DRIVEUNITOK: A UNIFIED MULTIMODAL TOKENIZER FOR DRIVING WORLD-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

World models are emerging as a fundamental component of policy learning for real-world agents, enabling agents to predict future world states and plan actions within a learned representation of the dynamic environment. Existing approaches, however, often model appearance, metric geometry represented by projected LiDAR depth, and ego motion either partially or independently, overlooking the fact that these modalities are different projections of a shared underlying world state. This separation can lead to information loss and cross-modal inconsistency, ultimately limiting downstream performance. Motivated by this unified view of the world, we propose **DriveUniTok**, a unified multimodal tokenizer for driving world-action models (WAMs) that encodes these modalities in a single decodable continuous latent. We freeze DriveUniTok and adapt a pretrained diffusion model, **DriveUniWAM**, to predict future unified latents; the tokenizer then decodes them into future appearance, projected LiDAR depth, and ego trajectories. Experiments on NAVSIM demonstrate that our unified design achieves better future prediction and planning performance at lower downstream computational cost than separate-stream alternatives. With reinforcement learning post-training, DriveUniWAM achieves a PDMS of 91.54, reaching state-of-the-art performance among the evaluated WAM-based planners. The anonymous code is available at `https://anonymous.4open.science/r/DriveUniTok-B358/`.

1 INTRODUCTION

Driving world models initially served as scene simulators and data engines, forecasting future observations for data augmentation and policy training. More recently, generative models of future observations have been integrated into policy backbones that jointly model scene evolution and ego trajectories (Zhang et al., 2025b; Zhao et al., 2025; Xia et al., 2026; Shi et al., 2026). In this setting, the latent representation is no longer merely an interface for visual reconstruction; it is the space in which the model captures the information required for both future prediction and planning.

Existing world-action models, however, expose asymmetric representation interfaces. Most inherit a pretrained RGB video autoencoder whose latent primarily captures appearance, while geometry is introduced through auxiliary supervision and ego pose remains outside the latent, where it is handled by a separate conditioning stream or action head (Team Wan et al., 2025; Kong et al., 2024; Agarwal et al., 2025). Driving tokenizers fuse camera and LiDAR observations into scene-centric bird's-eye-view latents, but still leave ego pose outside the scene representation (Zhang et al., 2024; Chen & Zhao, 2025; Zhuo et al., 2026; Liao et al., 2025b). Cross-modal and latent-action tokenizers incorporate actions alongside observations, yet commonly retain modality-specific streams or align them through a designated anchor or intermediary representation (Lu et al., 2026; Bachmann et al., 2024; Chen et al., 2025a; Fan et al., 2025; Chen et al., 2026). Figure 1 compares these representations according to whether appearance, metric geometry, and ego pose reside in the same decodable latent.

This asymmetry forces the downstream predictor to learn cross-modal correspondence together with temporal evolution, which can make future-scene and ego-trajectory predictions inconsistent. Therefore, a central question arises: Should visual appearance, metric geometry, and ego-motion be represented within a unified tokenization framework? DriveUniTok addresses this question by encoding RGB appearance, projected LiDAR geometry, and clip-relative ego pose in one jointly decodable latent.

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.

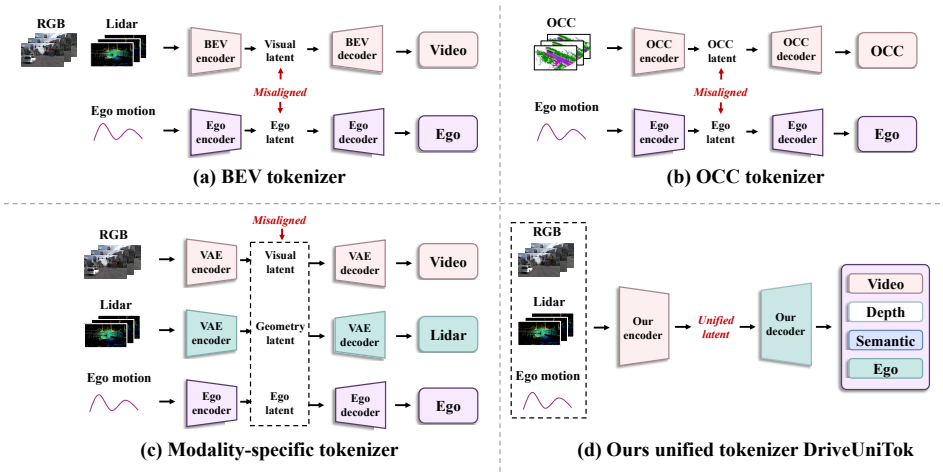

Figure 1: Comparison of latent representations for world-action modeling. (a) BEV tokenizers fuse RGB and LiDAR into a visual latent, while (b) Occupancy tokenizers represent the scene with an occupancy latent; both keep ego motion separate. (c) Modality-specific tokenizers encode RGB, LiDAR, and ego motion into separate streams. (d) DriveUniTok jointly encodes RGB, LiDAR-derived geometry, and ego motion into one unified latent.

