UNIVERSE: Unified Video Action Models for Autonomous Driving with Flexible Mask-Modulated Modality Generation

Mengmeng Liu[1] **Diankun Zhang**[2] **Jiuming Liu**[3] **Jianfeng Cui**[2]
Hongwei Xie[2] **Guang Chen**[2] **Hangjun Ye**[2]
Francesco Nex[1] **Hao Cheng**[1] **Michael Ying Yang**[4,*]

[1]University of Twente [2]Xiaomi EV
[3]University of Cambridge [4]University of Bath
[*]Corresponding author.

Abstract

World Action Models (WAMs) have shown strong potential for improving action generalization in autonomous driving by using future video prediction as dense supervision for scene dynamics. However, retaining this supervision without requiring expensive future-video generation at deployment remains an architectural challenge. Existing designs either separate video and action parameters or couple their future-token dependencies during joint denoising. We propose UNIVERSE, a unified video-action model built upon a *single mask-modulated Diffusion Transformer*. It retains shared video-action parameters so that dense video supervision directly shapes trajectory denoising, while introducing a *Modality-Decoupling Visibility Mask* that shares historical context and blocks mutual attention between future video and trajectory tokens. This makes future-video generation optional: removing future-video tokens at test time preserves the trajectory branch's training-time attention dependencies, enabling trajectory-only inference with a $4.3\times$ speedup over coupled video-action rollout and comparable planning accuracy. The same model also supports video-only and joint video-action rollouts. Experiments show that UNIVERSE achieves 91.0 PDMS on NAVSIM (vs. 89.6 for the Two-DiT variant) and strong zero-shot transfer to nuScenes and Bench2Drive without fine-tuning. Controlled comparisons and ablations on pretraining and loss weighting support the roles of shared parameters, video co-training, and mask-based modality decoupling.

Code: `https://github.com/xiaomi-mlab/UNIVERSE`

1 Introduction

Generalization to unseen conditions is a central challenge in visual learning [1, 2, 3, 4, 5] and autonomous driving [6, 7, 8, 9, 10, 11, 12, 13, 14, 15], where a planner trained on one dataset must safely transfer to unseen cities, traffic patterns, sensor distributions, and simulation environments. Standard end-to-end planners directly map historical observations to future ego trajectories, which is efficient but can overfit dataset-specific driving priors such as common trajectory patterns, road layouts, or benchmark-specific behaviors [16, 17, 18]. World Action Models (WAMs) offer a promising alternative by coupling future prediction with action generation [19]. Compared with sparse trajectory supervision, future world modeling provides dense signals about scene geometry, object motion, traffic-agent interactions, and temporal causality, helping the model learn both *how*

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

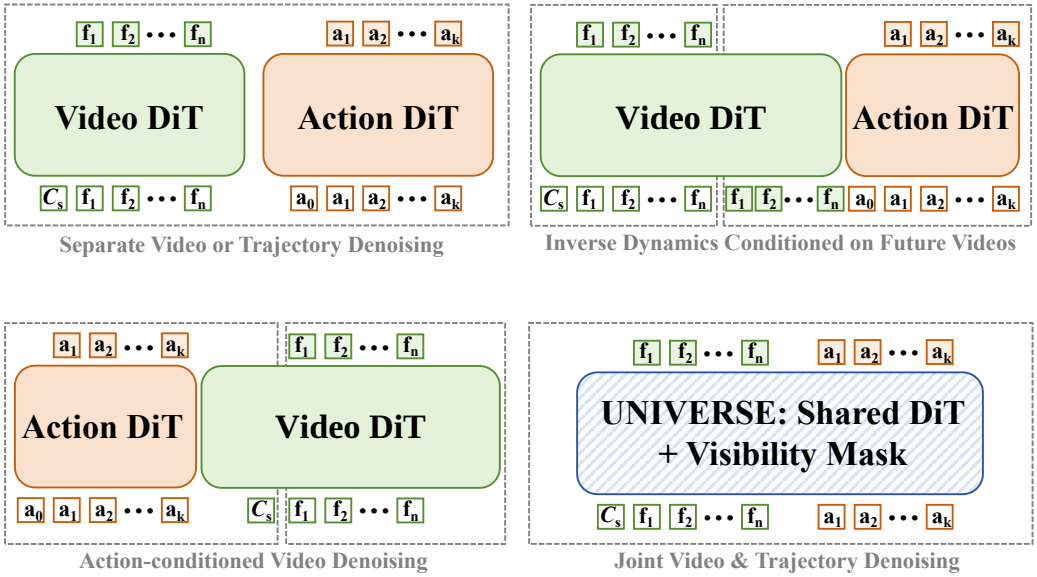


Figure 1: **Video-action denoising dependency patterns.** C_s denotes clean history and current ego-state conditioning; f_i and a_i denote video-frame and trajectory tokens. The first three panels illustrate separate or conditionally coupled modules; the last shows UNIVERSE's shared DiT with future-modality visibility decoupling.

the world evolves and *how the ego vehicle should act* [20, 21, 22, 23]. Recent driving world models and embodied video-action models further show that future visual prediction can improve planning, simulation, transferability, and zero-shot policy generalization [24, 19, 25, 26, 27, 28, 29].

A key question is whether these training benefits require expensive future-video generation at deployment. DriveVA [30], the direct predecessor of this work, establishes the feasibility of zero-shot driving through joint video-trajectory denoising in a shared DiT; its future video and trajectory tokens interact directly. Fast-WAM [31] instead shows that future-video prediction can serve as training-time supervision while action inference bypasses video rollout. These findings motivate a complementary question: *how can video supervision train the same parameters used for trajectory generation while future-video generation remains optional?* Cascaded or dual-DiT designs separate video and action parameters [32, 31, 33, 34], whereas a coupled shared-DiT design requires future-video tokens in the trajectory computation graph. Simply deleting those tokens at deployment changes the dependencies learned during training.

We propose UNIVERSE, a unified video-action model built upon a *single mask-modulated Diffusion Transformer* [35]. It retains the shared video-action backbone so that both losses update the same attention and feed-forward parameters, rather than regularizing a separate trajectory model indirectly. Our new design concerns the *future-token dependency structure*, not the use of a pretrained video backbone or shared DiT alone. The *Modality-Decoupling Visibility Mask* allows both future modalities to read common historical context, blocks mutual attention between future video and trajectory tokens, and prevents context tokens from relaying information from either future modality. This yields *coupling-by-parameters* and *decoupling-by-visibility*: training retains dense video supervision, while either future modality can be instantiated independently at inference.

Trajectory-only inference therefore removes future-video tokens without changing the trajectory branch's attention dependencies, reducing latency by $4.3\times$ relative to coupled rollout. The same weights support video-only and joint outputs for future-scene visualization and video-trajectory consistency analysis. Controlled no-video, no-mask, and Two-DiT experiments distinguish the roles of supervision, visibility, and parameter sharing. The cross-domain results support improved transfer; they do not by themselves establish that source-domain action priors are the sole cause of generalization errors.

Our contributions are:

- A modality-decoupled shared-DiT formulation that preserves video supervision of trajectory-generation parameters while making future-video generation optional at deployment.
- A parameter-free Modality-Decoupling Visibility Mask and output-dependent token instantiation that support dependency-consistent trajectory-only, video-only, and joint inference with the same model weights.
- An empirical study of planning, cross-domain transfer, video-action consistency, and computational cost, including comparisons with close prior methods and controlled variants. UNIVERSE achieves 91.0 NAVSIM PDMS, strong zero-shot transfer, and $4.3\times$ faster trajectory inference than coupled rollout.

2 Related Work

2.1 World Models for Autonomous Driving Video Generation

World modeling [36, 37, 38] has become a major direction for autonomous-driving video generation, aiming to synthesize temporally coherent and controllable future observations for simulation, data augmentation, and downstream decision-making. Early works such as DriveGAN [39] and MagicDrive [40] explore controllable driving-scene generation. Recent methods further improve multi-view consistency, 3D geometric control, long-horizon generation, and scenario diversity. Representative examples include DriveDreamer [41], DriveDreamer-2 [42], Panacea [43], VISTA [24], GAIA-1 [44], GAIA-2 [45], MagicDrive-V2 [46], MiLA [47], CoGen [48], Genesis [49], and OmniNWM [50]. Another line of work studies world models in BEV, occupancy, LiDAR, or 4D representations, such as OccWorld [51], DriveWorld [52], DrivingWorld [53], WoVoGen [54], UniScene [55], and related occupancy-centric driving simulators. These methods have significantly advanced the fidelity, controllability, and temporal consistency of driving-scene generation. However, they are primarily designed as scene simulators or data generators, with the main objective of improving visual realism, geometric consistency, or controllable rollout. Less attention has been paid to how video-learned world priors can be effectively transferred to trajectory generation for action generalization. In contrast, our work takes a planning-oriented perspective: UNIVERSE uses future video prediction as dense training-time supervision for trajectory generation, rather than treating video generation only as a standalone simulation task.

2.2 World–Action Models for Autonomous Driving Control

Beyond pure scene generation, world models have increasingly been incorporated into autonomous-driving planning and control. Representative methods include Driving into the Future [22], DrivingGPT [56], latent-world-model-assisted driving [57], Doe-1 [58], ReSim [59], WoTE [60], DriveLaW [61], Epona [62], Policy World Model [63], and DriveVLA-W0 [64]. Beyond these driving models, video-based robotic planners and policies also exploit visual dynamics for transferable control [65, 66, 67]. Meanwhile, recent embodied WAMs and video-action models, such as DreamZero [19], GigaWorld-Policy [33], LingBot-VA [26], Unified Video Action Model [27], Unified World Models [28], DreamGen [25], VideoVLA [68], and dual-stream diffusion [69], further suggest that modeling future observations can improve robot control and generalization beyond the training domain. Existing WAMs differ mainly in how future video prediction is coupled with control. Dual-stream diffusion [69] maintains modality-specific pathways linked by shared cross-modal attention. Imagine-then-plan methods retain video rollout in the online loop, offering interpretable future imagination but incurring diffusion or autoregressive latency. DriveVA [30] jointly denoises future videos and ego trajectories in a shared DiT with direct future-modality interaction. Fast-WAM [31] uses a pretrained video DiT and a separately initialized action-expert DiT, while Unified Video Action Model [27] uses a joint video-action latent representation with separate lightweight diffusion heads. UNIVERSE builds on DriveVA's shared-parameter foundation but redesigns its future-token dependencies with a visibility mask, enabling independent output modes without requiring future-video tokens for trajectory inference.

