HORIZONDRIVE: SELF-CORRECTIVE AUTOREGRESSIVE WORLD MODEL FOR LONG-HORIZON DRIVING SIMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Closed-loop driving simulation requires real-time interaction beyond short offline clips, pushing current driving world models toward autoregressive (AR) rollout. As generated frames become the context for subsequent chunks, prediction errors accumulate over time. The key difficulty is that a teacher trained only on clean histories provides unreliable corrective guidance when conditioned on the student's generated frames. A natural question is: *can a teacher provide reliable corrective guidance throughout a student's long AR rollout?* We address this with HorizonDrive, a teacher-centric training-and-distillation framework for AR driving simulation. Through scheduled rollout recovery (SRR), we mine semantically degraded yet layout-preserving states from long driving rollouts, turning their retained scene structure into corrective supervision for training a recovery-capable teacher. Further, teacher rollout distillation (TRD) transfers this capability to a short-chunk, few-step student through fixed-window distribution matching, enabling continual correction throughout long AR rollouts with bounded memory and per-window computation. HorizonDrive natively supports minute-scale AR rollout under bounded memory; on nuScenes, HorizonDrive reduces FID by 52% and FVD by 37%, and lowers ARE and DTW by 21% and 9% relative to the strongest long-horizon streaming baselines. Open- and closed-loop evaluations further show improved controllability and downstream planning utility.

1 INTRODUCTION

Driving world models aim to serve as closed-loop simulation testbeds, where generated environments respond to an agent's actions in real time. This requires long-horizon autoregressive (AR) rollout, in which the model repeatedly generates short future chunks, receives updated controls, and feeds its own predictions back as context. Such deployment induces exposure bias: each chunk is conditioned on imperfect generated frames rather than ground-truth observations, causing small errors to recursively compound into visual artifacts, geometric inconsistency, and semantic drift. As a result, even state-of-the-art driving world models (*e.g.*, Ren et al. (2025); Gao et al. (2025a)) degrade quickly, far short of the minute-scale horizons needed for meaningful closed-loop evaluation.

Existing anti-drifting methods expose AR video models to imperfect histories. Self-Forcing (Huang et al., 2025) and Self-Forcing++ (Cui et al., 2025) train students along their own rollouts, while Resampling Forcing (Guo et al., 2026) and SVI (Li et al., 2026) simulate or replay model-generated errors. However, these methods do not explicitly exploit the scene structure retained in long-rollout failures, and their corrective guidance can become unreliable on severely degraded contexts. This issue is especially acute in driving, where ego-motion and changing controls rapidly amplify drift.

A natural question then arises: *how can we sample informative histories to learn reliable long-horizon correction?* Our key insight is that long driving rollouts expose semantically degraded yet layout-preserving states whose retained structure provides effective recovery cues. We further instantiate this insight in **HorizonDrive**, a teacher-centric training-and-distillation framework that transfers the teacher's recovery capability to a short-chunk student under bounded memory.

In the first stage, *Scheduled Rollout Recovery (SRR)* equips a standard driving world model with rollout recovery. SRR mines long-rollout histories that are semantically degraded yet preserve the

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

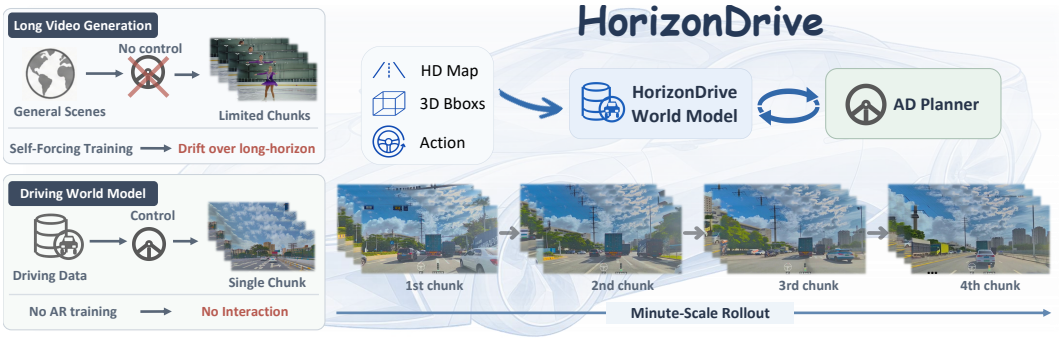

Figure 1: **Comparison with general long-video generators and driving world models.** The former lack driving-specific control and suffer from rollout drift; the latter are limited in AR rollout. HorizonDrive combines driving control with stable long-horizon AR generation for real-time interactive simulation.

underlying scene layout, and pairs them with aligned ground-truth futures and controls from the same driving clip. A coarse-to-fine recovery curriculum first targets severe late-rollout failures and then shifts toward subtler early-stage errors, while smoothing the transition between generated histories and ground-truth futures to preserve temporal continuity. Because recovery is learned within a fixed temporal window, the resulting teacher can repeatedly correct AR rollouts without frame sinks or a larger per-pass memory footprint.

In the second stage, *Teacher Rollout Distillation (TRD)* uses distribution matching distillation to transfer the SRR-trained teacher's corrective ability to a few-step student. During distillation, the student follows its own long AR rollout; the teacher scores fixed-size windows of student outputs conditioned on student-generated history and corresponding driving controls, providing repeated corrective supervision under bounded memory. A noise-truncated CFG schedule modulates teacher guidance across noise levels and training stages to balance controllability and visual quality. The student retains short generation chunks for frequent control updates at lower inference cost.

Overall, HorizonDrive enables interactive driving simulation through stable, action-controllable long-horizon AR rollout (Fig. 1). Its few-step student generates 10-frame chunks with bounded memory, conditioned on updated HD maps, bounding boxes, and ego actions at each step. On nuScenes (where the per-clip length is bounded to $\sim$20 s by the dataset), HorizonDrive reduces FID by 52% and FVD by 37%, and lowers ARE and DTW by 21% and 9% relative to the strongest evaluated long-horizon streaming baselines. It remains competitive with single-pass driving video generators on short-clip evaluation, and minute-scale rollout is demonstrated qualitatively in the appendix. Open- and closed-loop evaluations further show that its generated scenes better follow the specified driving controls and support downstream planning.

Our contributions are threefold: (1) We identify semantically degraded yet layout-preserving histories in driving rollouts as a structured source of corrective supervision, and propose SRR, which mines these histories, pairs them with aligned ground-truth futures, and trains a recovery-capable driving teacher through a joint local-and-global curriculum. (2) We introduce TRD, a long-horizon framework for distilling the recovery-capable SRR teacher into a short-chunk, few-step student. Fixed-window DMD supervision is applied along the student's own rollout with bounded memory, and a noise-truncated CFG schedule regulates teacher guidance. (3) We systematically evaluate HorizonDrive on nuScenes across visual quality, spatio-temporal consistency, and open- and closed-loop driving performance, demonstrating stronger long-horizon generation and downstream planning utility.

2 Related Work

Conditional Video Generation for Autonomous Driving. Recent progress in video synthesis has been largely enabled by diffusion-based generative models (Peebles & Xie, 2023; Brooks et al., 2024; Gao et al., 2025b; RunwayML, 2024; Yang et al., 2024; Kong et al., 2024; Ma et al., 2025; Wan et al., 2025; Team et al., 2025). Driving video generation requires both visual realism and adherence to

scene conditions and driving controls. Existing methods incorporate geometric conditions, such as HD maps, 3D boxes, and occupancy, to specify scene layout (Wen et al., 2024; Wang et al., 2024b; Gao et al., 2024a; Zhao et al., 2025; Gao et al., 2025a), or use ego-motion and trajectory cues to guide vehicle behavior (Hu et al., 2023; Lu et al., 2024; Gao et al., 2024b; Zhang et al., 2025; Russell et al., 2025; Zhan et al., 2026). Yet, most existing methods are limited by short generation horizons or high inference cost, restricting their use in closed-loop simulation.

Error-Robust Autoregressive Video Generation. Long-video methods mitigate accumulated errors through temporal extension (Qiu et al., 2023; Kim et al., 2024), noise-based history corruption (Chen et al., 2024; Liu et al., 2025b), or error recycling (Guo et al., 2025; Li et al., 2025). Closely related to SRR, Stable Video Infinity (SVI) (Li et al., 2025) recycles prediction residuals estimated via one-step bidirectional integration through replay banks. SRR instead uses clip-specific histories from contiguous multi-step AR rollouts, capturing both generic prediction errors and accumulated semantic drift. These histories are paired with ground-truth futures from the same driving clip under the corresponding controls to learn targeted recovery. Its joint local-and-global curriculum smooths prediction-to-ground-truth transitions and shifts training from severe late-rollout recovery to finer early-rollout correction. This prepares a driving teacher to provide corrective supervision under prediction-corrupted histories.

Efficient Video Generation via Diffusion Distillation. Distribution matching (Yin et al., 2024b;a; Liu et al., 2025a; Wu et al., 2026) and consistency-based methods (Song et al., 2023; Geng et al., 2024; 2025; Zheng et al., 2025) reduce diffusion sampling to few-step generation. Self-Forcing (Huang et al., 2025) incorporates student self-rollout into training, while Self-Forcing++ (Cui et al., 2025) and LongLive (Yang et al., 2025) extend training to long student rollouts using short-window teacher supervision. Our focus is on preparing the teacher for control-conditioned recovery before distillation. TRD then uses standard DMD to apply this recovery-trained teacher along the student's rollout, conditioned on preceding student-generated histories and driving controls, with a noise-truncated CFG schedule to regulate guidance.

3 METHODS

Problem formulation. We aim to build a real-time interactive autoregressive (AR) driving world model in a video-VAE latent space, where $\mathbf{z}_{1:T}$ denotes the AR context window and $\mathbf{c}_{T+1:T+K}$ denotes future driving controls over the next K frames. At each step, the model predicts the next K-frame chunk as

$$\hat{\mathbf{z}}_{T+1:T+K} \sim p_\theta \left(\mathbf{z}_{T+1:T+K} \mid \mathbf{z}_{1:T}, \mathbf{c}_{T+1:T+K} \right), \tag{1}$$

and appends it to the history buffer for the next step.

HorizonDrive first trains a conditional driving world model, then makes it rollout-capable through SRR, and finally distills its long-horizon behavior into a few-step student through TRD.

3.1 CONDITIONAL DRIVING WORLD MODEL TRAINING

The backbone of our conditional driving world model is a pretrained text-to-video (T2V) diffusion transformer with full bidirectional attention. To turn it into a video continuation model without modifying its architecture, we partition each training clip into a T-frame condition window and a K-frame generation chunk, and assign different noise levels to the two parts: condition latents $\mathbf{z}_{1:T}$ are kept clean with noise level $t{=}0$, while chunk latents $\mathbf{z}_{T+1:T+K}$ are noised and supervised with the flow-matching loss in Eq. (9).

To enable conditional control, we follow disentangled control (Ren et al., 2025; Russell et al., 2025; Zhan et al., 2026): spatial scene structure (HD map, bounding boxes) is injected as additive layout tokens via a zero-initialized projector, while the residual ego-action signal $\boldsymbol{a}{=}(\Delta x, \Delta y, \Delta\text{yaw})$ is injected through AdaLN-style gating (Peebles & Xie, 2023). Architectural details and the exact injection forms are deferred to Sec. D.

