, 7, 14

[20] Leheng Li, Weichao Qiu, Yingjie Cai, Xu Yan, Qing Lian, Bingbing Liu, and Ying-Cong Chen. Syntheocc: Synthesize geometric-controlled street view images through 3d semantic mpis. *arXiv preprint arXiv:2410.00337*, 2024. 2, 4

[21] Xiaofan Li, Yifu Zhang, and Xiaoqing Ye. Drivingdiffusion: layout-guided multi-view driving scenarios video generation with latent diffusion model. In *European Conference on Computer Vision*, pages 469–485. Springer, 2024. 1

[22] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 6, 14

[23] Ao Liang, Youquan Liu, Yu Yang, Dongyue Lu, Linfeng Li, Lingdong Kong, Huaici Zhao, and Wei Tsang Ooi. Lidarcrafter: Dynamic 4d world modeling from lidar sequences. *arXiv preprint arXiv:2508.03692*, 2025. 7, 14

[24] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022. 4, 8

[25] Yifan Lu, Xuanchi Ren, Jiawei Yang, Tianchang Shen, Zhangjie Wu, Jun Gao, Yue Wang, Siheng Chen, Mike Chen, Sanja Fidler, et al. Infinicube: Unbounded and controllable dynamic 3d driving scene generation with world-guided video models. In *Proceedings of the IEEE/CVF In-*

ternational Conference on Computer Vision, pages 27272–27283, 2025. 2

[26] Jianbiao Mei, Tao Hu, Xuemeng Yang, Licheng Wen, Yu Yang, Tiantian Wei, Yukai Ma, Min Dou, Botian Shi, and Yong Liu. Dreamforge: Motion-aware autoregressive video generation for multi-view driving scenes. *arXiv preprint arXiv:2409.04003*, 2024. 4, 6, 7

[27] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 4, 8

[28] Shuhan Tan, John Lambert, Hong Jeon, Sakshum Kulshrestha, Yijing Bai, Jing Luo, Dragomir Anguelov, Mingxing Tan, and Chiyu Max Jiang. Scenediffuser++: City-scale traffic simulation via a generative world model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1570–1580, 2025. 2

[29] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *Advances in Neural Information Processing Systems*, 36:64318–64330, 2023. 5, 12

[30] Lening Wang, Wenzhao Zheng, Yilong Ren, Han Jiang, Zhiyong Cui, Haiyang Yu, and Jiwen Lu. Occsora: 4d occupancy generation models as world simulators for autonomous driving. *arXiv preprint arXiv:2405.20337*, 2024. 5, 6

[31] 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, 2023. 7

[32] Wenming Weng, Ruoyu Feng, Yanhui Wang, Qi Dai, Chunyu Wang, Dacheng Yin, Zhiyuan Zhao, Kai Qiu, Jianmin Bao, Yuhui Yuan, et al. Art-v: Auto-regressive text-to-video generation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7395–7405, 2024. 3

[33] Yichen Xie, Chenfeng Xu, Chensheng Peng, Shuqi Zhao, Nhat Ho, Alexander T Pham, Mingyu Ding, Masayoshi Tomizuka, and Wei Zhan. X-drive: Cross-modality consistent multi-sensor data synthesis for driving scenarios. *arXiv preprint arXiv:2411.01123*, 2024. 2, 7, 8

[34] Yuwen Xiong, Wei-Chiu Ma, Jingkang Wang, and Raquel Urtasun. Ultralidar: Learning compact representations for lidar completion and generation. *arXiv preprint arXiv:2311.01448*, 2023. 5

[35] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*, 2024. 2

[36] Xuemeng Yang, Licheng Wen, Tiantian Wei, Yukai Ma, Jianbiao Mei, Xin Li, Wenjie Lei, Daocheng Fu, Pinlong Cai, Min Dou, et al. Drivearena: A closed-loop generative simulation platform for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26933–26943, 2025. 2

[37] Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, et al. Epona: Autoregressive diffusion world model for autonomous driving. *arXiv preprint arXiv:2506.24113*, 2025. 2

[38] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3836–3847, 2023. 4

[39] Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *European conference on computer vision*, pages 55–72. Springer, 2024. 2, 5, 6