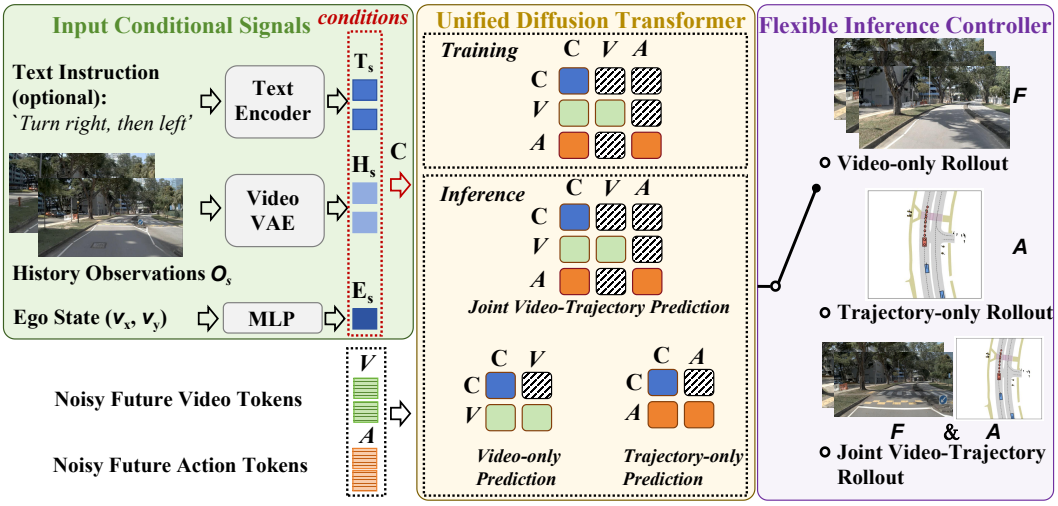


Figure 2: **Overall pipeline of UNIVERSE.** Given history observations, ego velocity (v_x, v_y), and language instructions, UNIVERSE encodes conditional tokens with a text encoder and video VAE [70]. A unified DiT [35] jointly trains future video latents and action tokens, while the Modality-Decoupling Visibility Mask blocks future video-action attention but preserves shared historical context. The model supports joint, video-only, and trajectory-only prediction.

3 Method

3.1 Unified Diffusion Transformer

Problem Definition. As shown in Fig. 2, given l-frame history observations $\mathcal{O}_s = \{\mathbf{F}_{s-l+1}, \ldots, \mathbf{F}_s\}$, current ego velocity (v_x, v_y), and optional text instruction $\mathcal{T}$, UNIVERSE predicts future actions and future visual rollouts either jointly or as a single modality. The outputs are defined as

$$\mathcal{A}_{s+1:s+K} = \{\boldsymbol{a}_{s+i} \in \mathbb{R}^3\}_{i=1}^{K}, \qquad \mathcal{F}_{s+1:s+N} = \{\mathbf{F}_{s+j}\}_{j=1}^{N}, \tag{1}$$

where each action $\boldsymbol{a}_{s+i}$ encodes the ego-vehicle (x, y) position and yaw angle, and $\mathcal{F}_{s+1:s+N}$ describes the anticipated visual evolution of the same driving segment. In practice, we predict latent visual and action representations following [62], without using history trajectories as input.

Architecture. A frozen text encoder from Wan2.2-TI2V-5B [70] encodes the instruction $\mathcal{T}$ into latent tokens $\mathbf{T}_s \in \mathbb{R}^{L_T \times d}$. A 3D-causal VAE adapted from Wan2.2-TI2V-5B encodes history observations into visual latents $\mathcal{H}_s^{\text{his}} = \{\mathbf{H}_{s-l+1}, \ldots, \mathbf{H}_s\}$, where $\mathbf{H} \in \mathbb{R}^{L_V \times d}$. The ego velocity is embedded by an MLP, and the self-attention context tokens are constructed as

$$\mathbf{T}_s = \text{Enc}_{\text{text}}(\mathcal{T}), \qquad \mathcal{H}_s^{\text{his}} = \text{VAE}(\mathcal{O}_s), \qquad \mathbf{E}_s = \text{MLP}(v_x, v_y), \qquad \mathcal{C}_s = \mathcal{H}_s^{\text{his}} \cup \mathbf{E}_s. \tag{2}$$

Text tokens $\mathbf{T}_s$ are injected through cross-attention.

For generation, future video latent tokens $\mathcal{V}_{s+1:s+N}$ and future trajectory tokens $\mathcal{A}_{s+1:s+K}$ are projected into the same hidden dimension d with modality-specific input projections, modality embeddings, and temporal positional embeddings. The DiT blocks, including self-attention and feed-forward layers, are shared by both modalities, while only the input projections and final prediction heads are modality-specific. The unified prediction process is written as

$$(\hat{\mathbf{v}}^{\text{vid}}, \hat{\mathbf{v}}^{\text{act}}) = \big(h_{\text{vid}}(\mathbf{Z}^{\text{vid}}), h_{\text{act}}(\mathbf{Z}^{\text{act}})\big), \quad [\mathbf{Z}^{\text{vid}}, \mathbf{Z}^{\text{act}}] = \text{DiT}_\theta\big(\mathcal{V}_{s+1:s+N}, \mathcal{A}_{s+1:s+K}, \mathcal{C}_s, \mathbf{T}_s\big), \tag{3}$$

where h_{vid} and h_{act} are modality-specific prediction heads for future video latents and trajectory tokens, respectively. Thus, future-video prediction updates the same generative backbone used for trajectory generation instead of regularizing a separated planner.

3.2 Modality-Decoupling Visibility Mask

Generating future videos and trajectories in one transformer can create *future-modality dependence*: trajectory tokens may rely on future visual tokens absent during trajectory-only inference, while

video tokens may depend on future trajectory tokens. The issue is a training–inference dependency mismatch, rather than access to clean future ground truth at test time. We therefore introduce a binary self-attention mask $M \in \{0,1\}^{L\times L}$ over context tokens $\mathcal{C}_s$, future video tokens $\mathcal{V}$, and future trajectory tokens $\mathcal{A}$, where $M(i,j)=1$ allows token i to attend to token j and $M(i,j)=0$ blocks this attention. Text tokens are excluded since they are injected through cross-attention.

The mask allows future video and trajectory tokens to attend to historical context, i.e., $M(i,j)=1$ for $i \in \mathcal{V}\cup\mathcal{A}$ and $j \in \mathcal{C}_s$, but prevents context tokens from attending back to future targets. It also blocks future video-action attention in both directions, i.e., $M(i,j)=0$ for $i \in \mathcal{V}, j \in \mathcal{A}$ or $i \in \mathcal{A}, j \in \mathcal{V}$, while preserving full attention within $\mathcal{C}_s$, $\mathcal{V}$, and $\mathcal{A}$. This design shares historical context across modalities, blocks cross-modality future-token dependencies, and maintains intra-modality coherence for video generation and trajectory denoising.

3.3 Flexible Inference Controller

The proposed mask enables flexible inference by instantiating only the requested future token groups. For trajectory-only rollout, future video tokens are removed and the model denoises $\mathcal{A}_{s+1:s+K}$ conditioned on $\mathcal{C}_s$ and $\mathbf{T}_s$. For video-only rollout, future trajectory tokens are removed and the model denoises $\mathcal{V}_{s+1:s+N}$ from the same conditions. Because trajectory tokens are already blocked from future video tokens during training, removing the video branch at inference does not change their attention dependency, allowing trajectory-only inference to bypass future-video diffusion and improve efficiency by $4.3\times$.

For joint video-action rollout, both future token groups are instantiated and denoised within the same shared DiT under the Modality-Decoupling Visibility Mask. Although future video and trajectory tokens remain mutually blocked, they are aligned through the shared denoising backbone, common historical context, and joint co-training, enabling interpretable rollout while preserving the attention dependencies used for trajectory-only deployment.

3.4 Training Objective

We train UNIVERSE with flow matching on both future trajectory tokens and future video latents. For a clean target sample $\mathbf{Y}_0^m$ from modality $m \in \{\mathrm{act}, \mathrm{vid}\}$, we sample a flow time $p \sim \mathcal{U}(0,1)$ and Gaussian noise $\boldsymbol{\epsilon}^m \sim \mathcal{N}(0, I)$. The noisy sample and target velocity are

$$\mathbf{Y}_p^m = (1-p)\boldsymbol{\epsilon}^m + p\mathbf{Y}_0^m, \qquad \dot{\mathbf{Y}}_p^m = \mathbf{Y}_0^m - \boldsymbol{\epsilon}^m. \tag{4}$$

The noisy future trajectory and video tokens are jointly fed into the shared DiT with the visibility mask:

$$(\hat{\mathbf{v}}_\theta^{\mathrm{act}}, \hat{\mathbf{v}}_\theta^{\mathrm{vid}}) = f_\theta(\mathbf{Y}_p^{\mathrm{act}}, \mathbf{Y}_p^{\mathrm{vid}}, p, \mathcal{C}_s, \mathbf{T}_s; M), \tag{5}$$

where f_θ denotes the shared DiT with modality-specific prediction heads. The final objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{act}} + \lambda_{\mathrm{vid}}\mathcal{L}_{\mathrm{vid}}, \qquad \mathcal{L}_m = \mathbb{E}\left[\left\|\hat{\mathbf{v}}_\theta^m - \dot{\mathbf{Y}}_p^m\right\|_2^2\right], \quad m \in \{\mathrm{act}, \mathrm{vid}\}. \tag{6}$$

Each modality loss is independently averaged over its batch and token dimensions before weighting, so the number of video tokens does not directly determine its contribution. We use $\lambda_{\mathrm{vid}} = 1.0$ by default; Sec. 4.6 reports sensitivity to 0.5 and 2.0. Both objectives update the same DiT parameters.

3.5 UNIVERSE Variants

We instantiate several variants to isolate the effects of video co-training, mask-based modality decoupling, and shared DiT parameterization. All variants use the same observations, trajectory targets, tokenization, training schedule, and evaluation protocol unless otherwise specified.

UNIVERSE. The full model co-trains future video latents and trajectory tokens within one shared DiT under the Modality-Decoupling Visibility Mask.

UNIVERSE w.o. Mask. To instantiate a controlled DriveVA-style coupled dependency structure [30], we remove the Modality-Decoupling Visibility Mask, allowing future video and trajectory tokens to attend to each other. This keeps the unified DiT and joint objective, but requires coupled video-action generation at inference.

UNIVERSE w.o. video co-train. This variant keeps the architecture, tokenization, mask, and inference procedure unchanged, but removes video supervision by setting $\lambda_{\text{vid}} = 0$, namely $\mathcal{L} = \mathcal{L}_{\text{act}}$.

UNIVERSE Two-DiT. Following Fast-WAM [31], we replace the shared DiT with two separated DiTs for video and trajectory denoising. They share the same conditions and losses but not transformer parameters, preserving video co-training while removing shared video-action parameterization.

4 Experiments

4.1 Datasets and Implementation Details

NAVSIM v1. Following prior works [62, 61], we adopt NAVSIM v1 [71], built upon OpenScene [72], as our main non-reactive simulation-based planning benchmark. It reports No Collision (NC), Drivable Area Compliance (DAC), Time-to-Collision (TTC), Comfort (C.), and Ego Progress (EP), aggregated as $\text{PDMS} = \text{NC} \times \text{DAC} \times (5\text{EP} + 5\text{TTC} + 2\text{C.})/12$. Unless otherwise specified, our model only uses front-view camera input without map, box, depth, occupancy, or LiDAR supervision.

nuScenes. For cross-dataset evaluation, we directly test the NAVSIM-trained model on the nuScenes validation split of 150 scenes from the official 1,000-scene dataset [73]. Following prior planning works [16, 74], we report L2 displacement error and collision rate at 1 s, 2 s, and 3 s horizons. We also report FID and FVD for future-video generation quality following driving world-model literature [41, 24].