After training, we obtain a standard controllable driving world model, denoted $\mathcal{G}_0$. At inference, $\mathcal{G}_0$ rolls out long sequences via a sliding window: the most recent T latents form the next condition, a new K-frame chunk is generated via Eq. (1), and the context window is then shifted forward by K

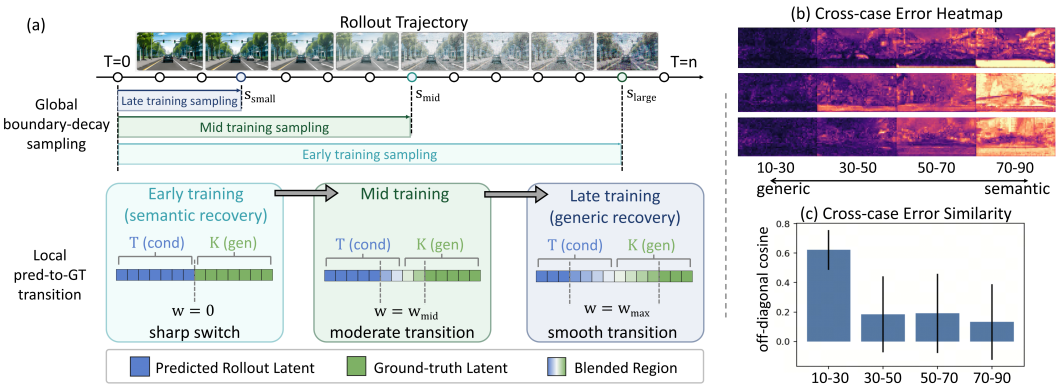

Figure 2: **Scheduled rollout recovery (SRR).** (a) Teacher rollouts support global boundary-decay sampling and a local hard-to-smooth prediction-to-GT curriculum. (b,c) Cross-case error magnitudes and similarities motivate learning first from severe late-rollout drift, then from more shared early-rollout errors.

frames for the next AR step. However, since $\mathcal{G}_0$ is trained only with clean ground-truth histories, direct autoregressive rollout suffers from exposure bias.

3.2 SRR: SCHEDULED ROLLOUT RECOVERY TRAINING

To obtain a recovery-capable teacher, we replace clean-history-only training with semantically degraded yet layout-preserving conditions sampled from long rollout trajectories of $\mathcal{G}_0$. Starting from a ground-truth latent history with length T, we roll out $\mathcal{G}_0$ for N autoregressive steps. At each step, the model predicts a chunk of K future latents and appends them to a fixed-length history buffer of size T:

$$\hat{\mathbf{z}}_{s_n+1:s_n+K} = \mathcal{G}_0\left(\hat{\mathbf{z}}_{s_n-T+1:s_n}, \mathbf{c}_{s_n+1:s_n+K}\right), \quad n = 1, \ldots, N, \tag{2}$$

where $s_n = T + (n-1)K$. This produces an N-step rollout trajectory $\hat{\mathbf{z}}_{T+1:T+NK}$, in which prediction errors accumulate across AR steps. Here and below, $\hat{\mathbf{z}}$ denotes latents predicted along the model's own AR rollout, whereas $\mathbf{z}^\star$ denotes the corresponding clean ground-truth target latents.

For a sampled starting index s for generation, we replace the conditioning history with rollout predictions while keeping the supervision target unchanged:

$$\tilde{\mathbf{z}}_{s-T+1:s} = \hat{\mathbf{z}}_{s-T+1:s}, \qquad \mathbf{z}^\star_{s+1:s+K} = \mathbf{z}_{s+1:s+K}. \tag{3}$$

As illustrated in Fig. 2, we first smooth the local pred-to-GT transition within each sampled segment; we then separately schedule the global sampling boundary along the rollout trajectory.

Local pred-to-GT transition. Directly switching from the predicted history to the ground-truth future creates an abrupt temporal boundary. To smooth this transition, we introduce a pred-to-GT blending window of radius w around the generation boundary s. The final mixed training sequence $\bar{\mathbf{z}}_i$ is constructed as

$$\bar{\mathbf{z}}_i = \begin{cases} \tilde{\mathbf{z}}_i, & s-T+1 \le i \le s-w, \\ \alpha_i \tilde{\mathbf{z}}_i + (1-\alpha_i)\mathbf{z}^\star_i, & s-w+1 \le i \le s+w, \\ \mathbf{z}^\star_i, & s+w+1 \le i \le s+K, \end{cases} \tag{4}$$

where α_i linearly decreases from 1 to 0 within the transition window. This constructs a continuous temporal bridge from rollout-degraded latents to ground-truth latents around the boundary.

We further schedule the blending radius during training. At the beginning, we set $w = 0$, which exposes the model to a sharp boundary and encourages direct recovery from large rollout-induced deviations. As training progresses, w is gradually increased, expanding the transition region and shifting the task toward smoother, fine-grained correction.

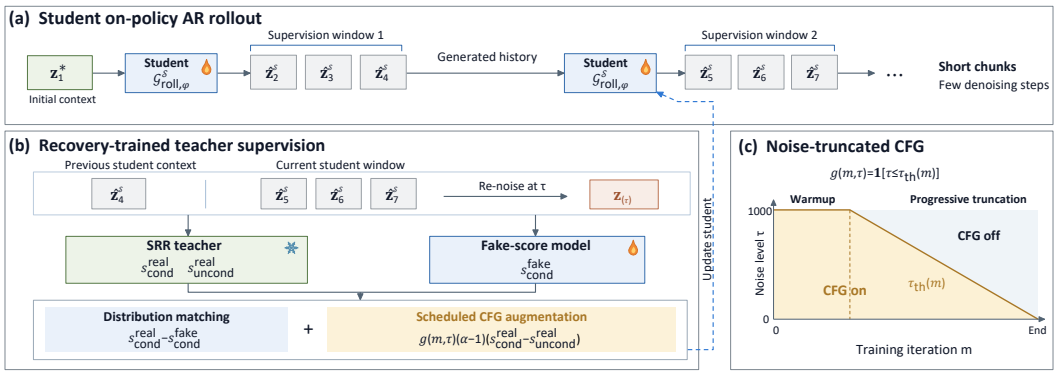

Figure 3: **Teacher Rollout Distillation (TRD).** The student generates autoregressively in short chunks. For each fixed-size supervision window, the recovery-trained teacher scores re-noised student outputs conditioned on preceding student-generated history and driving controls. DMD combines distribution matching with scheduled CFG augmentation.

Global boundary-decay sampling. We further schedule the sampled generation boundary s along the rollout trajectory. To characterize the evolution of rollout errors, we group s into temporal intervals spanning early to late rollout positions. Our rollout-error analysis reveals two complementary properties. First, cross-case error magnitude becomes progressively stronger toward later boundaries, indicating that longer autoregressive rollout leads to more severe accumulated degradation and semantic drift. Second, off-diagonal cosine similarity is highest in the early interval, showing that early-stage errors are more shared across cases, while later-stage semantic errors are more case-specific. Motivated by these observations, we adopt a boundary-decay curriculum: training starts from larger s, where the model is forced to recover from severe long-horizon semantic drift, and s is gradually decayed toward earlier rollout positions. This schedule first builds robustness to accumulated semantic failures and then refines the model on more generic, cross-case consistent rollout degradation.

Together with the local transition curriculum, this schedule trains $\mathcal{G}_0$ to recover from rollout-induced degradation, yielding the rollout-capable teacher $\mathcal{G}_{\text{roll}}$ for subsequent distillation.

3.3 TRD: Teacher Rollout Distillation

Distillation from the recovery-trained teacher. Despite its improved rollout stability, $\mathcal{G}_{\text{roll}}$ remains constrained by the high inference cost of multi-step diffusion sampling, making it impractical for real-time interactive simulation. We therefore use Teacher Rollout Distillation (TRD), built on standard DMD, to transfer its recovery capability into a few-step student for real-time deployment. Both the teacher $\mathcal{G}^{\mathcal{T}}_{\text{roll}}$ and the student $\mathcal{G}^{\mathcal{S}}_{\text{roll}}$ are initialized from $\mathcal{G}_{\text{roll}}$. The teacher is kept frozen during student optimization. They share the same conditioning window length T, but use different chunk sizes during generation. The teacher uses a longer chunk size $K^{\mathcal{T}}$, while the student uses a shorter chunk size $K^{\mathcal{S}}$, with $K^{\mathcal{T}} > K^{\mathcal{S}}$.

For long-horizon distillation, we apply the recovery-trained teacher to successive windows along the student's rollout, retaining a fixed $(T, K^{\mathcal{T}})$ window. Each teacher supervision window is conditioned on the T context frames immediately preceding it and the corresponding driving controls. After the initial clean context, these history frames come from the student's own rollout. The student generates autoregressively with the fixed $(T, K^{\mathcal{S}})$ window:

$$\hat{\mathbf{z}}^{\mathcal{S}}_{s_n+1:s_n+K^{\mathcal{S}}} = \mathcal{G}^{\mathcal{S}}_{\text{roll},\phi}\left(\hat{\mathbf{z}}^{\mathcal{S}}_{s_n-T+1:s_n}, \mathbf{c}_{s_n+1:s_n+K^{\mathcal{S}}}\right), \quad n = 1, \ldots, N, \tag{5}$$

where ϕ denotes the student parameters and $s_n = T + (n-1) \cdot K^{\mathcal{S}}$.

Whenever the accumulated student rollout covers a teacher-length interval $K^{\mathcal{T}}$, we apply the DMD gradient to the latest $K^{\mathcal{T}}$ frames and immediately backpropagate. The frozen recovery-trained teacher evaluates the re-noised student outputs, conditioned on the student-generated history preceding this interval and the corresponding controls, to provide the real-score component of the DMD update.

Repeating this procedure along the student's trajectory transfers the teacher's recovery capability through fixed-size supervision windows.

Noise-truncated CFG schedule. To mitigate guidance-induced oversaturation during long rollouts, we adopt a noise-truncated CFG schedule. Inspired by the separation of distribution matching and CFG augmentation in Decoupled DMD (Liu et al., 2025a), we restrict the CFG term to noise levels below a threshold τ_{th}. We retain full-range CFG during an initial warmup and then progressively decrease τ_{th}, restricting guidance to lower noise levels. This schedule is designed to support conditional learning early in training and reduce guidance-induced visual distortion later.

Using the standard DMD gradient in (10) with the scheduled CFG term, the student update is written as

$$\nabla_\phi \mathcal{L}_{\text{TRD}} = -\mathbb{E}_\tau \left[\left(\Delta s_{\text{DM}} + g(\tau, m) \Delta s_{\text{CFG}} \right) \frac{\partial \mathcal{G}^{\mathcal{S}}_{\text{roll},\phi}}{\partial \phi} \right], \tag{6}$$

where $\Delta s_{\text{DM}} = s^{\text{real}}_{\text{cond}} - s^{\text{fake}}_{\text{cond}}$ and $\Delta s_{\text{CFG}} = s^{\text{real}}_{\text{cond}} - s^{\text{real}}_{\text{uncond}}$ denote the distribution-matching and CFG augmentation directions, respectively. All scores are evaluated at $\mathbf{z}_{(\tau)}$, the student-generated latent sequence over the current teacher-length supervision window, re-noised to level τ. The scheduled weight $g(\tau, m) = (\alpha - 1)\mathbf{1}_{\{\tau \le \tau_{\text{th}}(m)\}}$ uses CFG scale α and training iteration m. The real scores are provided by the frozen teacher's conditional and unconditional CFG branches, while $s^{\text{fake}}_{\text{cond}}$ estimates the conditional score of the current student distribution. Both conditional scores use the same student-generated history preceding the supervision window and the corresponding driving controls; score arguments are omitted for brevity. The schedule restricts CFG augmentation to $\tau \le \tau_{\text{th}}(m)$, while distribution matching remains active across all sampled noise levels.