[40] Zehao Zhu, Yuliang Zou, Chiyu Max Jiang, Bo Sun, Vincent Casser, Xiukun Huang, Jiahao Wang, Zhenpei Yang, Ruiqi Gao, Leonidas Guibas, et al. Scenecrafter: Controllable multi-view driving scene editing. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 6812–6822, 2025. 2

[41] Vlas Zyrianov, Xiyue Zhu, and Shenlong Wang. Learning to generate realistic lidar point clouds. In *European Conference on Computer Vision*, pages 17–35. Springer, 2022. 2

[42] Vlas Zyrianov, Henry Che, Zhijian Liu, and Shenlong Wang. Lidardm: Generative lidar simulation in a generated world. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6055–6062. IEEE, 2025. 2

Drive-Cascade: Autoregressive Occupancy to LiDAR and Video Synthesis

Supplementary Material

A. Implementation Details

In this section, we provide the detailed training configurations, hyperparameter settings, and sampling strategies for our Drive-Cascade framework.

A.1. Training Configuration

We implement the framework using PyTorch and conduct training on a server equipped with 8 $\times$ NVIDIA A100 GPUs (80GB). Given the computational complexity of multi-modal generation, we adopt a multi-stage training strategy. We train the three core modules—Occupancy Dreamer, LiDAR Dreamer, and Video Dreamer—independently, with the training process for each module requiring approximately one week. To enhance training efficiency and memory utilization, we employ Automatic Mixed Precision (AMP) throughout the training phase.

We utilize the AdamW optimizer with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, and a weight decay of 0.01. We set the initial learning rate to 1×10^{-4} and adjust it using a Cosine Annealing Scheduler after a warm-up period. To ensure training stability, we apply gradient clipping with a threshold of 1.0.

A.2. Preprocessing Configuration

Regarding data preprocessing, we align the nuScenes dataset to the model inputs by upsampling all data modalities to 12Hz. Specifically, we resize input video frames to a resolution of $H \times W = 448 \times 800$. For 3D occupancy prediction, we set the voxel resolution to 0.4m. Furthermore, to ensure a consistent geometric representation, we voxelize LiDAR point clouds within a spatial range of $[-50m, 50m]$ along the X and Y axes.

A.3. Inference and Sampling Details

During the inference phase, we employ distinct sampling strategies tailored to the characteristics of each modality. For the Occupancy Dreamer, which utilizes Flow Matching, we use the Euler ODE solver with 50 integration steps. For the Video and LiDAR Dreamers, we employ the DDIM sampler with 50 denoising steps to strike an optimal balance between generation fidelity and computational efficiency.

To enhance the alignment between the generated outputs and the conditioning signals (*e.g.*, HD maps, 3D boxes, and historical context), we utilize Classifier-Free Guidance (CFG). During training, we randomly drop the conditioning signals with a 10% probability to train the unconditional diffusion model. During inference, we empirically set the CFG scales to 4.5 for the Video Dreamer, 3.0 for the LiDAR Dreamer, and 2.5 for the Occupancy Dreamer. In the autoregressive generation phase, we set the historical context window k to 3 frames to provide sufficient temporal priors without overwhelming the cross-attention modules or exceeding memory constraints.

A.4. Conditioning Robustness Strategy

We utilize high-level controls, such as HD maps and 3D bounding boxes, to guide the scene layout. In practical data engine applications, these controls may be sourced from automated perception pipelines and inevitably contain moderate noise. To ensure robustness against such imperfections, we introduce controlled perturbations during the training phase. Specifically, we apply random translational noise ($\pm 0.5m$) and rotational noise ($\pm 5^\circ$) to the 3D bounding boxes with a 30% probability. Additionally, we randomly drop individual HD map elements (*e.g.*, a lane segment or pedestrian crossing) with a 10% probability. This robust training strategy prevents the model from overfitting to perfect annotations and compels the network to rely more heavily on the spatiotemporally coherent 4D occupancy scaffold, thereby improving our overall system resilience during extended autoregressive rollouts.

B. System Complexity and Efficiency

Table 10 outlines the parameter count, peak inference memory, and generation speed (FPS) for each component of our framework. We measure speed on a single A100 GPU representing per-frame, single-sample throughput. While our target application is an offline data engine rather than real-time deployment, the decoupled nature of the modules maintains practical generation efficiency and a manageable memory footprint.