Bench2Drive. Bench2Drive [75] is a CARLA v2 benchmark [76] with diverse interactive scenarios and routes. We use it for real-to-simulation zero-shot transfer by directly evaluating the NAVSIM-trained model on its validation split, where the model must handle shifts in visual appearance, vehicle dynamics, and agent behaviors [77].

Implementation Details. Training is conducted on NVIDIA H20 GPUs with AdamW, a learning rate of 10^{-4}, weight decay of 0.01, and distributed bf16 mixed-precision training. We first train with a batch size of 80 for 20k steps for fast convergence, and then continue training for another 10k steps with an effective batch size of 640 using gradient accumulation. We use a 1k-step linear warm-up from 10^{-3} of the base learning rate, followed by a constant learning-rate schedule. The training objective combines a flow-matching loss for future video supervision and a trajectory prediction loss. During inference, we use the trajectory-only rollout mode with 2 flow sampling steps unless otherwise stated. Wan2.2-TI2V-5B [70] is used as the pre-trained video generation backbone. Each training sample contains $l = 4$ history frames and $N = 8$ future frames at a resolution of 832×480. The DiT attention/FFN parameters, modality-specific input projections, and prediction heads are trainable; the text encoder and VAE are frozen. All zero-shot results use one NAVSIM-trained checkpoint without target-domain fine-tuning or adaptation. The Bench2Drive L2 and collision results are open-loop; input settings and preprocessing are detailed in Sec. D.

4.2 Comparison Results on NAVSIM v1

Table 1 reports non-reactive simulation-based planning results on NAVSIM v1. UNIVERSE achieves a PDMS of 91.0, outperforming prior traditional end-to-end planners and recent world-model-based planners. Compared with DiffusionDrive [18], our method improves PDMS from 88.1 to 91.0. Compared with WoTE [60], which uses both camera and LiDAR inputs, UNIVERSE still obtains a higher PDMS using only the front-view camera. Compared with DriveLaW [61], UNIVERSE improves PDMS from 89.1 to 91.0, with higher DAC, TTC, and EP scores under non-reactive simulation. DriveVA [30] achieves 90.9 PDMS, compared with 91.0 for UNIVERSE.

These results support our central hypothesis: future-video supervision is more effective for planning when it directly updates the same generative parameters used for trajectory denoising. Instead of using future videos only as an auxiliary output or a separate imagination module, UNIVERSE co-trains video latents and trajectory tokens in one shared DiT, allowing dense visual dynamics to regularize action generation. The UNIVERSE variant rows are included to preview the controlled comparison results; since these variants are designed to isolate architectural factors rather than serve as standalone baselines, we defer their detailed discussion to Sec. 4.5.

Table 1: **Performance comparison on NAVSIM *Navtest* using non-reactive simulation-based metrics.** Methods are grouped by whether they employ an explicit world model: *Traditional End-to-End Methods* and *World Model Methods*.

Method	Ref	Image	Lidar	NC↑	DAC↑	TTC↑	Comf.↑	EP↑	PDMS↑
Traditional End-to-End Methods									
VADv2-$\mathcal{V}_{8192}$ [78]	ICLR'26	✓		97.2	89.1	91.6	**100**	76.0	80.9
UniAD [16]	CVPR'23	✓		97.8	91.9	92.9	**100**	78.8	83.4
TransFuser [17]	TPAMI'23	✓	✓	97.7	92.8	92.8	**100**	79.2	84.0
PARA-Drive [79]	CVPR'24	✓		97.9	92.4	93.0	99.8	79.3	84.0
ReCogDrive-IL [80]	ICLR'26	✓		98.1	94.7	94.2	**100**	80.9	86.5
DiffusionDrive [18]	CVPR'25	✓	✓	98.2	96.2	94.7	**100**	82.2	88.1
World Model Methods									
DrivingGPT [56]	ICCV'25	✓		98.9	90.7	94.9	95.6	79.7	82.4
LAW [57]	ICLR'25	✓		96.4	95.4	88.7	99.9	81.7	84.6
Epona [62]	ICCV'25	✓		97.9	95.1	93.8	99.9	80.4	86.2
Resim [59]	NeurIPS'25	✓		–	–	–	–	–	86.6
WoTE [60]	ICCV'25	✓	✓	98.5	96.8	94.9	99.9	81.9	88.3
DriveVLA-W0 [64]	ICLR'26	✓		98.4	95.3	95.2	**100**	80.9	87.2
PWM [63]	NeurIPS'25	✓		98.6	95.9	95.4	**100**	81.8	88.1
DriveLaW [61]	CVPR'26	✓		99.0	97.1	96.7	**100**	81.3	89.1
DriveVA [30]	ECCV'26	✓		**99.2**	97.5	**98.7**	**100**	83.5	90.9
UNIVERSE Variants									
UNIVERSE	-	✓		99.1	**97.6**	98.5	**100**	**83.6**	**91.0**
UNIVERSE w.o. Mask	-	✓		**99.2**	97.5	**98.7**	**100**	83.5	90.9
UNIVERSE w.o. video co-train	-	✓		98.8	95.5	98.2	99.7	80.4	88.2
UNIVERSE Two-DiT	-	✓		98.9	96.2	98.3	99.9	82.7	89.6

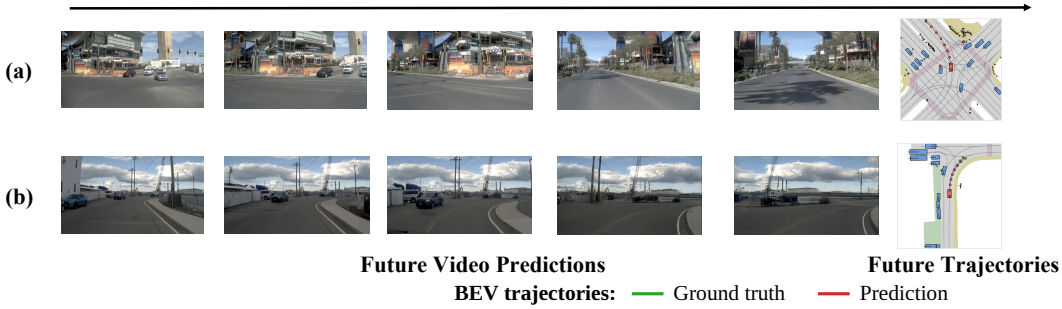


Figure 3: **Qualitative future video and trajectory generation on NAVSIM v1.** The rightmost BEV panels compare the ground-truth trajectory (green) and prediction (red); the video panels show predictions rather than paired ground-truth video rows. Paired ground-truth/generated video examples and DPVO [81] analysis are provided in Secs. E and G.

4.3 Zero-Shot Generalization on nuScenes and Bench2Drive

From NAVSIM to nuScenes. To evaluate cross-dataset action generalization, we directly test the NAVSIM-trained UNIVERSE on nuScenes without any fine-tuning. As shown in Table 2, UNIVERSE achieves an average L2 error of 0.83 m and an average collision rate of 0.03%. Compared with PWM [63], which is also trained on NAVSIM and evaluated zero-shot, UNIVERSE reduces the average L2 error from 3.99 m to 0.83 m and the average collision rate from 0.38% to 0.03%. Compared with DriveVLA-W0 [64], UNIVERSE also improves the average L2 error from 1.43 m to 0.83 m and the average collision rate from 0.77% to 0.03%. These results support improved transfer beyond NAVSIM; a source-domain fitting analysis and its interpretive limits are given in Sec. C.2.

From NAVSIM to Bench2Drive. We further evaluate zero-shot transfer from real-world logs to simulation on Bench2Drive. As reported in Table 2, UNIVERSE obtains an average L2 error of 1.33 m and an average collision rate of 1.19%. Compared with PWM [63], UNIVERSE reduces the average L2 error from 2.80 m to 1.33 m and the average collision rate from 3.76% to 1.19%. Fig. 4 further provides a qualitative zero-shot example: when a stationary vehicle blocks the lane ahead, UNIVERSE predicts a reasonable bypass maneuver, while the Two-DiT variant tends to stop behind the obstacle. This indicates stronger driving-intention understanding under real-to-simulation domain shift.

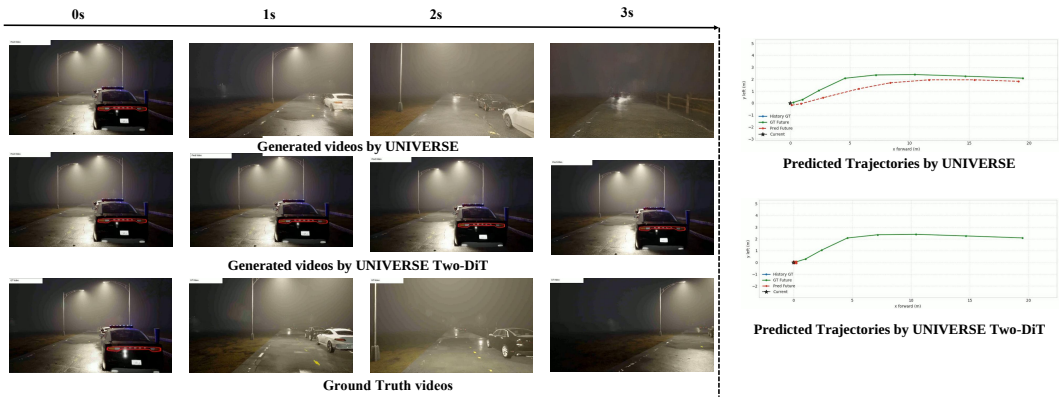


Figure 4: **Zero-shot driving-intention comparison.** UNIVERSE is evaluated on Bench2Drive without fine-tuning. Red and green curves denote predicted and ground-truth trajectories, respectively. When a stationary vehicle blocks the lane ahead, UNIVERSE predicts a bypass trajectory, while the Two-DiT variant tends to stop behind it. Enlarged trajectory plots appear in Fig. 8.

Table 2: **Zero-shot end-to-end motion planning performance on nuScenes [73] and Bench2Drive (CARLA) [75].** The UNIVERSE variants use a NAVSIM-trained checkpoint and are directly tested on the target datasets without fine-tuning. Results for external baselines follow their respective cited protocols. Both target-dataset evaluations are open-loop. † indicates averages recomputed from the displayed 1s, 2s, and 3s collision rates.