4 Experiments

4.1 Implementation

Implementation details. HorizonDrive is built on Wan 2.1 1.3B (Wan et al., 2025) with the driving control modules of Sec. 3.1 and no text conditioning. For TRD, the teacher and student share a context window T=11, with chunk sizes $K^{\mathcal{T}}$=40 (50-step teacher) and $K^{\mathcal{S}}$=10 (four-step student); the student rolls out N=20 AR steps during training. We use 700 nuScenes (Caesar et al., 2020) videos for training and 150 for validation. The per-clip length of $\sim$20 s bounds our quantitative horizon, while minute-scale rollouts are shown qualitatively on a self-collected dataset (Sec. M). At 256×512 on one RTX 5090, the student runs at 5.6 FPS in PyTorch and 10.8 FPS with TensorRT–FP8. Full training schedules, cache costs, and implementation details are in Secs. D and I.

Evaluation metrics. We report **visual quality** via FID and FVD over the full rollout, as well as VBench (Huang et al., 2024). For **spatio-temporal consistency**, we recover per-frame poses with VGGT (Wang et al., 2025) and report **ARE** (mean geodesic rotation error against ground truth) and **DTW** (dynamic time warping distance between predicted and ground-truth ego-motion trajectories (Keogh & Pazzani, 2000)); these metrics apply only to driving-controlled methods. Full definitions are in Sec. E, and direct control and planner metrics are reported in Sec. 4.3.

4.2 Comparisons

Comparison methods. We compare three baseline groups on nuScenes (protocols in Sec. F). *(i) Long-horizon interactive world model frameworks* include Matrix-Game3 (Wang et al., 2026), Helios (Yuan et al., 2026), Causal-Forcing (Zhu et al., 2026), HY-WorldPlay (Sun et al., 2025), and LingBot-World (Team et al., 2026). They do not natively accept our driving controls, so we condition them only on initial frames and report visual-quality metrics. *(ii) Long-horizon streaming methods* include Self-Forcing (Huang et al., 2025), Self-Forcing++ (Cui et al., 2025), and LongLive (Yang et al., 2025). We retrain each with the same backbone, data, and driving control modules as HorizonDrive to compare long-horizon training frameworks under a common driving setup. *(iii) Driving video generators* include DriveDreamer (Wang et al., 2024a), Panacea (Wen et al., 2024), DreamForge (Mei et al., 2024), Vista (Gao et al., 2024b), and MagicDrive-V2 (Gao et al., 2025a). Their original fixed-length results are grouped by short- and long-video settings.



Figure 4: **Long-horizon rollout comparison with streaming video generation methods.** Our method preserves clearer scene structure, more stable object geometry, and better visual quality over time.

Visual quality of initial-frame-conditioned world models. The top group of Tab. 1 yields FID of 30.53–49.07 and FVD of 218.23–580.72, with Image Quality scores of 55.55–60.44 compared with HorizonDrive's 62.50. Given the differing conditioning protocols, these results provide a reference for visual quality on nuScenes.

Improved long-horizon streaming with HorizonDrive. Under matched backbone, data, and driving controls, HorizonDrive reduces FID/FVD by 52%/37% and ARE/DTW by 21%/9% relative to the best evaluated streaming baseline for each metric (Tab. 1). Qualitative results (Fig. 4) show clearer structure and less drift. Our framework first trains the teacher for recovery from prediction-corrupted histories through SRR, then transfers this capability to the student through TRD. Per-chunk FID (Fig. 6) further reveals late-horizon degradation in the streaming baselines that is less apparent in aggregate metrics.

Driving-specific methods. Table 5 compares HorizonDrive with domain-specific generators under their original protocols. For short videos, our four-step student with $N{=}1$ generates 21 frames, attaining the best listed FVD (84.53) and competitive FID (12.54), while jointly supporting HD maps, boxes, and ego actions. For long videos, its $N{=}20$ AR rollout produces 211 frames with FID/FVD of $13.82/92.99$, compared with $20.91/94.84$ for MagicDrive-V2's single-pass 241-frame generation. Although the settings differ, these results suggest that few-step AR generation can remain competitive with many-step, single-pass driving generators across longer horizons.

4.3 Downstream Perception and Planning Evaluation

To evaluate downstream utility, we generate visual rollouts conditioned on the recorded ego trajectories and apply perception models to the generated observations. Road/lane IoU and box mAP measure agreement with ground-truth map and object annotations, while planner ADE/FDE measure the discrepancy between Epona trajectories predicted from generated and real observations. As shown in Tab. 2, HorizonDrive achieves the best results among the evaluated methods across all metrics, indicating better preservation of scene information for downstream perception and planner responses closer to those obtained from real observations.

Table 1: **Long-horizon autoregressive generation on nuScenes val.** Top: world-model frameworks conditioned on initial frames only (ARE/DTW not applicable). Bottom: streaming methods retrained with our backbone, data, and driving controls. HorizonDrive uses the final four-step TRD student. Best results are in **bold**.

Method	Visual		VBench			ST Cons.	
	FID↓	FVD↓	Qual.↑	Mot.↑	Img.↑	ARE↓	DTW↓
Long-horizon interactive world model frameworks							
Matrix-Game3 (Wang et al., 2026)	35.69	338.22	78.99	93.78	60.44	N/A	N/A
Helios (Yuan et al., 2026)	30.53	218.23	79.02	95.03	58.82	N/A	N/A
Causal-Forcing (Zhu et al., 2026)	49.07	373.29	74.35	92.42	59.00	N/A	N/A
HY-WorldPlay (Sun et al., 2025)	33.51	580.72	76.58	**99.48**	58.60	N/A	N/A
LingBot-World (Team et al., 2026)	37.67	325.55	77.08	92.87	55.55	N/A	N/A
Long-horizon streaming video generation methods (retrained on our base model and data)							
Self-Forcing (Huang et al., 2025)	41.53	161.00	79.27	94.17	59.65	3.47	6.22
Self-Forcing++ (Cui et al., 2025)	28.84	147.57	79.47	93.92	60.25	3.78	3.61
LongLive (Yang et al., 2025)	29.05	161.41	79.35	93.46	60.80	3.28	3.65
HorizonDrive (Ours)	**13.82**	**92.99**	**79.53**	93.85	**62.50**	**2.60**	**3.27**

Table 2: **Perception consistency and planner-response agreement on nuScenes val.** Road/lane IoU measures agreement with map annotations, box mAP measures agreement with object annotations, and planner ADE/FDE measures the discrepancy between Epona trajectories predicted from generated and real observations.

Method	Road IoU↑	Lane IoU↑	Box mAP↑	ADE↓	FDE↓
Self-Forcing	0.7828	0.2643	10.56	0.6351	1.2130
Self-Forcing++	0.7918	0.2548	11.27	0.4770	0.9177
LongLive	0.7969	0.2758	11.41	0.3568	0.6929
HorizonDrive	**0.8390**	**0.3512**	**18.47**	**0.3139**	**0.6158**

As a supplementary closed-loop evaluation, we conduct a 10-step planner-in-the-loop test, feeding each predicted Epona trajectory back as the next action. HorizonDrive improves PDMS from 82.64 to 85.57 compared with Self-Forcing++ (see Tab. 13 in the appendix).

4.4 Ablation studies

We first evaluate the two SRR curricula, then examine how SRR-trained teacher and student initialization affect distillation. With both initializations fixed to SRR-trained weights, we further evaluate the training rollout horizon and the CFG schedule. The CFG threshold τ_{th} controls the maximum noise level at which CFG is applied in the DMD loss; we compare four schedules: **None** (CFG disabled), **Full** (τ_{th}=1000 throughout), **Early** (τ_{th} decays 1000→0 from step 0), and **Delayed** (τ_{th} decays 1000→0 after a 100-step warmup).

The two SRR curricula provide complementary benefits. Table 3 evaluates the proposed curricula within the same rollout-recovery training setup. The fixed-w, uniform-boundary variants provide recovery-training baselines without either curriculum. A hard prediction-to-GT boundary (w=0) achieves the lowest FID but severely worsens ARE/DTW, indicating a trade-off between distributional fidelity and temporal consistency. Expanding w:0→8 balances the two objectives, and adding boundary decay improves all four metrics over uniform boundary sampling. Full SRR also improves all four metrics over both fixed smooth-transition baselines, demonstrating benefits from the curricula beyond exposure to self-generated histories alone. Under matched teacher architecture and initialization, data and training budget, SRR outperforms SVI across all four metrics (Tab. 8), reducing FID/FVD from $25.39/173.98$ to $13.73/94.84$ and ARE/DTW from $2.61/3.31$ to $1.95/2.99$. These results support the effectiveness of clip-specific contiguous rollout histories combined with the proposed curricula for improving both visual quality and geometric consistency.

Table 3: **Teacher enhancement with SRR on nuScenes val.** Transition and boundary-sampling ablations under identical self-rollout evaluation. The first row is the teacher-forcing base model.

Transition schedule	Boundary sampling	FID↓	FVD↓	ARE↓	DTW↓
–	–	31.13	164.17	4.26	6.58
Fixed w=0	Uniform	**13.08**	118.03	11.78	19.43
Fixed w=4	Uniform	16.20	132.51	2.23	3.31
Fixed w=8	Uniform	17.58	150.70	2.17	<u>3.08</u>
w:0→8	Uniform	15.52	<u>103.02</u>	<u>2.05</u>	3.16
w:0→8	Boundary decay	<u>13.73</u>	**94.84**	**1.95**	**2.99**

Table 4: **Effect of SRR initialization on TRD on nuScenes val.** Student and teacher are initialized from Base or SRR weights; all variants use N=20 and delayed CFG. Joint SRR initialization performs best.

Student init.	Teacher init.	FID↓	FVD↓	ARE↓	DTW↓
Base	Base	19.24	141.88	2.76	<u>3.30</u>
SRR	Base	20.34	128.77	3.15	3.39
Base	SRR	<u>14.44</u>	<u>107.54</u>	<u>2.75</u>	3.80
SRR	SRR	**13.82**	**92.99**	**2.60**	**3.27**

SRR-trained supervision and student initialization play complementary roles. In Tab. 4, student-only SRR initialization improves FVD but worsens FID, ARE, and DTW. Teacher-only SRR initialization improves FID/FVD from $19.24/141.88$ to $14.44/107.54$, while ARE changes little and DTW worsens from 3.30 to 3.80. With the SRR-initialized student fixed, replacing the Base teacher with the SRR teacher improves all four metrics. Joint SRR initialization achieves the best overall results, showing that the gains cannot be attributed to student initialization alone. The resulting four-step student achieves comparable FID/FVD to the 50-step SRR teacher ($13.73/94.84$).

Longer training rollouts improve deployment-time stability. Restricting training to N=1 limits its exposure to errors accumulated over long rollouts, yielding ARE of 3.39 and DTW of 5.42 (Tab. 6). Increasing N from 1 to 4 to 20 reduces ARE monotonically ($3.39\rightarrow3.03\rightarrow2.60$) and DTW ($5.42\rightarrow3.66\rightarrow3.27$). FVD is not monotonic at short horizons but improves substantially at N=20 (92.99 versus 139.28 at N=4), supporting long AR training chains for stable deployment. All variants use the same SRR-initialized teacher and student, delayed CFG, and total number of DMD backpropagation updates.

CFG augmentation benefits from noise truncation. Full-range CFG worsens FVD from 110.81 without CFG to 184.06 (Tab. 7). Early decay improves visual quality over full-range CFG but does not outperform disabling CFG in FID/FVD. Delayed decay achieves the best FID (13.82), FVD (92.99), and ARE (2.60), while Early achieves slightly better DTW (3.13 versus 3.27). These results support retaining full-range guidance during warmup before progressively restricting its noise range; we therefore use Delayed by default.