To construct such a representation, we propose **DriveUniTok**, a query-based multimodal tokenizer. Shared learnable queries aggregate appearance features from RGB, metric geometry features from projected LiDAR depth, and clip-relative ego-pose features into a common sequence of per-frame multimodal tokens. A causal temporal compressor then maps this sequence into a compact continuous unified latent. All three modalities are reconstructed from the shared bottleneck through modality-specific readouts, without using a privileged anchor modality. In addition, a training-only semantic decoder is supervised by representations from a vision foundation model, encouraging the unified latent to preserve high-level scene semantics alongside metric geometry and pose.

To evaluate whether this representation supports joint forecasting and planning, we freeze DriveUniTok and adapt a pretrained diffusion model into **DriveUniWAM**, using the same downstream backbone and training protocol for unified and separate representations. The Wan-initialized flow-matching transformer predicts future unified latents under block-causal attention, conditioned on DriveUniTok-encoded historical observations and clip-relative ego poses together with a navigation signal. Future RGB, depth, and ego trajectories are decoded from the same predicted latent evolution rather than generated through independent observation and action pathways. Because supervised flow matching fits recorded futures but does not directly optimize the NAVSIM planning metric, we additionally post-train DriveUniWAM with GRPO against the PDM reward while regularizing it toward the supervised reference model.

Our key contributions are as follows:

- We propose DriveUniTok, a query-based tokenizer that encodes RGB appearance, projected LiDAR geometry, and clip-relative ego pose in a unified continuous latent.
- We develop DriveUniWAM, a diffusion-based WAM that forecasts this latent for joint future-scene prediction and ego-trajectory planning.
- We show on NAVSIM that the unified design achieves stronger prediction and planning at lower downstream computational cost than separate-stream controls, with DriveUniWAM reaching state-of-the-art performance at 91.54 PDMS after reinforcement learning post-training.

2 Related Work

2.1 General and Driving Tokenizers

For visual tokenizers, discrete codebooks are commonly paired with autoregressive or masked token prediction (van den Oord et al., 2017; Yu et al., 2024), whereas continuous latents are often used with diffusion and flow-matching models. Large video VAEs provide high-fidelity latent spaces for RGB generation (Team Wan et al., 2025; Kong et al., 2024; Yang et al., 2025; Agarwal et al., 2025), and quantization and model design materially affect representation quality (Luo et al., 2024; Yu et al., 2024; Wang et al., 2024a; Tang et al., 2024; Wang et al., 2025; Guo et al., 2026). General-purpose multimodal tokenizers place vision, language, audio, and other signals in a shared token space through semantic or contrastive objectives (Lu et al., 2026; Bachmann et al., 2024; Lu et al., 2023; 2024). These tokenizers, however, are optimized for general visual generation or cross-modal alignment, and do not explicitly capture the metric geometry or ego motion required for driving scenes. Driving-specific tokenizers instead focus on scene representations. BEV-VAE and DriveTok construct tokens from multiview observations (Chen & Zhao, 2025; Zhuo et al., 2026), while OccWorld and I^2-World tokenize structured 3D occupancy for world modeling and 4D forecasting (Zheng et al., 2024; Liao et al., 2025b). Moving beyond camera- or occupancy-centric scene tokens, BEVWorld fuses surround-view images and LiDAR into a BEV latent that can be rendered back to both modalities, but the latent still describes the scene without encoding ego pose (Zhang et al., 2024). These limitations motivate a unified tokenizer that keeps appearance, metric geometry, and ego pose jointly decodable.

2.2 Driving Planners and World-Action Models

End-to-end driving systems jointly optimize perception and planning, from feature fusion and waypoint regression (Chitta et al., 2023) and planning-centric perception and prediction (Hu et al., 2023b) to anchored trajectory sets refined by truncated diffusion (Liao et al., 2025a) or autoregressive mixture-of-experts decoding (Feng et al., 2025). World-model-assisted planners expose future-scene information through future rollouts, predicted BEV features, or world-objective constraints on the planning latent (Li et al., 2025b; Zhang et al., 2025a; Yang et al., 2026). Vision-language-action planners add semantic priors and explicit reasoning, using either chain-of-thought reasoning with discrete action tokens or language-derived representations with diffusion planning and reward refinement (Zhou et al., 2025; Li et al., 2026d;a), and recent variants further incorporate generated futures through interleaved generation, latent conditioning, iterative replanning, or auxiliary supervision (Liu et al., 2026; Li et al., 2026b; Zuo et al., 2026a; Sheng et al., 2026; Li et al., 2026c; Zhou et al., 2026). These planners use future information in different ways, but generally treat future-scene prediction and trajectory generation as separate modeling components.