Table 10. System complexity, memory footprint, and inference speed measured on a single A100 GPU.

Module	Params (B)	Memory (GB)	Inference (FPS)
Occ Dreamer	0.12	52	0.67
LiDAR Dreamer	1.29	62	0.22
Video Dreamer	2.48	65	0.18

C. Detailed Network Architectures

In this section, we provide a comprehensive elaboration on the architectural designs of the three core modules: the Occupancy Dreamer, the Video Dreamer, and the LiDAR Dreamer. We build all modules upon the Diffusion

Transformer (DiT) paradigm but incorporate distinct encoding and conditioning mechanisms tailored to their specific modalities.

C.1. Occupancy Dreamer

We task the Occupancy Dreamer with generating high-fidelity 4D semantic occupancy sequences. Directly modeling high-dimensional 4D voxel grids (time $\times$ height $\times$ width $\times$ depth) incurs prohibitive computational costs. To address this, we incorporate a HexPlane Variational Autoencoder (VAE) combined with a Flow Matching generative backbone. Specifically, the HexPlane VAE factorizes the dense 4D spatiotemporal volume into six orthogonal 2D feature planes (*e.g.*, XY, XT, YT). This decomposition effectively eliminates spatiotemporal redundancy and compresses the scene into compact latent embeddings.

In the latent space, we employ a DiT architecture to model the distribution. Unlike standard diffusion models that predict Gaussian noise, we adopt the optimal transport Flow Matching objective. By regressing the vector field that transforms the prior distribution to the data distribution, we construct straighter generation trajectories. This design not only accelerates the sampling process but also enhances the stability of generating complex dynamic scene structures.

C.2. Video Dreamer

The Video Dreamer functions as a conditional generative model, synthesizing multi-view driving videos that are strictly aligned with the underlying 3D geometry. We found the architecture on a spatiotemporal Video DiT, which processes video data as a sequence of flattened patches. To seamlessly integrate the 3D occupancy guidance into the 2D video generation process without disrupting the pre-trained visual priors, we utilize a ControlNet-based injection mechanism.

First, we project and encode the 4D semantic occupancy into bird's-eye-view (BEV) or perspective-view feature maps via a lightweight encoder. These encoded conditions are then passed to the ControlNet, which creates a trainable copy of the encoding layers of the DiT. We inject the structural guidance into the main branch through zero-convolution layers, ensuring that the generated visual elements—such as vehicles, lanes, and obstacles—spatially correspond to the occupancy layout while maintaining high photorealism.

C.3. LiDAR Dreamer

We design the LiDAR Dreamer to autoregressively generate realistic LiDAR point cloud sequences. Given the sparsity of point clouds compared to dense video or voxel data, we employ a specialized DiT architecture for point set generation, conditioned via a Feature Alignment Adapter.

Since the input condition (occupancy) is a dense volumetric representation while the target (LiDAR) is sparse and discrete, we consider direct concatenation suboptimal. The Adapter consists of a series of convolutional blocks and Multi-Layer Perceptrons (MLPs) acting as a bridge to align these two modalities. It extracts hierarchical geometric features from the occupancy grid and maps them into the latent space of the LiDAR DiT. Guided by these aligned features, the DiT denoises random latent codes to recover precise 3D point coordinates and intensity values, effectively synthesizing sensor-realistic LiDAR sweeps that accurately reflect the scene geometry.

D. Data Processing

D.1. nuScenes Dataset

We conduct all experiments on the nuScenes dataset [4], a large-scale autonomous driving benchmark comprising 1000 driving scenes, each spanning 20 seconds. The dataset is officially partitioned into 700 scenes for training, 150 for validation, and 150 for testing. Each scene is equipped with comprehensive sensor data, including six surround-view cameras, one 32-beam LiDAR, and high-definition (HD) maps.

D.2. Data Preprocessing and Upsampling

Standard semantic occupancy annotations provided by nuScenes-Occupancy [29] are limited to 2Hz keyframes. However, effective autonomous driving simulation necessitates a higher temporal resolution to accurately capture fine-grained dynamic object motion. To bridge this gap and achieve a 12Hz frame rate, we employ the Neural Kernel Surface Reconstruction (NKSR) [18] technique to upsample the occupancy labels.