5 Conclusion and Limitations

We presented HorizonDrive, which first stabilizes a driving world model under its own predictions through SRR and then transfers the teacher's recovery capability to a few-step student through TRD along the student's own rollout. This enables reliable AR supervision under bounded memory and efficient interactive generation. HorizonDrive demonstrates minute-scale rollouts and outperforms the evaluated streaming distillation baselines on visual quality and geometric consistency, highlighting its practical potential as an environment for closed-loop driving simulation and evaluation.

SRR currently relies on offline collection of rollout errors; extending error collection and recovery training to online interaction trajectories remains a promising direction.

AI USE STATEMENT

Generative AI tools were used for language polishing and figure preparation. The authors reviewed all AI-assisted content and take full responsibility for the final manuscript, including its text, figures, claims, and reported results.

REPRODUCIBILITY STATEMENT

We provide pseudocode for SRR and TRD in Sec. C, implementation and cache-cost details in Sec. D, metric definitions and baseline protocols in Secs. E and F, and stage-wise hyperparameters in Sec. I. The main text also reports inference hardware, resolution, throughput, and a controlled teacher–student quality/efficiency comparison.

REFERENCES

Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. *arXiv preprint arXiv:2209.15571*, 2022.

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

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024.

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*, pp. 11621–11631, 2020.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Self-forcing++: Towards minute-scale high-quality video generation. *arXiv preprint arXiv:2510.02283*, 2025.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

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

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*, pp. 28135–28144, 2025a.

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, 2024b.

Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025b.

Zhengyang Geng, Ashwini Pokle, William Luo, Justin Lin, and J Zico Kolter. Consistency models made easy. *arXiv preprint arXiv:2406.14548*, 2024.

Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *arXiv preprint arXiv:2505.13447*, 2025.

Yuwei Guo, Ceyuan Yang, Hao He, Yang Zhao, Meng Wei, Zhenheng Yang, Weilin Huang, and Dahua Lin. End-to-end training for autoregressive video diffusion via self-resampling. *arXiv preprint arXiv:2512.15702*, 2025.

Yuwei Guo, Ceyuan Yang, Hao He, Yang Zhao, Meng Wei, Zhenheng Yang, Weilin Huang, and Dahua Lin. End-to-end training for autoregressive video diffusion via self-resampling. In *European Conference on Computer Vision*, pp. 324–344. Springer, 2026.

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.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

Eamonn J Keogh and Michael J Pazzani. Scaling up dynamic time warping for datamining applications. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 285–289, 2000.

Jihwan Kim, Junoh Kang, Jinyoung Choi, and Bohyung Han. Fifo-diffusion: Generating infinite videos from text without training. *Advances in Neural Information Processing Systems*, 37:89834–89868, 2024.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Wuyang Li, Wentao Pan, Po-Chien Luan, Yang Gao, and Alexandre Alahi. Stable video infinity: Infinite-length video generation with error recycling. *arXiv preprint arXiv:2510.09212*, 2025.

Wuyang Li, Wentao Pan, Po-Chien Luan, Yang Gao, and Alexandre Alahi. Stable video infinity: Infinite-length video generation with error recycling. In *International Conference on Learning Representations*, volume 2026, pp. 23406–23432, 2026.

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

Dongyang Liu, Peng Gao, David Liu, Ruoyi Du, Zhen Li, Qilong Wu, Xin Jin, Sihan Cao, Shifeng Zhang, Hongsheng Li, et al. Decoupled dmd: Cfg augmentation as the spear, distribution matching as the shield. *arXiv preprint arXiv:2511.22677*, 2025a.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. *arXiv preprint arXiv:2509.25161*, 2025b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

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*, pp. 329–345. Springer, 2024.

Guoqing Ma, Haoyang Huang, Kun Yan, Liangyu Chen, Nan Duan, Shengming Yin, Changyi Wan, Ranchen Ming, Xiaoniu Song, Xing Chen, et al. Step-video-t2v technical report: The practice, challenges, and future of video foundation model. *arXiv preprint arXiv:2502.10248*, 2025.

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

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

Haonan Qiu, Menghan Xia, Yong Zhang, Yingqing He, Xintao Wang, Ying Shan, and Ziwei Liu. Freenoise: Tuning-free longer video diffusion via noise rescheduling. *arXiv preprint arXiv:2310.15169*, 2023.

Xuanchi Ren, Yifan Lu, Tianshi Cao, Ruiyuan Gao, Shengyu Huang, Amirmojtaba Sabour, Tianchang Shen, Tobias Pfaff, Jay Zhangjie Wu, Runjian Chen, Seung Wook Kim, Jun Gao, Laura Leal-Taixe, Mike Chen, Sanja Fidler, and Huan Ling. Cosmos-drive-dreams: Scalable synthetic driving data generation with world foundation models, 2025. URL https://arxiv.org/abs/2506.09042.

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

GA RunwayML. Introducing gen-3 alpha: a new frontier for video generation, 2024.

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.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. 2023.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, Yunhong Wang, Jun Zhang, Tengfei Wang, and Chunchao Guo. Worldplay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025.

Meituan LongCat Team, Xunliang Cai, Qilong Huang, Zhuoliang Kang, Hongyu Li, Shijun Liang, Liya Ma, Siyu Ren, Xiaoming Wei, Rixu Xie, et al. Longcat-video technical report. *arXiv preprint arXiv:2510.22200*, 2025.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, et al. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

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*, pp. 55–72. Springer, 2024a.

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*, pp. 14749–14759, 2024b.

Zile Wang, Zexiang Liu, Jaixing Li, Kaichen Huang, Baixin Xu, Fei Kang, Mengyin An, Peiyu Wang, Biao Jiang, Yichen Wei, et al. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. *arXiv preprint arXiv:2604.08995*, 2026.

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*, pp. 6902–6912, 2024.

Tianhe Wu, Ruibin Li, Lei Zhang, and Kede Ma. Diversity-preserved distribution matching distillation for fast visual synthesis. *arXiv preprint arXiv:2602.03139*, 2026.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024b.

Shenghai Yuan, Yuanyang Yin, Zongjian Li, Xinwei Huang, Xiao Yang, and Li Yuan. Helios: Real real-time long video generation model. *arXiv preprint arXiv:2603.04379*, 2026.

Yifan Zhan, Zhengqing Chen, Qingjie Wang, Zhuo He, Muyao Niu, Xiaoyang Guo, Wei Yin, Weiqiang Ren, Qian Zhang, and Yinqiang Zheng. Composing driving worlds through disentangled control for adversarial scenario generation. *arXiv preprint arXiv:2603.12864*, 2026.

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

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, pp. 10412–10420, 2025.

Kaiwen Zheng, Yuji Wang, Qianli Ma, Huayu Chen, Jintao Zhang, Yogesh Balaji, Jianfei Chen, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Large scale diffusion distillation via score-regularized continuous-time consistency. *arXiv preprint arXiv:2510.08431*, 2025.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, 2024. *URL https://github.com/hpcaitech/Open-Sora*, 2024.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*, 2026.

A PRELIMINARIES

Diffusion models. Diffusion models (Rombach et al., 2022; Esser et al., 2024; Blattmann et al., 2023; Brooks et al., 2024; Zheng et al., 2024) generate data from noise through a vector field v_Θ:

$$\mathrm{d}z_{(t)} = v_\Theta(z_{(t)}, t)\,\mathrm{d}t, \quad t \in [0, 1]. \tag{7}$$

The latent variables $z_{(t)}$ evolve from random Gaussian noise $\boldsymbol{\epsilon}$ at $t = 1$ to the data distribution at $t = 0$. The intermediate latent states are constructed as

$$z_{(t)} = \sigma_t\, z_{(0)} + (1 - \sigma_t)\,\boldsymbol{\epsilon}, \tag{8}$$

where σ_t is a predefined noise schedule. The flow matching objective (Albergo & Vanden-Eijnden, 2022; Lipman et al., 2022; Liu et al., 2022) is defined by the v-prediction loss as

$$\mathcal{L}_{CFM} = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(\mathbf{0},\mathbf{I})} \left\| v_\Theta(z_{(t)}, t) - (z_{(0)} - \boldsymbol{\epsilon}) \right\|_2^2. \tag{9}$$

Distribution matching distillation. Distribution matching distillation (DMD) (Yin et al., 2024b) trains a few-step student to match the teacher's output distribution with gradient

$$\nabla_\theta \mathcal{L}_{DMD} = \mathbb{E}_\tau \left[-\left(s_{\text{real}}(z_{(\tau)}) - s_{\text{fake}}(z_{(\tau)}) \right) \right] \frac{\partial G_\theta}{\partial \theta}, \tag{10}$$

where τ denotes the re-noising level.

B OVERALL FRAMEWORK OF HORIZONDRIVE

As illustrated in Fig. 5, the training pipeline of HorizonDrive centers on SRR for teacher stabilization. We first train a conditional video generation model, which provides the initialization for SRR. As our core contribution, SRR stabilizes this model to serve as the teacher for subsequent distillation. Finally, TRD transfers the stabilized teacher's generation capability into a short-chunk, few-step student for efficient inference.

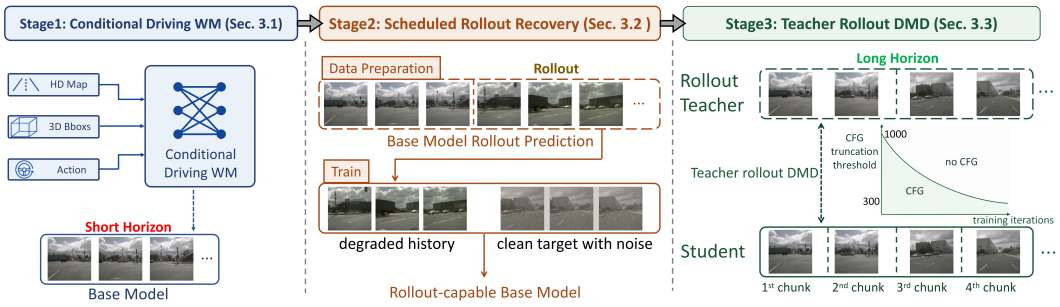

Figure 5: **Overview of the HorizonDrive framework.** Conditional model training is followed by SRR teacher stabilization and TRD distillation into a short-chunk, few-step student.

C ALGORITHMIC DETAILS OF SRR AND TRD

We provide the full pseudocode of the two training stages introduced in Secs. 3.2 and 3.3. Algorithm 1 details **Scheduled Rollout Recovery (SRR)**: starting from a rollout cache produced by the current model, each iteration samples a generation boundary s within the cache span, blends the predicted history with the ground-truth target around s via radius w, and supervises only the K-frame target chunk under flow matching. The cache is periodically refreshed with the latest θ and a (typically decaying) AR depth $N(k)$, so that the training distribution evolves alongside the model's own rollout behavior. Algorithm 2 details **Teacher Rollout Distillation (TRD)**: the student rolls out autoregressively with chunk size $K^{\mathcal{S}}$ (with gradients enabled at a randomly chosen denoising step per chunk), and every D student chunks the teacher provides fixed-window supervision along the student's rollout, conditioned on preceding student-generated history and driving controls

Algorithm 1 Scheduled Rollout Recovery (SRR) Training

Require: Ground-truth latent dataset $\mathcal{D}$, continuation model G_θ
Require: Schedules $N(k)$, $s(k)$, $w(k)$ for training step k; cache refresh period R
Require: AR context length T, chunk size K; CFM objective Eq. (9)
Initialize rollout cache $\mathcal{C}$ by running $N(0)$ AR steps per clip in $\mathcal{D}$ with G_θ ▷ no gradient
$k \leftarrow 0$
loop ▷ training
  $k \leftarrow k + 1$
  if $k \bmod R = 0$ **then**
    Refresh $\mathcal{C}$ by rolling out each clip again using the current θ and depth $N(k)$
  end if
  Sample paired $(\mathbf{Z}, \hat{\mathbf{Z}})$ from $\mathcal{C}$
  Sample boundary index $s \sim s(k)$ and blending radius $w \sim w(k)$
  Build $\{\bar{\mathbf{z}}_i\}_{i=s-T+1}^{s+K}$ via pred-to-GT blending with radius w around s
  Sample $t \sim \mathcal{U}[0, 1]$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Noise only the target chunk $\mathbf{z}^\star_{s+1:s+K}$ into $\mathbf{z}_{(t)}$; keep $\bar{\mathbf{z}}_{s-T+1:s}$ at $t{=}0$
  $\mathcal{L} \leftarrow \mathcal{L}_{\mathrm{CFM}}\big(G_\theta;\, \bar{\mathbf{z}}_{s-T+1:s},\, \mathbf{z}_{(t)},\, \mathbf{z}^\star_{s+1:s+K}\big)$ ▷ Eq. (9)
  Update θ using $\mathcal{L}$
end loop

D Implementation details

Backbone and VAE. HorizonDrive is built on Wan 2.1 1.3B (Wan et al., 2025) with full bidirectional attention, and adopts the disentangled driving-control modules described in Sec. 3.1. Since driving scenes involve fast ego-motion and rapidly changing fine details, we fine-tune the original VAE to reduce its temporal compression ratio from 4 to 1, preserving full temporal resolution for higher visual fidelity.