Method	Finetune	Ref	nuScenes								Bench2Drive (CARLA)							
			L2 (m) ↓				Collision (%) ↓				L2 (m) ↓				Collision (%) ↓			
			1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
VLA-World Model Methods																		
DriveVLA-W0 [64]	✗	ICLR'26	0.43	1.26	2.60	1.43	0.22	0.66	1.42	0.77	1.01	2.77	5.22	3.00	1.49	2.53	3.53	2.52
World Model Methods																		
PWM [63]	✗	NeurIPS'25	2.06	3.91	6.00	3.99	0.12	0.15	0.86	0.38†	1.70	2.74	3.97	2.80	4.01	3.73	3.53	3.76
UNIVERSE Variants																		
UNIVERSE	✗	–	**0.31**	**0.76**	1.44	**0.83**	**0.00**	**0.03**	**0.05**	**0.03**	**0.67**	**1.29**	**2.03**	**1.33**	**0.75**	**1.12**	**1.70**	**1.19**
UNIVERSE w.o. Mask	✗	–	0.33	**0.76**	**1.43**	0.84	**0.00**	0.07	0.12	0.06	0.69	**1.29**	**2.03**	1.34	1.38	1.97	2.65	2.00†
UNIVERSE w.o. video co-train	✗	–	0.48	0.94	1.65	1.02	0.37	0.31	0.38	0.35	0.75	1.38	2.15	1.43	1.07	1.47	2.16	1.57
UNIVERSE Two-DiT	✗	–	0.44	0.81	1.52	0.92	0.17	0.22	0.25	0.21	0.80	1.39	2.11	1.43	1.30	1.83	2.32	1.82

Comparison with close video-action models. Table 3 complements the controlled variants with an adapted Fast-WAM baseline and DriveVA results. UNIVERSE preserves comparable planning accuracy while supporting trajectory-only inference with the same weights.

4.4 Future-Video Generation Quality

Although motion planning is our primary goal, UNIVERSE also generates high-quality future videos. As shown in Table 4, it achieves the best FID of 6.7 and FVD of 81.7 on nuScenes among the compared driving video-generation methods. This indicates that the video branch learns realistic and temporally coherent scene evolution, while the shared DiT allows such video dynamics modeling to regularize trajectory generation.

4.5 Controlled Comparison with UNIVERSE Variants

The UNIVERSE variants are designed to answer which architectural factors are responsible for the observed action generalization. All variants use the same observations, trajectory targets, tokenization, training schedule, and evaluation protocol, but differ in one key factor: whether future-video supervision is used, whether future video and trajectory tokens are decoupled by the visibility mask, and whether video and trajectory denoising share the same DiT parameters. We compare them jointly across in-domain planning, zero-shot transfer, video-action consistency, inference latency, and memory consumption.

Dense video supervision improves transfer. Removing future-video supervision reduces NAVSIM PDMS from 91.0 to 88.2 in Table 1. The degradation becomes more evident under domain shift: on

Table 3: **Comparison with closely related video-action methods.** Fast-WAM is adapted from official code and trained on NAVSIM.

Method	NAVSIM	nuScenes		Bench2Drive	
	PDMS↑	L2 (m)↓	Col. (%)↓	L2 (m)↓	Col. (%)↓
Fast-WAM [31]	89.8	0.89	0.23	1.45	1.85
DriveVA [30]	90.9	0.84	0.06	**1.33**	1.79
UNIVERSE	**91.0**	**0.83**	**0.03**	**1.33**	**1.19**

Table 4: **Quantitative evaluation of future-video generation on the nuScenes validation set [73].** UNIVERSE achieves competitive visual generation quality while being optimized for planning-oriented video-action co-training.

Metric	DriveGAN [39]	DriveDreamer [41]	DrivingGPT [56]	DrivingWorld [53]	Vista [24]	Epona [62]	**Ours**
FID ↓	73.4	52.6	12.8	7.4	6.9	7.5	**6.7**
FVD ↓	502.3	452.0	142.6	90.9	89.4	82.8	**81.7**

nuScenes, the average L2 error increases from 0.83 m to 1.02 m and the collision rate increases from 0.03% to 0.35%; on Bench2Drive, the collision rate increases from 1.19% to 1.57%. This confirms that future-video prediction is not merely an auxiliary visualization task, but provides dense dynamics supervision that improves transferable action generation.

Shared denoising backbone matters. UNIVERSE Two-DiT preserves both video and trajectory objectives but separates them into two diffusion transformers. Although it still benefits from video co-training at the system level, its NAVSIM PDMS drops to 89.6, and its zero-shot performance is consistently weaker than the shared-DiT UNIVERSE. On nuScenes, the average L2 error increases from 0.83 m to 0.92 m and the collision rate from 0.03% to 0.21%; on Bench2Drive, the collision rate increases from 1.19% to 1.82%. Qualitatively, Fig. 4 shows that the Two-DiT variant tends to stop behind a stationary obstacle, while UNIVERSE predicts a bypass trajectory, indicating stronger driving-intention understanding from shared video-action denoising. It also produces the largest video-action consistency error in Table 5 and the highest training peak memory in Table 6. These results show that video co-training is most effective when video and trajectory losses update the same generative backbone.

Visibility decoupling enables flexible deployment. The no-mask variant obtains comparable in-domain performance with a PDMS of 90.9, but this does not translate into better deployment behavior. Because future video and trajectory tokens can attend to each other, the model relies on coupled video-action rollout and cannot remove one future modality without changing the learned dependency structure. This leads to weaker zero-shot safety: the collision rate increases from 0.03% to 0.06% on nuScenes and from 1.19% to 2.00% on Bench2Drive. It also increases trajectory inference latency from 376 ms to 1623 ms, since future-video denoising must be retained during rollout. The visibility mask therefore preserves training–inference dependency consistency while enabling flexible trajectory-only deployment. Small submetric trade-offs are discussed in Sec. C.3.

Video-action consistency and efficiency. The qualitative results in Fig. 3 show that generated future videos and predicted trajectories evolve consistently. To quantify this alignment, we use DPVO [81] to reconstruct ego-motion from ground-truth and generated future videos, align the reconstructed trajectories to their references with a similarity transform, and report the average L2 error over the future 4 s horizon. As shown in Table 5, UNIVERSE achieves an average error of 0.13, close to the GT-reference error of 0.08, while the no-mask and Two-DiT variants increase the error to 0.15 and 0.19. Meanwhile, Table 6 shows that UNIVERSE achieves the lowest latency, inference memory, and training memory among the variants. Together, these results support the coupling-by-parameters and decoupling-by-visibility design: video and trajectory objectives should share the same DiT parameters, while their future tokens should remain mutually invisible.

Additional comparisons of video backbones and NAVSIM Train–Navtest gaps are provided in Sec. C, and horizon and sampling ablations are in Sec. C.4.

Table 5: **Video-action consistency.** DPVO [81] 4s ego-motion L2 (m); protocol in Sec. E.

Method	NAVSIM↓	nuScenes↓	Avg.↓
GT reference	0.09	0.07	0.08
UNIVERSE	**0.12**	**0.14**	**0.13**
UNIVERSE w.o. Mask	0.16	**0.14**	0.15
UNIVERSE Two-DiT	0.17	0.21	0.19

Table 6: **Computational efficiency.** Trajectory inference latency, inference memory, and training memory.

Method	Latency (ms)↓	Infer. Peak Mem. (GB)↓	Train Peak Mem. (GB)↓
UNIVERSE	**376**	**23.3**	**52.1**
UNIVERSE w.o. Mask	1623	26.3	54.4
UNIVERSE Two-DiT	551	32.8	88.3

4.6 Pretraining and Sensitivity Analyses

Pretrained initialization. Table 7 compares the same 5B architecture trained from scratch with one initialized from pretrained video-generation weights. The gain from 63.5 to 91.0 NAVSIM PDMS shows that pretrained initialization substantially contributes to performance, alongside the architectural effects studied above.

Table 7: **Effect of pretrained initialization.** Both rows use the 5B UNIVERSE architecture. Target-dataset metrics are average open-loop L2 (m) and collision rate (%).

Initialization	NAVSIM	nuScenes		Bench2Drive	
	PDMS↑	L2↓	Col.↓	L2↓	Col.↓
From scratch	63.5	1.32	1.23	2.48	4.85
Pretrained weights	**91.0**	**0.83**	**0.03**	**1.33**	**1.19**

Video-loss weight sensitivity. Table 8 studies the effect of the video-loss weight λ_{vid} while fixing the Wan2.2-TI2V-5B backbone. The default setting $\lambda_{\text{vid}} = 1.0$ gives the best overall balance across source and cross-domain evaluations.

Table 8: **Video-loss weight sensitivity.** Wan2.2-TI2V-5B is fixed; $\lambda_{\text{vid}} = 1.0$ is the default. L2 is in meters and collision is in percent.

λ_{vid}	NAVSIM	nuScenes		Bench2Drive	
	PDMS↑	L2↓	Col.↓	L2↓	Col.↓
0.5	90.8	0.83	0.05	1.34	**1.18**
1.0 (default)	**91.0**	**0.83**	**0.03**	**1.33**	1.19
2.0	90.9	**0.83**	0.05	1.36	1.21

5 Conclusion

We present UNIVERSE, a video-action model that retains dense video supervision of shared DiT parameters while making future-video generation optional at deployment. The Modality-Decoupling Visibility Mask separates future-token dependencies and enables trajectory-only, video-only, and joint inference with one set of weights. Experiments show strong NAVSIM performance and zero-shot transfer to nuScenes and Bench2Drive, with $4.3\times$ faster trajectory inference than coupled rollout and lower memory use than the compared variants. The results support shared-parameter training with dependency-consistent modality decoupling.

Acknowledgments and Disclosure of Funding

This work has been supported by the Centre for Spatial Intelligence (RCSI) at University of Bath, the European Union under grant agreement no. 101136006-XTREME, and the European Innovation Council under grant agreement no. 101257536-CEREBRIS.

Competing Interests. The authors declare no competing interests.

References

[1] Xiaole Tang, Xiaoyi He, Jiayi Xu, Xiang Gu, and Jian Sun. Learning continuous wasserstein barycenter space for generalized all-in-one image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2026.

[2] Mengmeng Liu, Jiuming Liu, Yunpeng Zhang, Jiangtao Li, Michael Ying Yang, Francesco Nex, and Hao Cheng. 4dstr: Advancing generative 4d gaussians with spatial-temporal rectification for high-quality and consistent 4d generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 7224–7232, 2026.

[3] Shuo Lu, Yinuo Xu, Kecheng Yu, Siru Jiang, Yongcan Yu, Yubin Wang, Haitao Yang, Yuxiang Zhang, Bin Wang, Ran He, et al. Worldcoder-bench: Benchmarking physically grounded 3d world synthesis. *arXiv preprint arXiv:2606.01869*, 2026.

[4] Jiuming Liu, Mengmeng Liu, Siting Zhu, Yunpeng Zhang, Jiangtao Li, Michael Yang, Francesco Nex, Hao Cheng, and Hesheng Wang. Arflow: Auto-regressive optical flow estimation for arbitrary-length videos via progressive next-frame forecasting. In *International Conference on Learning Representations*, volume 2026, pages 130125–130154, 2026.