The upsampling pipeline operates through three sequential stages. First, during the sparsification phase, we convert the dense voxel occupancy grids of the original keyframes into sparse point clouds while strictly retaining their semantic labels. Subsequently, we utilize NKSR to learn a continuous implicit surface representation from these sparse points, which allows for the interpolation of geometric structures between discrete keyframes. Finally, in the resampling stage, we query this reconstructed implicit surface at intermediate timestamps to generate dense occupancy grids at the target 12Hz frequency. This rigorous process yields a high-frequency, temporally consistent 4D semantic occupancy dataset, serving as the high-quality ground truth for training our Occupancy Dreamer.

E. Evaluation Metrics

We comprehensively evaluate our Drive-Cascade framework across three modalities: occupancy, video, and LiDAR. The specific metrics and their evaluation protocols

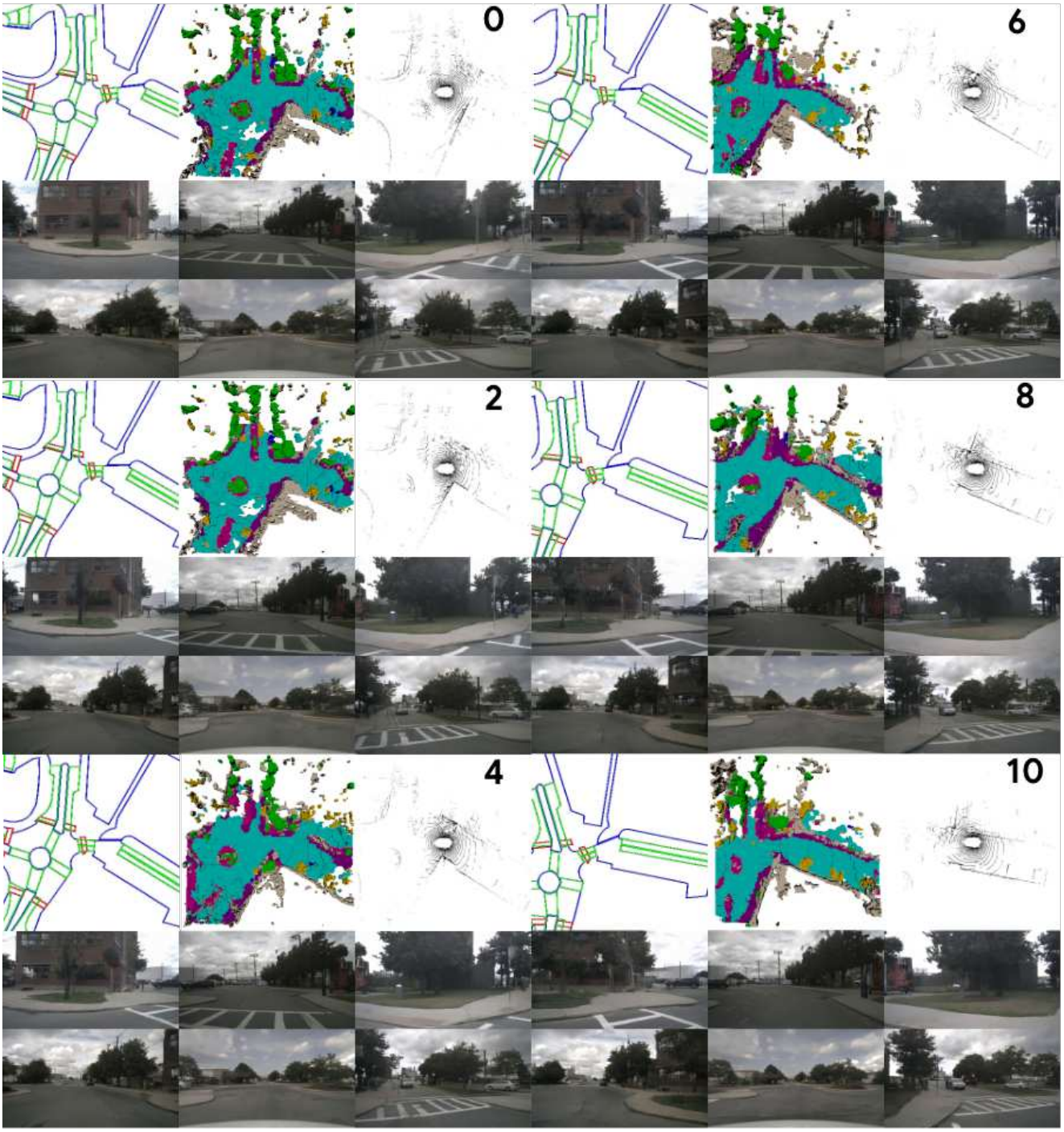

Figure 7. Multimodal Scene Generation. Conditioned on inputs such as HD maps, our framework generates three modalities: Occupancy, LiDAR, and Video.

are detailed below.