Conditional control modules. We elaborate on the disentangled control architecture summarized in Sec. 3.1. HD map and bounding box information are first rendered as structural conditions $z_{\boldsymbol{b}_f} \in \mathbb{R}^{c\times f\times h\times w}$, processed by a lightweight convolutional adapter, and reshaped into layout tokens $\boldsymbol{h}_{\boldsymbol{b}_f} \in \mathbb{R}^{f\times s\times d}$. The evolving feature representation at diffusion time t, denoted $\boldsymbol{h}_{(t)}$, is updated by adding the projected layout tokens following $\boldsymbol{h}_{(t)} \leftarrow \boldsymbol{h}_{(t)} + f_{\text{zero}}\big(\boldsymbol{h}_{\boldsymbol{b}_f}\big)$, where $f_{\text{zero}}(\cdot)$ is a zero-initialized projector. Action control is injected by computing the relative transformation $\Delta\mathbf{T}_i = \mathbf{T}_i^{-1}\mathbf{T}_{i+1}$ from a continuous trajectory of shape $F \times 4 \times 4$. We extract the residual signal $\boldsymbol{a} = (\Delta x, \Delta y, \Delta\text{yaw}) \in \mathbb{R}^{F\times 3}$ and inject it via AdaLN-style gating (Peebles & Xie, 2023) as $f_{\text{zero}}(\phi(\boldsymbol{a})) \in \mathbb{R}^{f\times 6\times d}$, where $\phi(\cdot)$ is the sinusoidal frequency embedding. The resulting six channels are split into two groups, representing the pre-normalization shift/scale and post-layer residual gate for self-attention and feed-forward sublayers, respectively.

Training stages. We train the base conditional driving world model $\mathcal{G}_0$ for 40K steps on a 100-hour proprietary driving dataset and nuScenes (Caesar et al., 2020). We then apply SRR for 10K steps on nuScenes to obtain $\mathcal{G}_{\text{roll}}$. For TRD distillation, the teacher and student share a context window of $T{=}11$ frames; the student generates $K^{\mathcal{S}}{=}10$ new frames per chunk with 4 denoising steps and rolls out $N{=}20$ autoregressive steps during training, while the teacher generates $K^{\mathcal{T}}{=}40$ frames per chunk and slides a $(T + K^{\mathcal{T}}){=}51$-frame supervision window across student's rollout to cover each student chunk, providing memory-bounded supervision over the long-horizon trajectory. Full optimization hyperparameters and stage-specific schedules of all three training stages are listed in Tabs. 9 and 10.

Datasets and evaluation horizon. For nuScenes (Caesar et al., 2020), we use 700 multi-view videos of $\sim$20 seconds for training and 150 for validation; the per-clip length of $\sim$20 s is the dataset upper bound, which determines the horizon of our quantitative evaluation. Longer rollouts of up to one minute on a self-collected dataset are demonstrated qualitatively in Sec. M.

Inference efficiency. With our temporally-uncompressed VAE and a 4-step student denoiser, each AR rollout step generates one $K^{\mathcal{S}}{=}10$-frame chunk on a single NVIDIA 5090 GPU. The per-

Algorithm 2 Teacher Rollout Distillation (TRD) Training

Require: Frozen teacher G^{T}, student G^{S}_{ϕ}, fake critic G^{F} (same init as student)
Require: T, teacher chunk $K^{\mathcal{T}}$, student chunk $K^{\mathcal{S}}$; student rollout depth N
Require: Student denoising steps $M^{\mathcal{S}}$, teacher denoising steps $M^{\mathcal{T}}$; renoise schedule for τ
Require: DMD interval D ▷ perform DMD every D student chunks
loop ▷ training
  Sample clip latents $\mathbf{Z}$ and controls $\mathbf{c}$
  $H^S \leftarrow \mathbf{Z}_{1:T}$; initialize list $\hat{\mathcal{Z}}^S \leftarrow \emptyset$
  for $n = 1$ **to** N **do**
    // Student AR rollout
    $T_{\mathrm{grad}} \sim \mathrm{Uniform}\{1, \ldots, M^{\mathcal{S}}\}$
    Sample $\mathbf{x}_{t_{M^{\mathcal{S}}}} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ over $K^{\mathcal{S}}$ frames
    for $j = M^{\mathcal{S}}, M^{\mathcal{S}}-1, \ldots, 1$ **do**
      if $j = T_{\mathrm{grad}}$ **then**
        Enable gradient
      else
        Disable gradient
      end if
      $\hat{\mathbf{x}}_0 \leftarrow G^{\mathrm{S}}(\mathbf{x}_{t_j};\, t_j,\, H^S,\, \mathbf{c}_{\tau_n+1:\tau_n+K^{\mathcal{S}}})$
      if $j > 1$ **then**
        Sample $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$; $\mathbf{x}_{t_{j-1}} \leftarrow \Psi(\hat{\mathbf{x}}_0, \boldsymbol{\epsilon}, t_{j-1})$
      end if
    end for
    Append $\hat{\mathbf{x}}_0$ to $\hat{\mathcal{Z}}^S$; slide H^S forward by $K^{\mathcal{S}}$ with $\mathrm{stopgrad}(\hat{\mathbf{x}}_0)$
    if $n \bmod D = 0$ **then** ▷ DMD every D chunks
      // Teacher AR rollout (no gradient)
      Roll out G^{T} with window $(T, K^{\mathcal{T}})$ covering chunks $n-D+1$ to n
      Concatenate student outputs $\hat{\mathcal{Z}}^S_{n-D+1:n}$; align teacher supervision
      Sample τ; $\mathbf{z}_{(\tau)} \leftarrow \mathrm{Renoise}(\hat{\mathbf{z}}^S_{n-D+1:n}, \tau)$
      $s^{\mathrm{c}}_{\mathrm{real}} \leftarrow s^{\mathrm{real}}_{\mathrm{cond}}\big(\mathbf{z}_{(\tau)}\big)$ ▷ from G^{T}
      $s^{\mathrm{c}}_{\mathrm{fake}} \leftarrow s^{\mathrm{fake}}_{\mathrm{cond}}\big(\mathbf{z}_{(\tau)}\big), \quad s^{\mathrm{u}}_{\mathrm{real}} \leftarrow s^{\mathrm{real}}_{\mathrm{uncond}}\big(\mathbf{z}_{(\tau)}\big)$ ▷ from G^{F}
      $\mathbf{d} \leftarrow \big(s^{\mathrm{c}}_{\mathrm{real}} - s^{\mathrm{c}}_{\mathrm{fake}}\big) + \mathrm{trunc_cfg}\big(\tau,\, s^{\mathrm{c}}_{\mathrm{real}},\, s^{\mathrm{u}}_{\mathrm{real}}\big)$
      Accumulate $-\mathbf{d} \cdot \partial G^{\mathrm{S}}_{\phi} / \partial \phi$
      Update ϕ; update G^{F} via distribution matching on $\hat{\mathcal{Z}}^S_{n-D+1:n}$
    end if
  end for
end loop

chunk wall-clock latency is **1.8 s** at 256×512 resolution and **5.8 s** at 384×768, yielding effective generation rates of $\sim$5.6 FPS and $\sim$1.7 FPS, respectively. Because the rolling-window rollout (Sec. 3.1) bounds per-step compute by $T+K^{\mathcal{S}}$ regardless of total horizon, this per-chunk latency is sustained throughout the rollout, enabling stable streaming generation up to minute-scale horizons (Sec. M). With TensorRT compilation and FP8 quantization, the 256×512 model reaches 10.8 FPS on the same single RTX 5090.

SRR cache cost. SRR constructs its rollout cache without gradient computation and refreshes it every 2,000 optimization steps while rollout depth decreases from 10 to 4. Each refresh processes 1,024 clips and takes 0.77 hours on 64 RTX 5090 GPUs (49.1 GPU-hours). Across the 36.34-hour SRR stage, cache construction accounts for 10.6% of wall-clock training time and introduces no inference-time overhead.

E Evaluation metrics

Visual quality. We report Fréchet Inception Distance (FID) and Fréchet Video Distance (FVD) computed over the full rollout sequence, as well as VBench (Huang et al., 2024) as an overall video-quality measure.

Spatio-temporal consistency. These metrics are applicable only to methods that accept driving control signals. We recover per-frame camera poses from both generated and ground-truth videos with VGGT (Wang et al., 2025). **ARE** (Average Rotation Error) measures the mean geodesic distance between predicted and GT rotation matrices across frames, reflecting heading accuracy. **DTW** (Dynamic Time Warping) (Keogh & Pazzani, 2000) aligns predicted and GT ego-motion trajectories via non-rigid time warping and computes the cumulative Euclidean distance under the optimal alignment, capturing path-shape fidelity even under temporal misalignment.

F BASELINE EVALUATION PROTOCOLS

Group (i): long-horizon interactive world model frameworks. These methods are designed for general open-domain or interactive world simulation and do not natively accept our driving control signals (actions, HD maps, bounding boxes). For each method, we use the publicly released checkpoint, condition it only on the initial frame(s) following the original interface, and let it autoregressively roll out to match our nuScenes evaluation horizon. We report visual-quality metrics (FID, FVD, VBench) only; ego-motion consistency (ARE/DTW) is marked "N/A" since no driving control is provided.

Group (ii): long-horizon streaming video generation methods. Self-Forcing (Huang et al., 2025), Self-Forcing++ (Cui et al., 2025), and LongLive (Yang et al., 2025) are general streaming long-video recipes. To ensure fair comparison, for each baseline we (a) attach our disentangled driving control modules (Sec. 3.1) to the same Wan 2.1 1.3B backbone, (b) initialize from our base model $\mathcal{G}_0$, and (c) retrain using each method's original training recipe (loss, scheduler, rollout configuration, hyperparameters) on our nuScenes training split. Architecture, base model, training data, and driving conditioning are therefore shared with HorizonDrive, so the only remaining variable is the long-horizon training framework itself. All metrics (FID, FVD, VBench, ARE, DTW) are reported under the same protocol as HorizonDrive.

Group (iii): driving video generation models. These are domain-specific driving generators that accept driving control signals but produce only a fixed-length single-pass clip (8–241 frames depending on the method). Since each method's evaluation setting differs (clip length, validation split, conditioning, evaluation protocol), we directly take FID/FVD numbers from the original papers and report them in two groups—*short-video* (8–25 frames) and *long-video* (241 frames)—rather than re-running them under a unified protocol. Results in this table are intended as a domain reference rather than a strict head-to-head ranking.