[5] Shuo Lu, Yanyin Chen, Wei Feng, Jiahao Fan, Fengheng Li, Zheng Zhang, Jingjing Lv, Junjie Shen, Ching Law, and Jian Liang. Uni-layout: Integrating human feedback in unified layout generation and evaluation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 7709–7718, 2025.

[6] Mengmeng Liu, Jiuming Liu, Michael Ying Yang, Chaokang Jiang, Jiangtao Li, Yunpeng Zhang, Hesheng Wang, Francesco Nex, and Hao Cheng. Streamvlo: Streaming visual-lidar odometry with cumulative drift compensation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 39086–39097, 2026.

[7] Mengmeng Liu, Michael Ying Yang, Jiuming Liu, Yunpeng Zhang, Jiangtao Li, Sander Oude Elberink, George Vosselman, and Hao Cheng. Dvlo4d: Deep visual-lidar odometry with sparse spatial-temporal fusion. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9740–9747. IEEE, 2025.

[8] Mengmeng Liu, Hao Cheng, and Michael Ying Yang. Tracing the influence of predecessors on trajectory prediction. In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 3245–3255. IEEE, 2023.

[9] Hengrui Zhang, Yuhu Cheng, CL Chen, and Xuesong Wang. Diffusion subgoal planning for long-horizon offline goal-conditioned reinforcement learning. *arXiv preprint arXiv:2609.34575*, 2026.

[10] Tianchen Deng, Xuefeng Chen, Yi Chen, Qu Chen, Yuyao Xu, Lijin Yang, Le Xu, Yu Zhang, Bo Zhang, Wuxiong Huang, and Hesheng Wang. Gaussiandwm: 3d gaussian driving world model for unified scene understanding and multi-modal generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10656–10667, June 2026.

[11] Tianchen Deng, Xuefeng Chen, Shuang Wu, Qu Chen, Jiajun Zhu, Bo Dai, Jianfei Yang, and Hesheng Wang. Gaussiandwm++: Language-grounded 3d gaussian driving world model for unified scene understanding, editing, and multi-modal generation. *arXiv preprint arXiv:2608.16234*, 2026.

[12] Mengmeng Liu, Hao Cheng, Lin Chen, Hellward Broszio, Jiangtao Li, Runjiang Zhao, Monika Sester, and Michael Ying Yang. Laformer: Trajectory prediction for autonomous driving with lane-aware scene constraints. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2039–2049. IEEE, 2024.

[13] Hao Cheng, Mengmeng Liu, Lin Chen, Hellward Broszio, Monika Sester, and Michael Ying Yang. Gatraj: A graph-and attention-based multi-agent trajectory prediction model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 205:163–175, 2023.

[14] Jiuming Liu, Guangming Wang, Zhe Liu, Chaokang Jiang, Marc Pollefeys, and Hesheng Wang. Regformer: An efficient projection-aware transformer network for large-scale point cloud registration. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8417–8426. IEEE, 2023.

[15] Jiuming Liu, Weicai Ye, Guangming Wang, Chaokang Jiang, Lei Pan, Jinru Han, Zhe Liu, Guofeng Zhang, and Hesheng Wang. Difflow3d: Hierarchical diffusion models for uncertainty-aware 3d scene flow estimation. *IEEE transactions on pattern analysis and machine intelligence*, 2025.

[16] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023.

[17] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022.

[18] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 12037–12047, 2025.

[19] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[20] Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. *Open Review*, 62(1):1–62, 2022.

[21] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[22] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14749–14759, 2024.

[23] Wenzhao Zheng, Ruiqi Song, Xianda Guo, Chenming Zhang, and Long Chen. Genad: Generative end-to-end autonomous driving. In *European Conference on Computer Vision*, pages 87–104. Springer, 2024.

[24] 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.

[25] Joel Jang, Seonghyeon Ye, Zongyu Lin, Jiannan Xiang, Johan Bjorck, Yu Fang, Fengyuan Hu, Spencer Huang, Kaushil Kundalia, Yen-Chen Lin, Loïc Magne, Ajay Mandlekar, Avnish Narayan, You Liang Tan, Guanzhi Wang, Jing Wang, Qi Wang, Yinzhen Xu, Xiaohui Zeng, Kaiyuan Zheng, Ruijie Zheng, Ming-Yu Liu, Luke Zettlemoyer, Dieter Fox, Jan Kautz, Scott Reed, Yuke Zhu, and Linxi Fan. Dreamgen: Unlocking generalization in robot learning through video world models. In *9th Annual Conference on Robot Learning*, 2025.

[26] Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Luyao Zhang, Mingrui Yu, Zelin Gao, Nan Xue, Boyu Zhou, Xing Zhu, Mingyu Ding, Yujun Shen, and Yinghao Xu. Causal world modeling for robot control. In *Robotics: Science and Systems (RSS)*, 2026.

[27] Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025.

[28] Chuning Zhu, Raymond Yu, Siyuan Feng, Benjamin Burchfiel, Paarth Shah, and Abhishek Gupta. Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets. In *Robotics: Science and Systems (RSS)*, 2025.

[29] Shuo Lu, Haohan Wang, Wei Feng, Weizhen Wang, Shen Zhang, Yaoyu Li, Ao Ma, Zheng Zhang, Jingjing Lv, Junjie Shen, et al. One size, many fits: Aligning diverse group-wise click preferences in large-scale advertising image generation. *arXiv preprint arXiv:2602.02033*, 2026.

[30] Mengmeng Liu, Diankun Zhang, Jiuming Liu, Jianfeng Cui, Hongwei Xie, Guang Chen, Hangjun Ye, Michael Ying Yang, Francesco Nex, and Hao Cheng. DriveVA: Video action models are zero-shot drivers. In *Computer Vision – ECCV 2026*, volume 17066 of *Lecture Notes in Computer Science*, pages 315–335. Springer, 2026.

[31] Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

[32] Jingyu Li, Bozhou Zhang, Xin Jin, Jiankang Deng, Xiatian Zhu, and Li Zhang. ImagiDrive: A unified imagination-and-planning framework for autonomous driving. *arXiv preprint arXiv:2508.11428*, 2025.

[33] Angen Ye, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Hao Li, Hengtao Li, Jie Li, Jindi Lv, Jingyu Liu, Min Cao, Peng Li, Qiuping Deng, Wenjun Mei, Xiaofeng Wang, Xinze Chen, Xinyu Zhou, Yang Wang, Yifan Chang, Yifan Li, Yukun Zhou, Yun Ye, Zhichao Liu, and Zheng Zhu. Gigaworld-policy: An efficient action-centered world–action model. *arXiv preprint arXiv:2603.17240*, 2026.

[34] Xiaodong Wang and Peixi Peng. Prophetdwm: A driving world model for rolling out future actions and videos. *arXiv preprint arXiv:2505.18650*, 2025.

[35] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[36] Haoyu Chen, Kaichen Zhou, Hang Hua, Kaile Zhang, Jingwen Qian, Wufei Ma, Haonan Chen, Chunjiang Liu, Yizhou Zhao, Xiaoyuan Wang, et al. Memobench: Benchmarking world modeling in dynamically changing environments. In *European Conference on Computer Vision*, pages 658–676. Springer, 2026.

[37] Jack Wei Lun Shi, Kaichen Zhou, Haoyu Chen, Yufeng Weng, Keane Ong, Ruojin Cai, Hang Hua, Justin KW Yeoh, and Mengyu Wang. Honeycomb: Constant-size scene memory representation for video world models. *arXiv preprint arXiv:2609.37690*, 2026.

[38] Jiuming Liu, Chaojun Ni, Mengmeng Liu, Chensheng Peng, Fangjinhua Wang, Sitian Shen, Marc Pollefeys, Masayoshi Tomizuka, Ayush Tewari, and Per Ola Kristensson. Towards interactive video world modeling: Frontiers, challenges, benchmarks, and future trends. *arXiv preprint arXiv:2606.01164*, 2026.

[39] Seung Wook Kim, Jonah Philion, Antonio Torralba, and Sanja Fidler. Drivegan: Towards a controllable high-quality neural simulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5820–5829, 2021.

[40] 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 (ICLR)*, 2024.

[41] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drivedreamer: Towards real-world-drive world models for autonomous driving. In *European conference on computer vision*, pages 55–72. Springer, 2024.

[42] Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Xinze Chen, Guan Huang, Xiaoyi Bao, and Xingang Wang. Drivedreamer-2: Llm-enhanced world models for diverse driving video generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 10412–10420, 2025.

[43] 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.

[44] 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.

[45] Lloyd Russell, Anthony Hu, Lorenzo Bertoni, George Fedoseev, Jamie Shotton, Elahe Arani, and Gianluca Corrado. Gaia-2: A controllable multi-view generative world model for autonomous driving. *arXiv preprint arXiv:2503.20523*, 2025.

[46] 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.

[47] Haiguang Wang, Daqi Liu, Hongwei Xie, Haisong Liu, Enhui Ma, Kaicheng Yu, Limin Wang, and Bing Wang. Mila: Multi-view intensive-fidelity long-term video generation world model for autonomous driving. *arXiv preprint arXiv:2503.15875*, 2025.

[48] Yishen Ji, Ziyue Zhu, Zhenxin Zhu, Kaixin Xiong, Ming Lu, Zhiqi Li, Lijun Zhou, Haiyang Sun, Bing Wang, and Tong Lu. Cogen: 3d consistent video generation via adaptive conditioning for autonomous driving. *arXiv preprint arXiv:2503.22231*, 2025.

[49] Xiangyu Guo, Zhanqian Wu, Kaixin Xiong, Ziyang Xu, Lijun Zhou, Gangwei Xu, Shaoqing Xu, Haiyang Sun, Bing Wang, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. Genesis: Multimodal driving scene generation with spatio-temporal and cross-modal consistency. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

[50] Bohan Li, Zhuang Ma, Dalong Du, Baorui Peng, Zhujin Liang, Zhenqiang Liu, Xianda Guo, Zheng Zhu, Chao Ma, Yueming Jin, Xin Jin, Hao Zhao, and Wenjun Zeng. OmniNWM: Omniscient driving navigation world models. In *European Conference on Computer Vision*, volume 17028 of *Lecture Notes in Computer Science*, pages 187–208. Springer, 2026.

[51] 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.

[52] Chen Min, Dawei Zhao, Liang Xiao, Jian Zhao, Xinli Xu, Zheng Zhu, Lei Jin, Jianshu Li, Yulan Guo, Junliang Xing, et al. Driveworld: 4d pre-trained scene understanding via world models for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15522–15533, 2024.

[53] Xiaotao Hu, Mingkai Jia, Xiaoyang Guo, Qian Zhang, Xiao-xiao Long, and Wei Yin. DrivingWorld: Constructing world model for autonomous driving via video GPT. In *Pattern Recognition (ICPR 2026), Part II*, volume 16813 of *Lecture Notes in Computer Science*, pages 276–291, Cham, 2027. Springer.

[54] 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.

[55] 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.

[56] Yuntao Chen, Yuqi Wang, and Zhaoxiang Zhang. Drivinggpt: Unifying driving world modeling and planning with multi-modal autoregressive transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26890–26900, 2025.