E.1. Occupancy Evaluation

To rigorously assess the quality of our generated 4D occupancy, we employ a combination of semantic and distribution-based metrics. First, we use mIoU (Mean Intersection over Union) as the primary metric for semantic segmentation accuracy. We calculate the IoU for each semantic class (excluding the 'empty' class) and average them; a high mIoU indicates that our generated geometry

and semantics closely align with the ground truth. Second, to measure generation fidelity, we compute F3D (Fréchet 3D Distance). Similar to FID in 2D images, F3D measures the distance between the feature distributions of generated and real 3D occupancy, with lower values indicating higher fidelity. Finally, we calculate MMD (Maximum Mean Discrepancy) to quantify the distributional distance between our generated and ground-truth occupancy sequences, focusing specifically on temporal dynamics and physical plausibility.

E.2. Video Generation Evaluation

We evaluate our video generation quality from two complementary perspectives: visual fidelity and perception performance (perception-as-a-metric). For visual quality, we employ FVD (Fréchet Video Distance) [13], which evaluates both visual realism and temporal coherence using an I3D network pre-trained on the Kinetics-400 dataset. A lower FVD score implies that our generated videos are smoother and more realistic. To validate whether the generated videos are physically meaningful for downstream tasks, we utilize perception metrics (mAP and mIoU). Using a pre-trained BEVFormer [22] as an oracle detector, we calculate the mAP for 3D object detection and mIoU for BEV map segmentation. High scores on these metrics indicate that our generated videos preserve accurate semantic layouts and object features recognizable by state-of-the-art perception models.

E.3. LiDAR Generation Evaluation

Following established protocols in previous works [19, 23], we evaluate the scene-level fidelity of our generated point clouds using two statistical metrics. We first compute the JSD (Jensen-Shannon Divergence) between the statistical distributions of our generated point clouds and the real validation set to evaluate how well the generated data covers the real-world distribution. Additionally, we calculate MMD (Maximum Mean Discrepancy) on the point cloud sets to measure the similarity between our generated and real LiDAR frames in the feature space.

F. Long-Horizon Stability

To evaluate our model capacity for extended generation without structural drift, we use video as a proxy for long-horizon stability due to the tight coupling across our modalities. We measured Fréchet Video Distance (FVD) across 5, 10, and 20-second clips (approaching the maximum nuScenes sequence length). As shown in Table 11, FVD increases gradually (87.47 $\rightarrow$ 101.17 $\rightarrow$ 113.74) without any abrupt collapse. This robust long-horizon capability suggests that our proposed 4D occupancy scaffold effectively mitigates error accumulation over extended autoregressive rollouts.

Table 11. Long-horizon stability evaluation using FVD over different clip lengths.

Method	$\text{FVD}_5 \downarrow$	$\text{FVD}_{10} \downarrow$	$\text{FVD}_{20} \downarrow$
Ours	**87.47**	**101.17**	**113.74**

G. Additional Qualitative Results

As shown in Figure 7, we present a sequence of generated multi-frame data within a representative driving scene. Our visualization demonstrates superior multi-modal consistency, where the geometric structures in our generated LiDAR and occupancy perfectly align with the visual semantics in the video frames. Furthermore, our results exhibit excellent long-term temporal consistency, maintaining smooth object motion and stable background details across the sequence without flickering artifacts.

H. Limitations and Broader Impact

Limitations. While our Drive-Cascade demonstrates superior performance in generating high-fidelity multi-modal data, the iterative nature of the diffusion process results in substantial inference latency. This currently constrains our applicability in real-time simulation scenarios.

Broader Impact. Our work contributes to autonomous driving simulation, potentially reducing the cost of real-world data collection. However, synthesized data should be carefully validated before being used to train safety-critical perception modules to avoid domain gap issues.