G ADDITIONAL QUANTITATIVE RESULTS

We provide detailed results for the driving-generator comparison (Tab. 5) and TRD ablations on the training rollout horizon and CFG schedule (Tabs. 6 and 7) discussed in the main paper.

H COMPARISON WITH SVI

We reproduce SVI (Li et al., 2026) under the same teacher architecture and initialization, training data, rollout protocol, and training budget as SRR. As shown in Tab. 8, SRR achieves lower FID and FVD, supporting the effectiveness of clip-specific contiguous rollout histories combined with the proposed recovery curricula.

Table 8: **Comparison with SVI on nuScenes val.**

Method	FID↓	FVD↓	ARE↓	DTW↓
SVI (Li et al., 2026)	25.39	173.98	2.61	3.31
SRR (Ours)	**13.73**	**94.84**	**1.95**	**2.99**

Table 5: **Comparison with driving video generation methods on nuScenes val.** All HorizonDrive results use the final four-step TRD student. Results from other methods follow their original protocols and are provided for reference rather than as a strict ranking; HorizonDrive attains the best listed short-video FVD and competitive FID.

Method	Unroll N	Frames	Conditions	FID ↓	FVD ↓
Short-Video Evaluation					
DriveDreamer (Wang et al., 2024a)	1	8	T+M+B	14.90	340.80
Panacea (Wen et al., 2024)	1	8	T+M+B	16.90	139.00
DreamForge (Mei et al., 2024)	1	16	T+M+B	14.61	103.61
Vista (Gao et al., 2024b)	1	25	T+A	6.90	89.40
HorizonDrive (Ours)	1	21	M+B+A	12.54	84.53
Long-Video Evaluation					
MagicDrive-V2 (Gao et al., 2025a)	1	241	T+M+B+A	20.91	94.84
HorizonDrive (Ours)	20	211	M+B+A	13.82	92.99

Conditions: T = text, M = HD map, B = bounding boxes, A = ego action. HorizonDrive uses M+B+A and is not text-conditioned.

Table 6: **Effect of unroll length on TRD on nuScenes val.** Both student and teacher use SRR initialization, and CFG is applied with the delayed schedule. Bold and underlined values indicate the best and second-best results, respectively.

Unroll N	FID↓	FVD↓	ARE↓	DTW↓
1	21.15	<u>135.35</u>	3.39	5.42
4	<u>18.19</u>	139.28	<u>3.03</u>	<u>3.66</u>
20	**13.82**	**92.99**	**2.60**	**3.27**

I TRAINING CONFIGURATION

Section I and Table 10 list the training configuration of all three stages. The three stages share the same backbone, context window length T, and disentangled control modules; they differ in the rollout setup (chunk size, AR rollout depth) and stage-specific schedules. Stage 1 trains the base conditional driving world model $\mathcal{G}_0$ on short clips of length $T+K$ with ground-truth conditioning latents. SRR keeps the same (T, K), but additionally maintains a per-clip rollout cache that is refreshed every R optimizer steps with the current θ and a decaying AR rollout depth $N(k)$; within each clip the boundary index s is sampled in $[T,\ T+N(k)K]$ and the blending radius w is gradually enlarged. TRD distills the SRR-initialized teacher (chunk $K^{\mathcal{T}}$) into a few-step student (chunk $K^{\mathcal{S}}$, with $M^{\mathcal{S}}$ denoising steps), updating the chunk-wise distribution matching loss every D student chunks; CFG augmentation is applied only at low noise levels ($\tau \leq \tau_{\text{th}}$) with weight α.

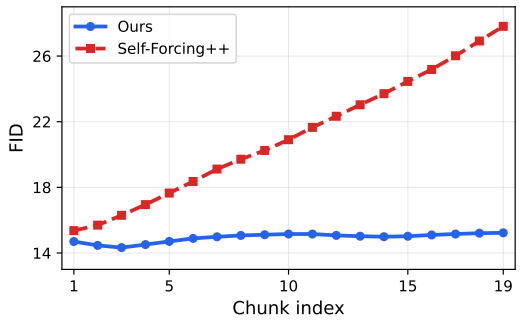

Figure 6: Long-horizon generation quality comparison.

Table 7: **Effect of CFG scheduling on TRD on nuScenes val.** Both student and teacher use SRR initialization, with the unroll length fixed at $N = 20$. Bold and underlined values indicate the best and second-best results, respectively.

CFG schedule	FID↓	FVD↓	ARE↓	DTW↓
None	<u>14.59</u>	<u>110.81</u>	2.77	3.28
Full	20.84	184.06	3.86	3.96
Early	14.70	111.99	<u>2.64</u>	**3.13**
Delayed	**13.82**	**92.99**	**2.60**	<u>3.27</u>

Table 9: **Training configuration of the teacher stages of HorizonDrive.** "–" marks entries not applicable to the corresponding stage.

Hyperparameter	Base ($\mathcal{G}_0$)	SRR ($\mathcal{G}_{\text{roll}}$)
Optimizer	AdamW	AdamW
Learning rate	1e-5	1e-5
Weight decay	1e-2	1e-5
Global batch size	96	64
Mixed precision	bf16	bf16
Training steps	40K (proprietary) + 10K (nuScenes)	10K (nuScenes)
GPU Usage	96 NVIDIA 5090	64 NVIDIA 5090
Context window length T	11	11
Chunk size K	10, 40	10, 40
Resolution	[256, 512], [384, 768]	[256, 512], [384, 768]
AR rollout depth N	–	$N(k)$ schedule
Cache refresh period R (optimizer steps)	–	2000
$N(k)$ schedule (start → end)	–	$10 \rightarrow 4$ (steps 0-8000)
Blending radius w schedule	–	$0 \rightarrow 8$ (steps 0-8000)

J Error accumulation analysis

Fig. 6 reports FID along autoregressive rollout. Our method maintains stable image quality across all 19 chunks, whereas Self-Forcing++ degrades monotonically. Aggregate full-sequence metrics can hide late-horizon failure: for the teacher-forcing base model, interval-wise FID over chunks 1–4, 5–8, 9–12, 13–16, and 17–20 rises from 15.70 to 32.41, 51.32, 70.26, and 89.71, respectively, even though the cumulative 20-chunk FID is only 31.13. This exposes the severe late-rollout distribution shift that motivates SRR.

K Additional qualitative results on nuScenes

We provide additional rollouts on nuScenes val to complement the quantitative comparison in the main paper. Figures 7 to 9 compare HorizonDrive against the long-horizon streaming baselines (Self-Forcing, Self-Forcing++, LongLive) under our shared base model and driving control conditions. Figures 10 and 11 compare against representative driving world models that are not autoregressive (e.g., Matrix-Game3, Helios, LingBot-World), where each method uses its own native generation setting. Across both groups, HorizonDrive preserves road structure, agent layout, and spatio-temporal consistency throughout the rollout, while baselines progressively drift in geometry, color, or scene composition.

L Results on the self-collected dataset

To evaluate generalization beyond nuScenes, we also benchmark on our internal self-collected end-to-end (e2e) driving dataset, which features higher ego speeds and more diverse scenarios than nuScenes. Table 11 reports the quantitative comparison against the long-horizon streaming baselines under the same protocol as the main paper (retrained on our base model and the e2e training split, evaluated on a held-out split). Figures 12 and 13 provide qualitative rollouts. HorizonDrive again outperforms

Table 10: **Training configuration of the distillation stage (TRD).**

Hyperparameter	TRD
Optimization	
Optimizer	AdamW
Student learning rate	2e-6
Critic learning rate	1e-5
Weight decay	0.1
Batch size	32
Mixed precision	bf16
Training steps	200
GPU Usage	32 NVIDIA 5090
Context window length T	11
Chunk size K	10 (student) / 40 (teacher)
AR rollout depth N	20 (student)
Denoising steps	4
DMD update interval D (student chunks)	4
CFG threshold τ_{th} schedule	1000 (steps 0–100), decay $\rightarrow$ 0 (step 400)
CFG scale α	6

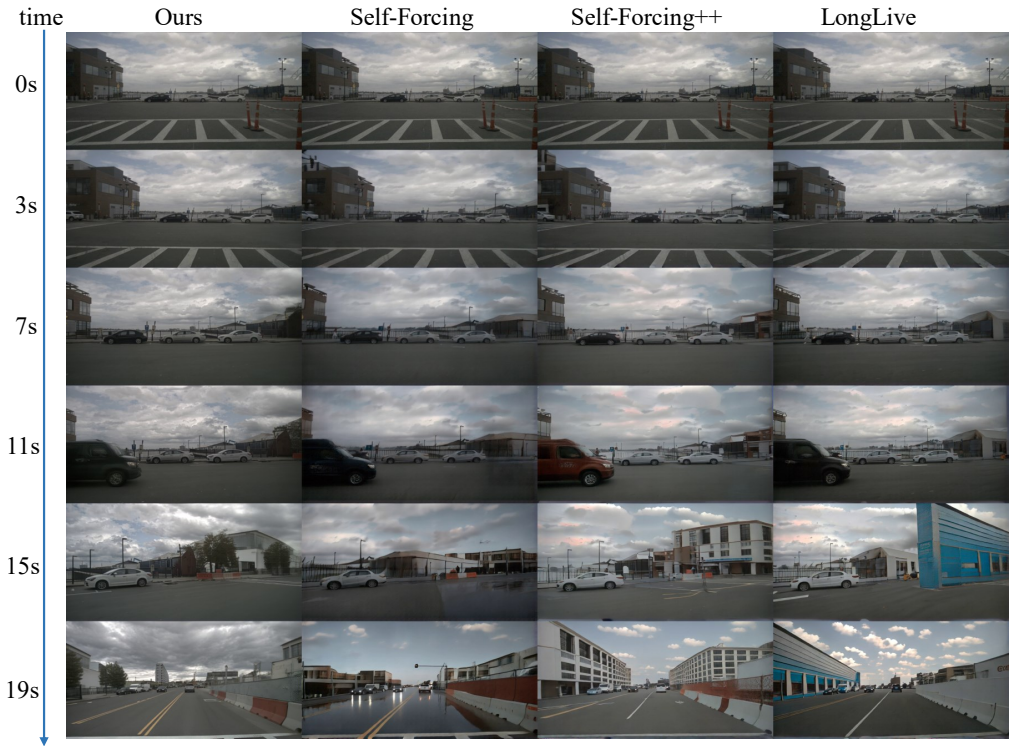

Figure 7: **Qualitative comparison with long-horizon streaming baselines on nuScenes val (scene 1).** From left to right: HorizonDrive, Self-Forcing, Self-Forcing++, LongLive

all baselines on visual quality (FID/FVD), VBench dimensions, and spatio-temporal consistency, indicating that the rollout-capable base model and TRD distillation transfer to a substantially different driving distribution.