[57] Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. In *International Conference on Learning Representations (ICLR)*, 2025.

[58] Wenzhao Zheng, Zetian Xia, Yuanhui Huang, Sicheng Zuo, Jie Zhou, and Jiwen Lu. Doe-1: Closed-loop autonomous driving with large world model. *arXiv preprint arXiv:2412.09627*, 2024.

[59] Jiazhi Yang, Kashyap Chitta, Shenyuan Gao, Long Chen, Yuqian Shao, Xiaosong Jia, Hongyang Li, Andreas Geiger, Xiangyu Yue, and Li Chen. ReSim: Reliable world simulation for autonomous driving. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

[60] Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 27137–27146, 2025.

[61] Tianze Xia, Yongkang Li, Lijun Zhou, Jingfeng Yao, Kaixin Xiong, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. Drivelaw: Unifying planning and video generation in a latent driving world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 39701–39712, June 2026.

[62] Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, Xun Cao, and Wei Yin. Epona: Autoregressive diffusion world model for autonomous driving. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 27220–27230, 2025.

[63] Zhida Zhao, Talas Fu, Yifan Wang, Lijun Wang, and Huchuan Lu. From forecasting to planning: Policy world model for collaborative state-action prediction. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

[64] Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, Lu Hou, Lue Fan, and Zhaoxiang Zhang. DriveVLA-W0: World models amplify data scaling law in autonomous driving. In *International Conference on Learning Representations (ICLR)*, 2026.

[65] Jonas Pai, Liam Achenbach, Oliver Sanchez, Stefanos Charalambous, Victoriano Montesinos, Benedek Forrai, Oier Mees, and Elvis Nava. mimic-video: Video-action models for generalizable robot control beyond vlas. In *Robotics: Science and Systems*, 2026.

[66] Junbang Liang, Pavel Tokmakov, Ruoshi Liu, Sruthi Sudhakar, Paarth Shah, Rares Ambrus, and Carl Vondrick. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*, 2025.

[67] Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, and Jinwei Gu. Cosmos Policy: Fine-tuning video models for visuomotor control and planning. In *International Conference on Learning Representations (ICLR)*, 2026.

[68] Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. VideoVLA: Video generators can be generalizable robot manipulators. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

[69] John Won, Kyungmin Lee, Huiwon Jang, Dongyoung Kim, and Jinwoo Shin. Dual-stream diffusion for world-model augmented vision-language-action model. In *Proceedings of the 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, pages 135349–135370. PMLR, 2026.

[70] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[71] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems*, 37:28706–28719, 2024.

[72] OpenScene Contributors. Openscene: The largest up-to-date 3d occupancy prediction benchmark in autonomous driving. `https://github.com/OpenDriveLab/OpenScene`, 2023.

[73] 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.

[74] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023.

[75] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. *Advances in Neural Information Processing Systems*, 37:819–844, 2024.

[76] 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.

[77] Xuemin Hu, Shen Li, Tingyu Huang, Bo Tang, Rouxing Huai, and Long Chen. How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence. *IEEE Transactions on Intelligent Vehicles*, 9(1):593–612, 2023.

[78] Bo Jiang, Shaoyu Chen, Hao Gao, Bencheng Liao, Qian Zhang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust, editors, *International Conference on Learning Representations*, volume 2026, pages 68886–68900, 2026.

[79] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15449–15458, 2024.

[80] Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. ReCog-Drive: A reinforced cognitive framework for end-to-end autonomous driving. In *International Conference on Learning Representations (ICLR)*, 2026.

[81] Zachary Teed, Lahav Lipson, and Jia Deng. Deep patch visual odometry. *Advances in Neural Information Processing Systems*, 36:39033–39051, 2023.

[82] NVIDIA. Cosmos-Predict2.5-2B: Model card. `https://huggingface.co/nvidia/Cosmos-Predict2.5-2B`, 2025.

[83] NVIDIA. Cosmos3-Edge: Model card. `https://huggingface.co/nvidia/Cosmos3-Edge`, 2026.

[84] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022.

[85] Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8406–8415, 2023.

A Supplementary Material

This appendix provides architecture and inference details (Sec. B), additional ablations and generalization analysis (Sec. C), and the zero-shot evaluation protocol (Sec. D). It also retains DPVO verification (Sec. E), the extended nuScenes comparison (Sec. F), qualitative results (Sec. G), limitations, asset licenses, and broader impacts.

B Additional Architecture Details

B.1 Visibility Mask Implementation

The main paper introduces the Modality-Decoupling Visibility Mask conceptually. Here we provide its explicit block form. Let the self-attention token groups be historical context tokens C, future video tokens V, and future trajectory tokens A. The binary visibility mask is

$$\mathbf{M} = \begin{array}{c|ccc} & C & V & A \\ \hline C & 1 & 0 & 0 \\ V & 1 & 1 & 0 \\ A & 1 & 0 & 1 \end{array}, \tag{7}$$

where rows denote query tokens, columns denote key/value tokens, 1 means attention is allowed, and 0 means attention is blocked.

This mask enforces three constraints. First, both future modalities read the same historical context through $V \to C$ and $A \to C$ attention. Second, context tokens remain read-only with respect to future targets, since $C \to V$ and $C \to A$ are blocked. This prevents context tokens from acting as an indirect information bridge between future video and trajectory tokens. Third, future video and trajectory tokens are mutually invisible through blocked $V \to A$ and $A \to V$ attention. Thus, video and trajectory denoising share model parameters but remain mutually invisible at the future-token level.

Full intra-modality attention is preserved within V and within A. This is valid because UNIVERSE predicts future video and trajectory chunks with diffusion-style denoising rather than autoregressive next-step decoding. Bidirectional attention within V helps maintain temporal coherence across generated frames, while bidirectional attention within A helps produce smooth trajectory chunks.

B.2 Inference Token Configurations

At inference time, UNIVERSE uses the same trained DiT weights but instantiates different future token groups. Table 9 summarizes the token configurations for each inference mode.

Table 9: **Inference token configurations.** Different inference modes are obtained by changing the instantiated future token groups while keeping the same shared DiT weights.

Mode	Context C	Future Video V	Future Trajectory A
Trajectory-only rollout	✓	–	✓
Video-only rollout	✓	✓	–
Joint video-action rollout	✓	✓	✓

The key property is dependency consistency. During training, trajectory tokens are blocked from attending to future video tokens. Therefore, removing V during trajectory-only inference does not change the attention dependency of the trajectory branch. This differs from naively coupled video-action generation, where removing future video tokens at test time would alter the learned dependency structure and introduce a train–test mismatch.

C Additional Experimental Analyses

C.1 Comparison of Pretrained Video Backbones

Table 10 compares Cosmos-Predict2.5-2B [82], Cosmos3-Edge (4B) [83], and the default Wan2.2-TI2V-5B [70], with $\lambda_{\text{vid}} = 1.0$. The comparison varies the pretrained backbone while retaining the video-action formulation and loss weight. All three configurations yield strong NAVSIM and zero-shot results, but the comparison changes model family and pretraining as well as size; it is not a pure parameter-scaling experiment. The sizes in the table identify the nominal backbone configurations rather than an audited count of every auxiliary component.

Table 10: **Pretrained backbone comparison.** The video-loss weight is fixed to 1.0. † marks the default. L2 is in meters and collision is in percent.

Backbone	Size	NAVSIM	nuScenes		Bench2Drive	
		PDMS↑	L2↓	Col.↓	L2↓	Col.↓
Cosmos-Predict2.5	2B	90.7	0.85	**0.03**	1.37	**1.18**
Cosmos3-Edge	4B	90.8	0.86	0.04	1.35	1.21
Wan2.2-TI2V†	5B	**91.0**	**0.83**	**0.03**	**1.33**	1.19

C.2 Source-Domain Fitting and Generalization

By *dataset-specific action priors*, we mean frequent source-dataset motion patterns that may help predict ego actions without fully using scene-dependent dynamics. The same high-level command can require continuing, braking, bypassing, or yielding depending on surrounding agents and obstacles. We examine source-domain fitting and NAVSIM Navtest performance using the same evaluation protocol (Table 11). Two-DiT and UNIVERSE have nearly identical training PDMS, but UNIVERSE has a smaller Train–Navtest gap and stronger zero-shot performance (Table 2). Thus, the weaker NAVSIM Navtest performance of Two-DiT is not accompanied by weaker training-set fit in this comparison.

Table 11: **Source-domain fitting and NAVSIM Train–Navtest gap.** Gap is NAVSIM Train PDMS minus NAVSIM Navtest PDMS, in score points.

Method	Train PDMS↑	Navtest PDMS↑	Train–Navtest Gap↓
Two-DiT	93.78	89.6	4.18
UNIVERSE	93.80	**91.0**	**2.80**

The average DPVO-based consistency error also decreases from 0.19 for Two-DiT to 0.13 for UNIVERSE (Table 5). Together, these observations are consistent with a more transferable representation

under shared video-action training. They are supportive rather than conclusive evidence about action-prior reliance: neither a Train–Navtest gap nor zero-shot collision rate uniquely measures this mechanism, and we do not claim a causal decomposition of the improvement.

C.3 Submetric Trade-offs and Inference Modes

The objective optimizes trajectory and video flow matching, not individual NAVSIM submetrics. Accordingly, small trajectory changes can trade off NC, DAC, TTC, comfort, and progress; the full model need not dominate every submetric. The no-mask variant has slightly higher NC and TTC, whereas UNIVERSE has higher aggregate PDMS, lower reported cross-domain collision rates, and substantially lower trajectory-inference latency. The visibility mask adds no learned parameters and resolves the tension between shared training parameters and independently selectable outputs.

Trajectory-only inference is the mode used for the principal planning and efficiency results. Video-only and joint inference demonstrate flexible output capabilities, and joint outputs support the reported consistency analyses. Synthetic-data generation, decision diagnosis, and self-simulation are potential uses, not separately validated downstream improvements in this work.

C.4 Additional Ablation on Horizon and Sampling

Table 12: **Ablation study on future video frames and sampling steps.** We use 8 future frames and 2 sampling steps as the default setting.

Factor	Value	NC↑	DAC↑	TTC↑	Comf.↑	EP↑	PDMS↑
Future frames	4	98.4	92.3	97.8	99.5	77.6	84.9
	8	**99.1**	**97.6**	**98.5**	**100**	**83.6**	**91.0**
	12	98.8	94.8	98.3	**100**	80.8	88.1
Sampling steps	1	61.2	37.9	50.4	1.8	14.1	13.7
	2	**99.1**	**97.6**	98.5	**100**	**83.6**	**91.0**
	3	**99.1**	**97.6**	**98.6**	**100**	83.5	**91.0**

Table 12 studies the number of future video frames and flow sampling steps. Using 4 future frames provides insufficient long-horizon supervision, while 8 frames achieves the best PDMS of 91.0. Further increasing the horizon to 12 frames brings no additional gain. For sampling, one step is unstable, whereas two steps already recover full performance and three steps offer negligible improvement. We therefore use 8 future frames and 2 sampling steps by default.