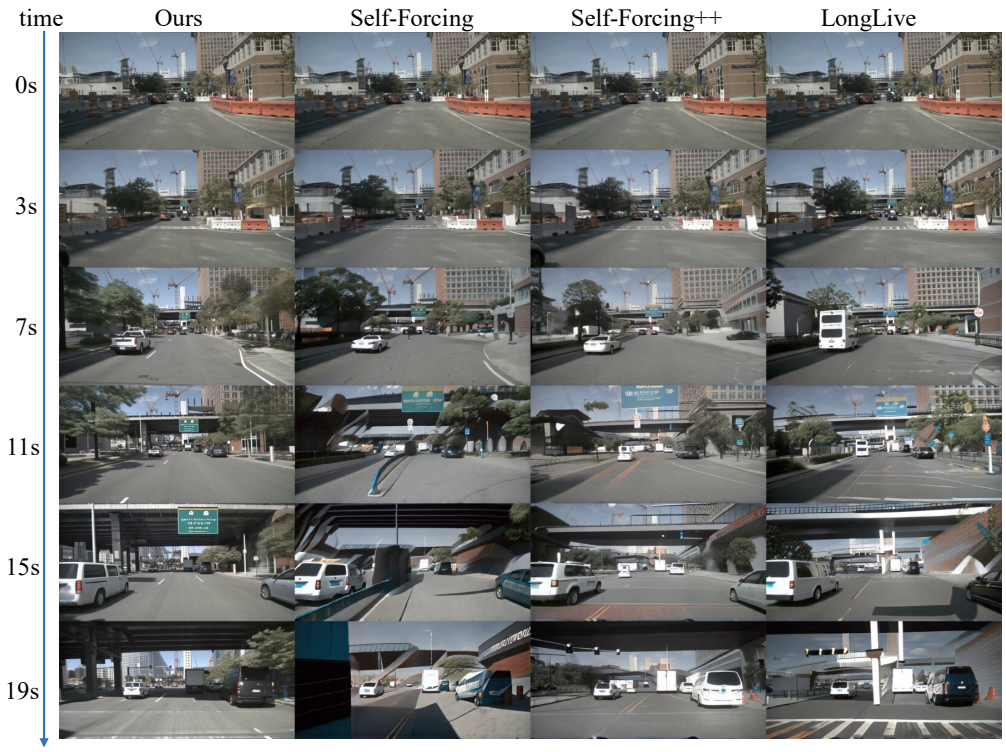

Figure 8: **Qualitative comparison with long-horizon streaming baselines on nuScenes val (scene 2).**

Table 11: **Comparison with long-horizon streaming video generation baselines on the self-collected (e2e) dataset.** All baselines share our base model and driving control modules, and are retrained on the e2e training split following each method's original recipe. Best results are in **bold**; second best are underlined.

Method	Visual		VBench			ST Cons.	
	FID↓	FVD↓	Qual.↑	Mot.↑	Img.↑	ARE↓	DTW↓
Long-horizon streaming video generation methods (retrained on our base model and e2e data)							
Self-Forcing (Huang et al., 2025)	58.23	561.11	76.68	94.48	63.18	5.43	14.13
Self-Forcing++ (Cui et al., 2025)	66.93	534.36	74.54	92.70	59.12	7.32	18.40
LongLive (Yang et al., 2025)	28.39	374.94	78.18	94.57	62.53	4.05	8.11
HorizonDrive (Ours)	**12.01**	**117.27**	**80.12**	**95.22**	**67.65**	**3.67**	**5.29**

M MINUTE-SCALE AUTOREGRESSIVE GENERATION

HorizonDrive supports minute-scale AR generation under bounded per-step memory and compute, which is otherwise infeasible for single-pass video diffusion baselines whose memory grows quadratically with sequence length. At inference, the sliding-window rollout (Sec. 3.1) keeps each step's compute bounded by $T+K^{\mathcal{S}}$, so the model can be rolled out indefinitely without re-warming the cache. Figure 14 shows a continuous rollout of approximately one minute on the self-collected dataset (10 FPS, $K^{\mathcal{S}}=10$, $T=11$), where the model maintains coherent road geometry, lane structure, and traffic-agent behavior throughout the rollout, demonstrating practical viability for closed-loop driving simulation.

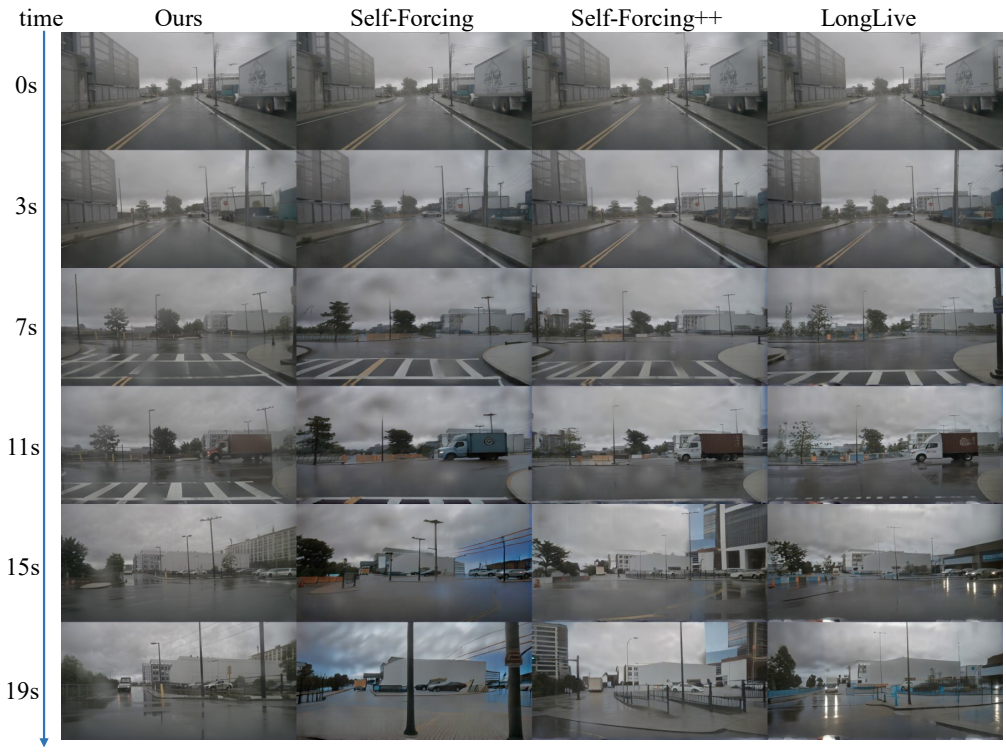

Figure 9: **Qualitative comparison with long-horizon streaming baselines on nuScenes val (scene 3).** Same layout as Fig. 7.

N CLOSED-LOOP DRIVING SIMULATION

We further evaluate HorizonDrive in a closed-loop driving simulation setup, where a planner and the world model interact step by step: at each AR step, the planner consumes the latest generated frame and outputs an ego trajectory, which is re-encoded (together with the corresponding HD map and bounding-box layouts) as the next-step action condition fed back into HorizonDrive, and the loop repeats indefinitely under the same bounded per-step memory budget as standard rollout (Sec. M). No ground-truth ego trajectory is used during simulation; instead, subsequent ego action conditions are derived from the planner's predictions. As shown in Fig. 15, despite the compounding of rollout error from the world model and prediction error from the planner, HorizonDrive maintains coherent road geometry, lane topology, and stable agent behavior throughout the rollout, demonstrating practical viability for closed-loop policy evaluation under self-generated ego trajectories.

Frozen-planner evaluation without feedback. As a complementary assessment of downstream planning utility, we evaluate a frozen Epona planner on the nuScenes validation set using generated observations from self-conditioned AR rollouts. The world model conditions on its own generated history, but the planner's predicted trajectories are not fed back into subsequent generation. This protocol therefore evaluates planner responses to generated observations, rather than closed-loop interaction as in Tab. 13.

As shown in Tab. 12, HorizonDrive performs best among the generation methods across all four metrics, with results closer to the Expert GT reference. Compared with LongLive, it improves PDMS from 85.76 to 87.17 and reduces trajectory L2 from 0.8645 m to 0.8033 m. These results provide supporting evidence that HorizonDrive produces observations more useful for downstream planning, motivating its use as a closed-loop simulator.

We also conduct a 10-step closed-loop evaluation using Epona as the planner. As shown in Tab. 13, HorizonDrive improves PDMS from 82.64 to 85.57 compared with Self-Forcing++.

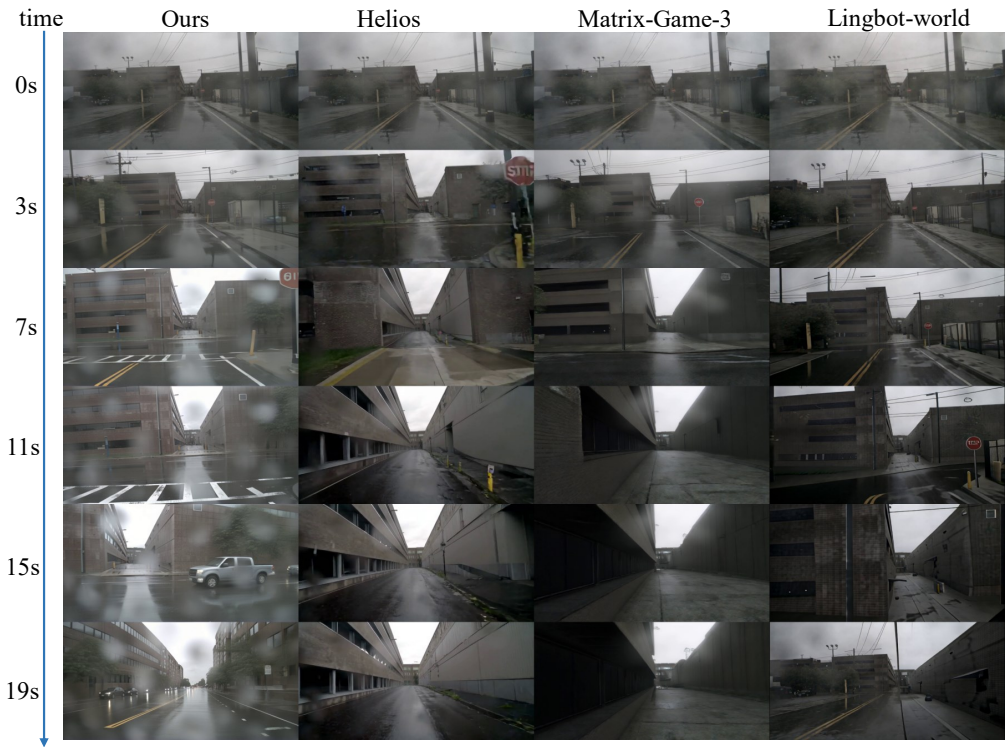

Figure 10: **Qualitative comparison with driving world models on nuScenes val (scene 1).** From left to right: HorizonDrive, Helios, Matrix-Game3, LingBot-World

Table 12: **Frozen-planner evaluation without feedback on nuScenes val.** Epona consumes observations from self-conditioned AR rollouts without feeding its predictions back to the world model. EP denotes ego progress. Expert GT is included as a reference; bold marks the best result among the four generation methods.

Method	Frame collision↓	EP↑	PDMS↑	Trajectory L2 (m)↓
Self-Forcing	0.578%	0.8533	83.32	1.1908
Self-Forcing++	0.667%	0.8579	84.65	1.0048
LongLive	0.578%	0.8732	85.76	0.8645
HorizonDrive	**0.533%**	**0.8834**	**87.17**	**0.8033**
Expert GT	0.311%	0.9200	89.95	0.0000

We further visualize multiple rounds of interaction between HorizonDrive and the planner in a real-world driving scenario in Fig. 15. The left panel shows how the counterfactual trajectory (purple) deviates from the original trajectory (yellow), while the right panel illustrates how HorizonDrive generates high-quality observations consistent with the updated trajectory over successive interaction steps.

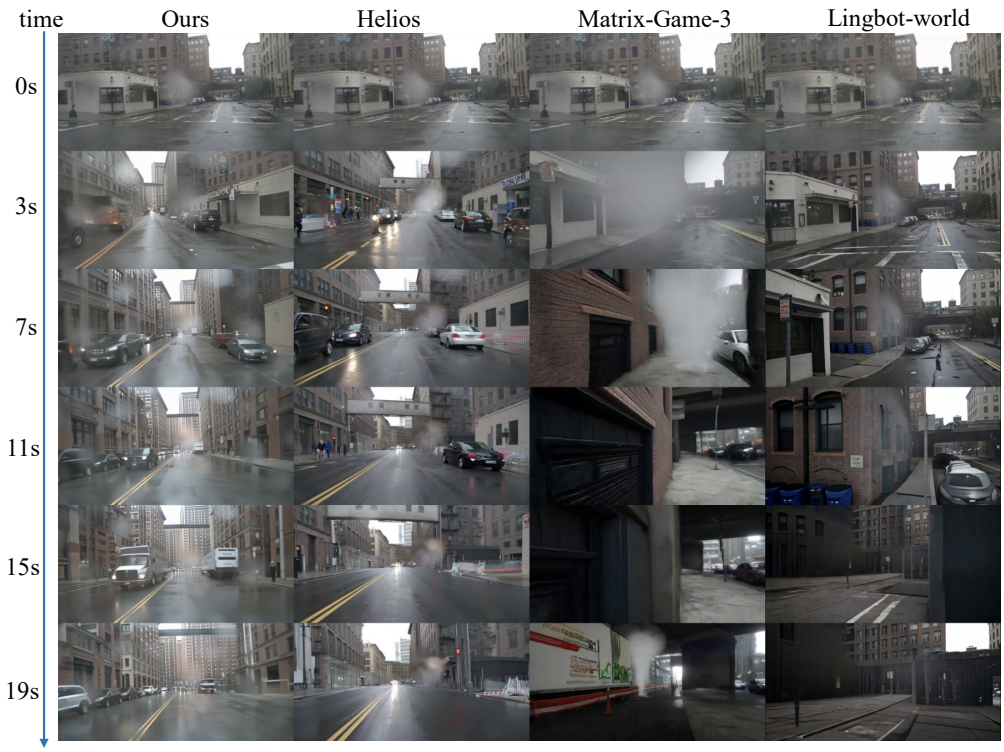

Figure 11: **Qualitative comparison with driving world models on nuScenes val (scene 2).**

Table 13: **Closed-loop driving evaluation on nuScenes val.** Each method is evaluated over 10 planner-in-the-loop steps, with predicted Epona trajectories fed back as action conditions. HorizonDrive improves PDMS over Self-Forcing++.

Method	Collision rate↓	Ego progress↑	Comfort↑	PDMS↑
Self-Forcing++	**4.58%**	0.8663	0.71	82.64
HorizonDrive	**4.58%**	**0.8810**	**0.86**	**85.57**

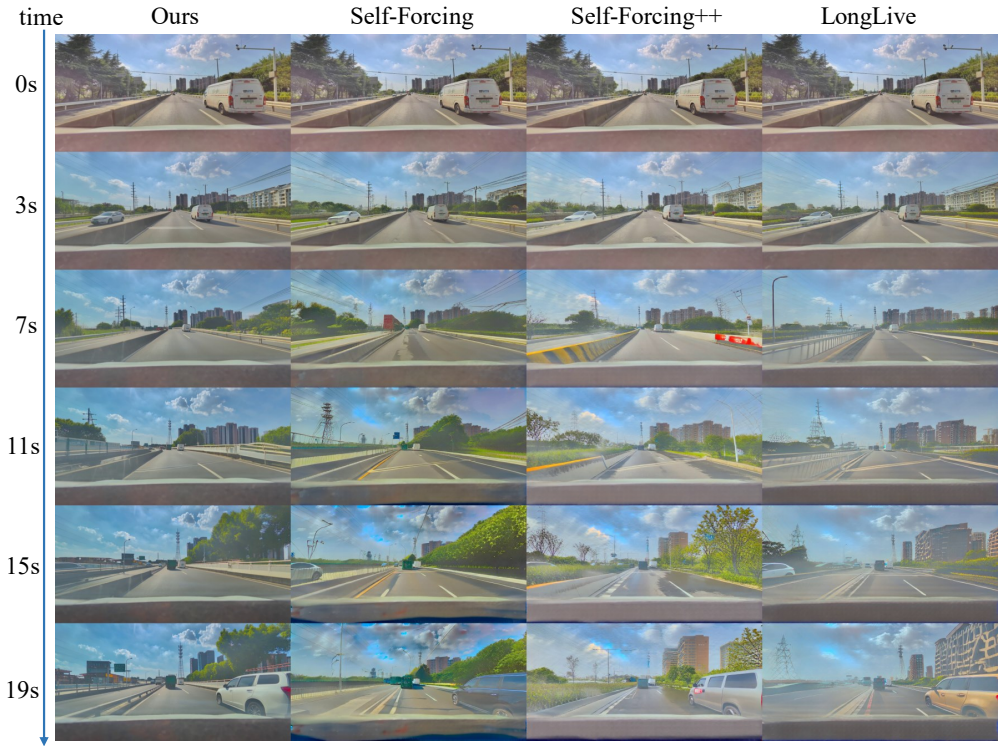

Figure 12: **Qualitative comparison with long-horizon streaming baselines on the self-collected (e2e) dataset (scene 1).** From left to right: HorizonDrive, Self-Forcing, Self-Forcing++, LongLive

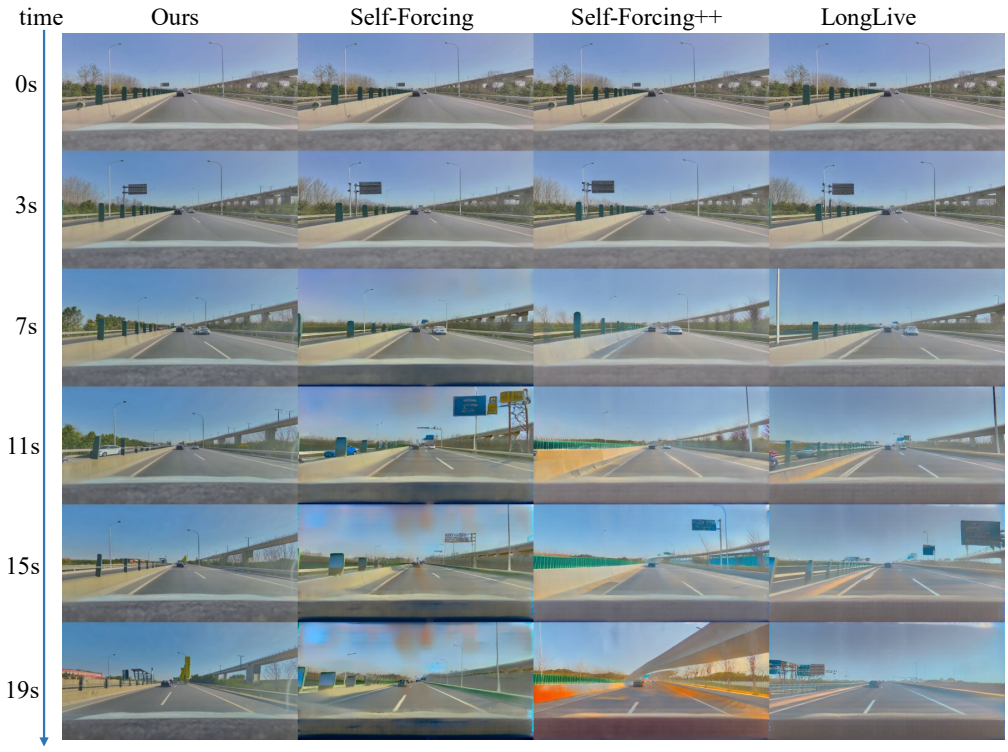

Figure 13: **Qualitative rollouts on the self-collected (e2e) dataset (scene 2).**



Figure 14: **Minute-scale autoregressive generation on the self-collected (e2e) dataset.**

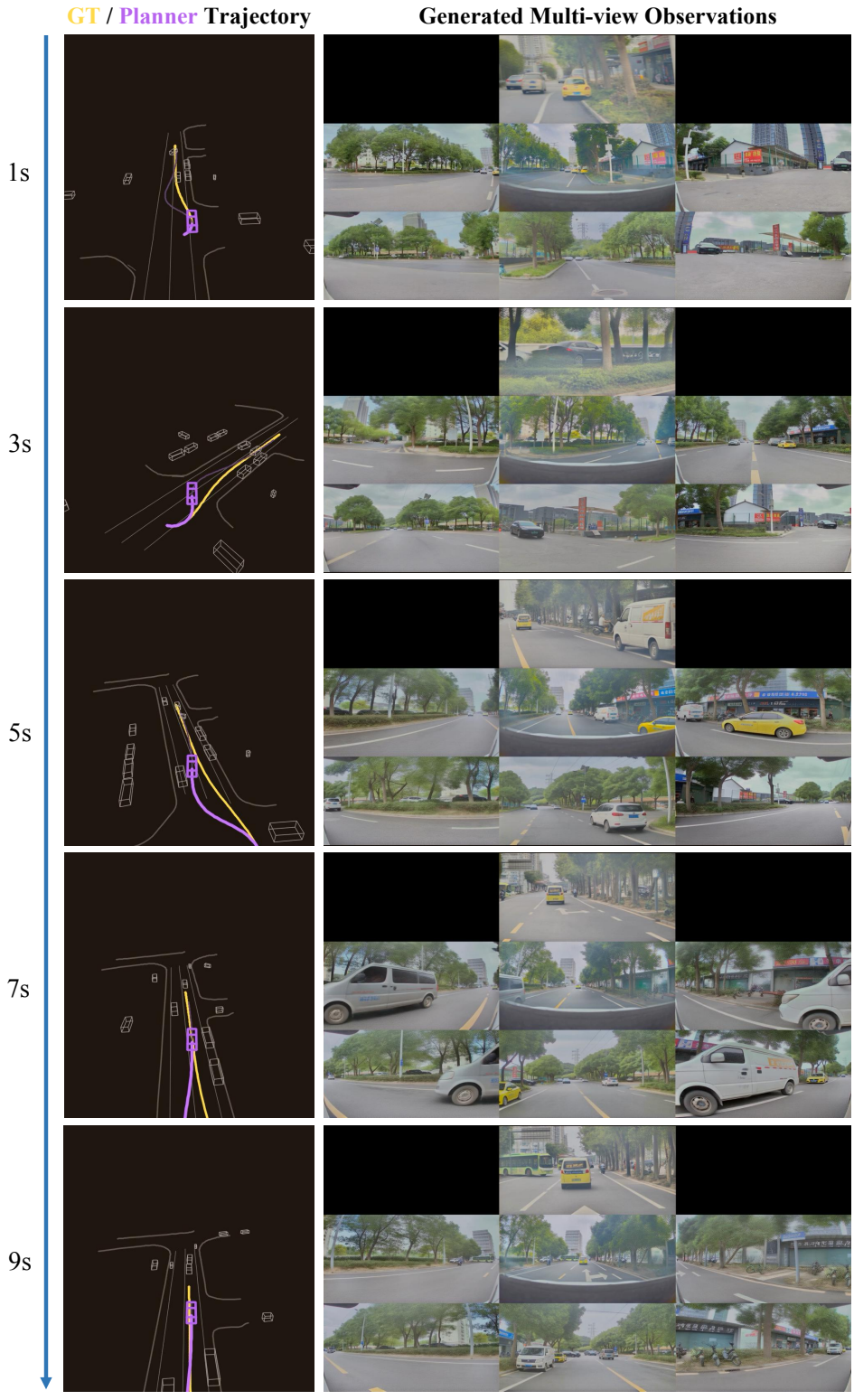

Figure 15: **Closed-loop driving simulation.** A planner consumes the latest generated frame at each step and outputs an ego trajectory, which HorizonDrive uses as the next-step action condition. Despite the planner-and-world-model loop being driven entirely by self-generated signals, HorizonDrive maintains coherent scene structure and stable agent behavior over long horizons.