D Zero-Shot Evaluation Protocol

D.1 Data Splits, Inputs, and Preprocessing

All cross-dataset results use the same checkpoint trained exclusively on NAVSIM v1, without target-domain fine-tuning or adaptation. Table 13 summarizes the settings used for the open-loop evaluations. The eight predicted trajectory points cover 4 s, while the main comparison reports errors at 1, 2, and 3 s and the reported averages.

Preprocessing. Images are converted to RGB, resized to 832×480, and normalized to $[-1, 1]$, without test-time cropping, augmentation, undistortion, or multi-view stitching. Trajectories are converted to a unified ego-centric coordinate system centered at the current ego pose; nuScenes global poses and Bench2Drive native poses are transformed accordingly.

Conditioning. The policy receives front-camera history, current ego velocity, and a left/straight/right high-level command. Its inputs do not include camera intrinsics/extrinsics, future coordinates, maps, bounding boxes, depth, occupancy, or LiDAR. This concerns the policy's conditioning inputs, not the benchmark annotations used to compute evaluation metrics.

No target-domain adaptation. There is no target-domain fine-tuning, LoRA, test-time optimization, parameter calibration, checkpoint selection, or normalization fitting.

Table 13: **Open-loop zero-shot evaluation settings.** Both datasets use front-camera history only, together with current ego velocity and a high-level command.

Setting	nuScenes	Bench2Drive
Evaluation split	Official v1.0-trainval/val split (150 scenes)	Official validation set
Camera	CAM_FRONT only	CAM_FRONT only
Temporal sampling	2 Hz	Downsampled from 10 Hz to 2 Hz
History	Four frames	Four frames
Prediction	Eight trajectory points over 4 s	Eight trajectory points over 4 s
Resolution	832×480	832×480
Checkpoint	The same NAVSIM-trained checkpoint; no target adaptation	

Model availability. The released UNIVERSE checkpoint is available at `https://huggingface.co/mengmengliu1998/UNIVERSE`. The preceding protocol specifies the evaluation splits, conditioning inputs, preprocessing, and absence of target-domain adaptation; public benchmark assets and external dependencies remain subject to their providers' respective terms.

E DPVO-based Verification of Video-Action Consistency

We use DPVO [81] as an external visual-odometry system to verify whether generated videos and predicted trajectories describe consistent ego motion. For each sample, DPVO is run on both the ground-truth future video and the future video generated by UNIVERSE. This produces two reconstructed trajectories, denoted as DPVO(gt img) and DPVO(pred img). We compare DPVO(gt img) with the ground-truth ego trajectory, and DPVO(pred img) with the trajectory predicted by UNIVERSE.

Since monocular visual odometry is ambiguous up to scale, rotation, and translation, we align each reconstructed trajectory to its reference using a 2D similarity transform. Let $\tau = \{\mathbf{p}_t\}_{t=1}^{T}$ be the reference trajectory and $\hat{\tau} = \{\hat{\mathbf{p}}_t\}_{t=1}^{T}$ be the DPVO-reconstructed trajectory on the ground plane. We solve

$$(s^\star, R^\star, \mathbf{t}^\star) = \arg\min_{s,R,\mathbf{t}} \sum_{t=1}^{T} \|sR\hat{\mathbf{p}}_t + \mathbf{t} - \mathbf{p}_t\|_2^2 , \tag{8}$$

and compute the aligned trajectory as $\tilde{\mathbf{p}}_t = s^\star R^\star \hat{\mathbf{p}}_t + \mathbf{t}^\star$. The final consistency error is the average L2 distance over the future 4s horizon:

$$\text{Avg. L2}(\tilde{\tau}, \tau) = \frac{1}{T} \sum_{t=1}^{T} \|\tilde{\mathbf{p}}_t - \mathbf{p}_t\|_2 . \tag{9}$$

As reported in Table 5, UNIVERSE achieves an average DPVO-based error of 0.13, close to the GT-reference error of 0.08. This indicates that the ego motion implied by the generated future videos is consistent with the trajectory predicted by the model. Fig. 5 further shows that DPVO(pred img) closely follows the predicted trajectory across diverse zero-shot driving scenarios, supporting the video-action consistency of the unified generative process.

F Additional Comparison on nuScenes

Table 14 provides an extended comparison on the nuScenes planning benchmark. Different from Table 2 in the main paper, which focuses on zero-shot transfer from NAVSIM to nuScenes and Bench2Drive, this table additionally includes methods trained or fine-tuned on nuScenes. Although UNIVERSE is directly evaluated on nuScenes without any fine-tuning, it achieves lower L2 error and collision rate than the listed nuScenes-trained or fine-tuned baselines. This further suggests that unified video-action co-training learns transferable planning priors instead of only fitting the source-domain trajectory distribution.

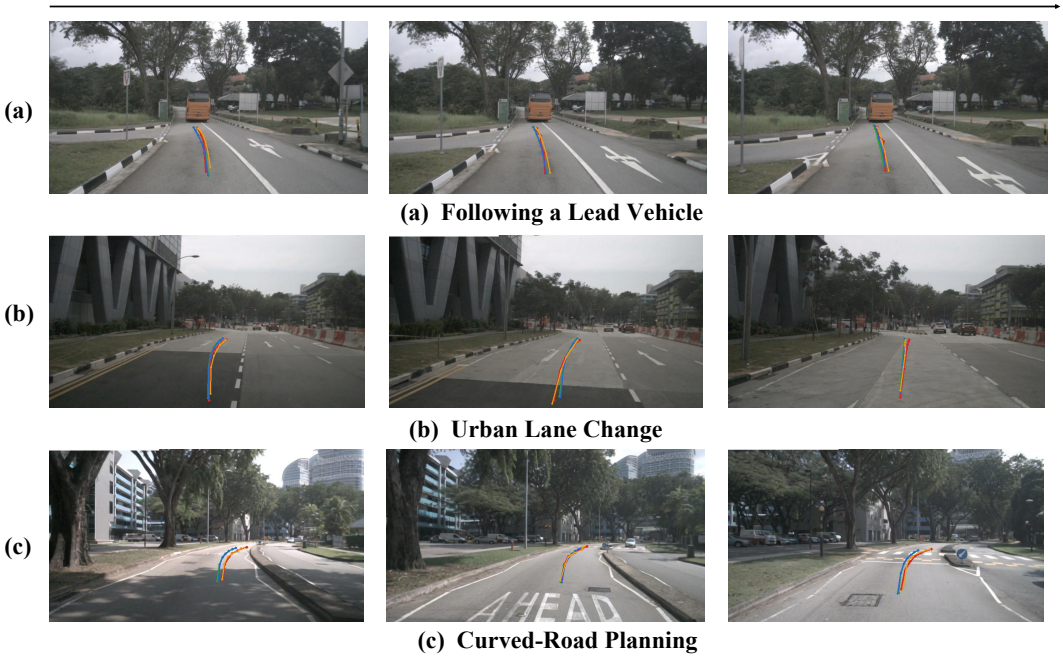


Figure 5: **Zero-shot video-action consistency across diverse driving scenarios.** UNIVERSE is directly evaluated on unseen nuScenes and Bench2Drive scenes without fine-tuning, covering lead-vehicle following, urban lane change, and curved-road planning. For nuScenes, DPVO [81] is used to reconstruct ego trajectories from both ground-truth and generated future videos. GT Future and Pred Future denote ground-truth and predicted trajectories, while DPVO(gt img) and DPVO(pred img) denote DPVO reconstructions from ground-truth and predicted videos. The close alignment among these trajectories shows that the generated visual dynamics remain consistent with the predicted ego motion under domain shift.

Table 14: **End-to-end motion planning performance on the nuScenes [73] dataset.** * indicates that only the front camera is used as input. Although UNIVERSE is not fine-tuned on nuScenes, it outperforms prior methods trained or fine-tuned on nuScenes.

Method	nuScenes Finetune	Ref	Input	Auxiliary Supervision	L2 (m) ↓				Collision Rate (%) ↓			
					1s	2s	3s	Avg.	1s	2s	3s	Avg.
ST-P3 [84]	✓	ECCV'22	Camera	Map&Box&Depth	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71
UniAD [16]	✓	CVPR'23	Camera	Map&Box&Motion	0.48	0.96	1.65	1.03	0.05	0.17	0.71	0.31
OccNet [85]	✓	ICCV'23	Camera	3D-Occ&Map&Box	1.29	2.13	2.99	2.14	0.21	0.59	1.37	0.72
OccWorld [51]	✓	ECCV'24	Camera	3D-Occ	0.52	1.27	2.41	1.40	0.12	0.40	2.08	0.87
VAD-Tiny [74]	✓	ICCV'23	Camera	Map&Box&Motion	0.60	1.23	2.06	1.30	0.31	0.53	1.33	0.72
VAD-Base [74]	✓	ICCV'23	Camera	Map&Box&Motion	0.54	1.15	1.98	1.22	0.04	0.39	1.17	0.53
GenAD [23]	✓	ECCV'24	Camera	Map&Box&Motion	0.36	0.83	1.55	0.91	0.06	0.23	1.00	0.43
Doe-1 [58]	✓	arXiv'24	Camera*	QA	0.50	1.18	2.11	1.26	0.04	0.37	1.19	0.53
Epona [62]	✓	ICCV'25	Camera*	None	0.61	1.17	1.98	1.25	0.01	0.22	0.85	0.36
UNIVERSE Variants												
UNIVERSE	✗	–	Camera*	None	**0.31**	**0.76**	1.44	**0.83**	**0.00**	**0.03**	**0.05**	**0.03**
UNIVERSE w.o. Mask	✗	–	Camera*	None	0.33	**0.76**	**1.43**	0.84	**0.00**	0.07	0.12	0.06
UNIVERSE w.o. video co-train	✗	–	Camera*	None	0.48	0.94	1.65	1.02	0.37	0.31	0.38	0.35
UNIVERSE Two-DiT	✗	–	Camera*	None	0.44	0.81	1.52	0.92	0.17	0.22	0.25	0.21

G Additional Zero-Shot Visualization

Fig. 6 provides additional qualitative results on unseen datasets. The model is trained on NAVSIM and directly evaluated on nuScenes and Bench2Drive without fine-tuning. For nuScenes, DPVO [81] reconstructs ego trajectories from both ground-truth and generated future videos. The close alignment between predicted trajectories, generated video evolution, and DPVO-reconstructed motion indicates that the video and trajectory branches learn consistent future dynamics even under domain shift. For Bench2Drive, the generated future frames and trajectories also remain temporally coherent despite the real-to-simulation visual and dynamics gap.



Figure 6: **Zero-shot qualitative results on nuScenes and Bench2Drive.** UNIVERSE is trained on NAVSIM and directly evaluated on unseen target datasets without fine-tuning. The generated future videos remain consistent with the predicted ego trajectories across both real-world and simulation-domain scenes. For nuScenes, DPVO [81] reconstructions from ground-truth and generated videos further show that the visual rollouts imply ego motion aligned with the reference and predicted trajectories.

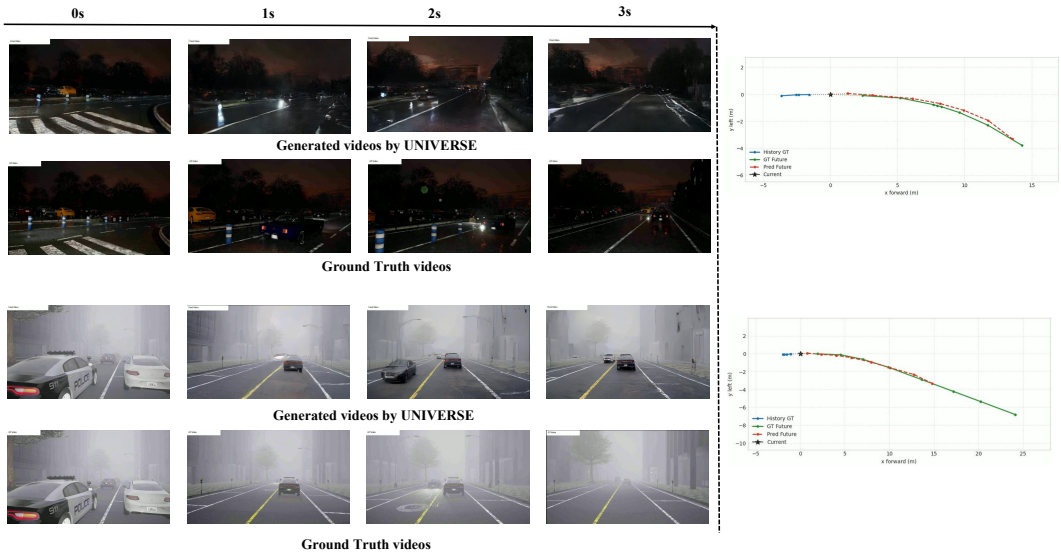
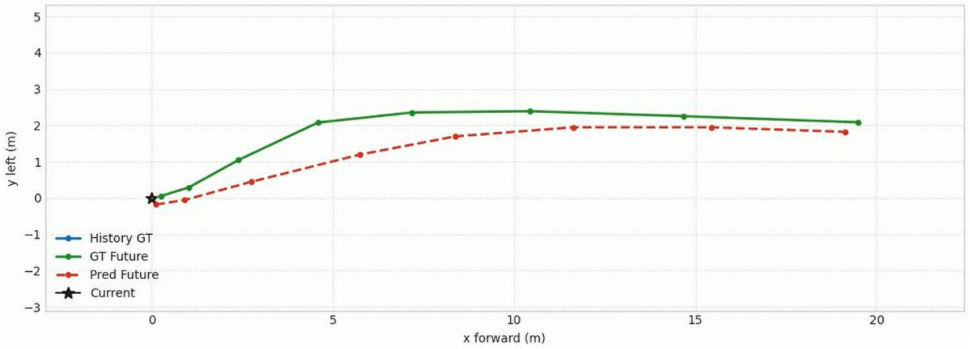


Figure 7: **Zero-shot video generation and trajectory consistency on Bench2Drive.** UNIVERSE is directly evaluated on unseen Bench2Drive scenes without fine-tuning, including challenging nighttime and foggy driving conditions. The generated future videos remain visually consistent with the ground-truth rollouts, while the predicted trajectories closely follow the reference ego motion. The right-side plots show historical ground-truth motion, ground-truth future motion, predicted future motion, and the current pose. The DPVO [81] consistency protocol is described separately in Sec. E.

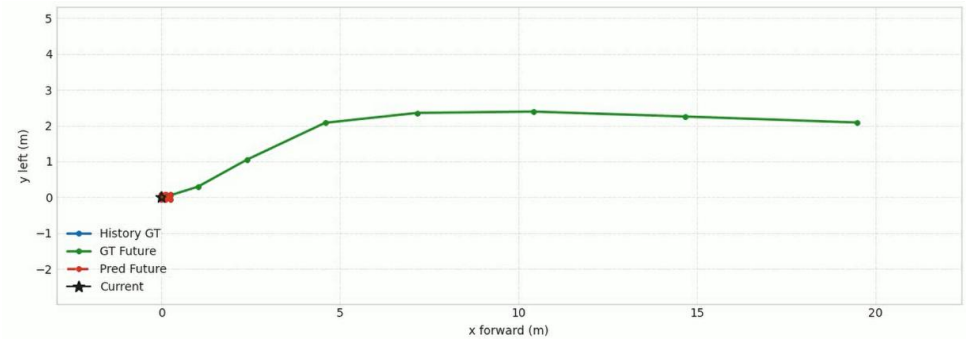
H Limitations

Although UNIVERSE achieves strong simulation-based planning and zero-shot generalization, several limitations remain. First, front-view camera input can miss side or rear-view cues in highly interactive scenes; multi-view extension remains future work. Second, trajectory-only inference reduces the cost of coupled rollout, but the reported 376 ms latency is not a real-vehicle real-time guarantee, and training still relies on a large video-generation backbone. Third, NAVSIM v1 is

(a) UNIVERSE



(b) UNIVERSE Two-DiT



Original trajectory panels enlarged; no curves or values changed.

Figure 8: **Enlarged trajectory comparison for the Bench2Drive example in Fig. 4.** The panels retain the original curves, axes, and legends. The full model predicts a bypass trajectory, whereas the Two-DiT variant stays close to the current position. No trajectories have been regenerated for this enlargement.

non-reactive and the reported cross-domain planning metrics are open-loop. These evaluations do not establish interactive closed-loop performance or real-vehicle safety. Fourth, backbone comparisons change model family as well as size and are not controlled scaling-law experiments. Aggregate results do not include repeated-run confidence intervals, so small numerical differences should not be interpreted as statistically established gains. Finally, NAVSIM Train–Navtest gaps and zero-shot performance support improved generalization but do not uniquely identify reliance on source-domain action priors. Future-scene generation and joint rollout are evaluated as model capabilities, not as validated downstream data-augmentation or safety-certification systems.

I Licenses for Existing Assets

We use public datasets, benchmarks, models, and evaluation tools according to their official licenses and terms of use.

NAVSIM and OpenScene. We use NAVSIM v1 for non-reactive simulation-based planning evaluation. Its code and assets are under Apache-2.0 unless otherwise specified; datasets retain their own licenses. OpenScene code is released under Apache-2.0, whereas the OpenScene data are based on nuPlan and distributed under Creative Commons Attribution-NonCommercial-ShareAlike and the nuPlan Dataset License Agreement for Non-Commercial Use. We follow these licenses and use the data for research training and evaluation.

nuScenes. We use the official nuScenes validation split for zero-shot planning and video-generation evaluation under its data license and terms of use (CC BY-NC-SA 4.0 for non-commercial research).

Bench2Drive and CARLA. We use Bench2Drive for real-to-simulation zero-shot transfer evaluation. The Bench2Drive dataset page states that its assets and code are released under Apache-2.0 unless specified otherwise. Bench2Drive is built on CARLA; CARLA-specific code is released under the MIT license, CARLA-specific assets under CC-BY, and third-party dependencies such as Unreal Engine follow their own licenses. We follow the corresponding licenses and use the benchmark only for research evaluation.

Wan2.2-TI2V-5B. The pretrained Wan2.2-TI2V-5B backbone is used under its Apache-2.0 model-card license and release terms.

Additional video backbones. The backbone comparison uses Cosmos-Predict2.5-2B [82] and Cosmos3-Edge (4B) [83]. Their official model cards provide the corresponding release terms; the NVIDIA Open Model License and OpenMDW-1.1, respectively, apply to these model releases. This comparison does not redistribute their weights.

DPVO. We use DPVO only as an evaluation tool for ego-motion reconstruction from ground-truth and generated videos. It is not used as a training signal. The official DPVO repository is released under the MIT license.

Baselines. Baseline papers and result provenance are cited. Their code, checkpoints, and assets retain the licenses of the original authors.

J Broader Impacts

This work studies video-action world models for autonomous-driving planning. Its potential benefit is improved action generalization under domain shift, while trajectory-only inference reduces the cost of test-time video rollout. Incorrect trajectories or plausible but inaccurate future videos could lead to unsafe decisions if deployed without independent validation; benchmark collision rates do not certify real-world safety. Domain and coverage biases may also cause unequal performance across locations, lighting, or road-user interactions. We recommend use as a research model, compliance with the original data/model licenses, and independent safety checks before any deployment; these are recommendations, not safeguards validated by our experiments. Training large video-action models also requires substantial compute, motivating future work on efficient training, compression, and distillation.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Secs. 3–4.5 support the architecture and experimental claims. The text distinguishes shared-parameter foundations from the new dependency design and limits the interpretation of generalization evidence.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Sec. H discusses front-view coverage, training/inference cost, evaluation scope, missing statistical uncertainty, and the limits of action-prior and downstream-use interpretations.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: This is an architectural and empirical study; it does not claim a theorem or theoretical performance guarantee.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Secs. 3, B, and D describe the model and evaluation protocol. The publicly released implementation, configurations, and trained checkpoint provide additional resources for reproducing the method; the evaluation protocol is detailed in Sec. D.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The implementation, executable configurations, and model checkpoint are publicly released, while the evaluation protocol is detailed in Sec. D.1. Existing benchmark data are obtained from their original providers under the terms summarized in Sec. I.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental setup reports the optimizer, learning rate, batch schedule, resolution, horizon, and sampling steps. Secs. C and D document loss sensitivity, inputs, preprocessing, splits, and the open-loop evaluation settings.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The tables report aggregate metrics without repeated-run error bars or confidence intervals. Sec. H explicitly cautions against interpreting small differences as statistically established gains.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: The paper reports NVIDIA H20 training GPUs, the optimization schedule, and measured latency and memory in Table 6, but not per-run GPU counts, wall-clock training time, or total project compute.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The reported study uses existing research datasets and simulation, credits external assets, and discusses limitations and potential harms in Secs. H–J. No new participant study or real-vehicle deployment is reported.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Sec. J discusses generalization and efficiency benefits as well as unsafe predictions, coverage bias, deployment cautions, and computational cost.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: The paper discusses research-use and deployment cautions, but does not report implemented or validated release-control safeguards for the model. Recommendations in Sec. J are not represented as completed safety measures.

    Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [No]

    Justification: Sec. I credits the major assets and states their licenses, including the additional video backbones. A complete per-version inventory of every baseline implementation and auxiliary component is not provided in this version.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The released implementation and UNIVERSE checkpoint are documented with setup and inference instructions. The manuscript specifies model and data settings (Secs. 3 and D.1); no new dataset is introduced.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The reported experiments use existing driving datasets and simulation, rather than new crowdsourcing or participant studies.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No new human-subject experiment or participant recruitment is reported in this work.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The reported core method does not use an LLM as an important, original, or non-standard methodological component. The frozen pretrained text encoder used for conditioning is described in Sec. 3.1.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.