World-action models instead couple the two within a generative model. Existing world-action models differ primarily in how predicted future observations interact with trajectory generation. Some use parallel diffusion heads over a shared spatiotemporal latent (Zhang et al., 2025b) or attention over discrete future-frame tokens in an autoregressive sequence (Zhao et al., 2025), while others condition action transformers on cached video-denoising features (Xia et al., 2026), decode action chunks from a sampled future-video latent (Shi et al., 2026), or autoregressively interleave scene and action tokens (Ye et al., 2025; Chen et al., 2025c). While these designs couple future prediction and action generation, they do not generally provide a single decodable latent that jointly represents appearance, metric geometry, and ego motion.

In contrast, DriveUniTok unifies RGB appearance, projected LiDAR geometry, and clip-relative ego pose at the tokenization stage in one jointly decodable continuous latent. DriveUniWAM forecasts this latent for joint future-scene prediction and planning. Our contribution centers on unifying appearance, metric geometry, and ego motion at the representation level for world-action modeling.

3 Methodology

3.1 Problem Setup

A driving observation sequence $\{o_1, \ldots, o_T\}$ contains $T = 5$ consecutive frames sampled at 2 Hz. Each observation $o_i = (I_i, D_i, p_i)$ consists of a front-camera image $I_i \in \mathbb{R}^{3\times H\times W}$, a sparse depth

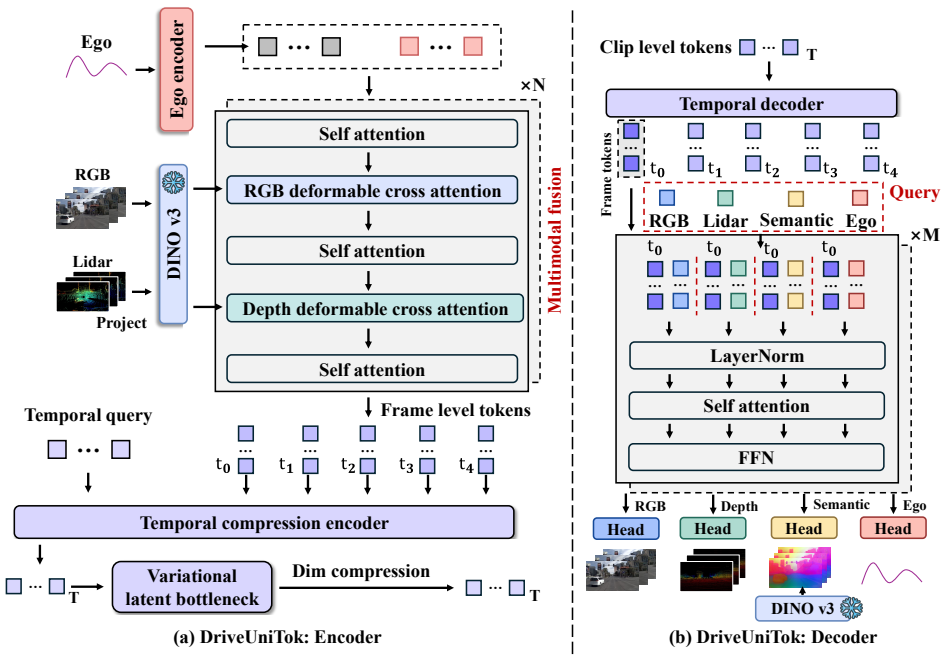

Figure 2: DriveUniTok architecture. **(a) Encoder.** RGB, projected depth, and ego pose are fused into S shared queries, whose T per-frame states are compressed into one latent. **(b) Decoder.** The latent is expanded back to T frame positions, and each target is read out through its own set of mask tokens, including a training-only semantic branch. Snowflakes mark frozen components, and all remaining modules are trained.

map $D_i \in \mathbb{R}^{1\times H\times W}$ obtained by projecting the LiDAR sweep onto the image plane of I_i, and an ego pose $p_i \in \mathbb{R}^3$ expressed relative to the first frame of the sequence, with $H \times W = 288 \times 512$. Given K observed sequences, we predict the next F sequences. Learning this mapping in observation space is expensive, and it leaves appearance, geometry, and motion in three different formats that a predictor would have to relate on its own. **DriveUniTok** therefore maps each sequence to a single unified latent $z \in \mathbb{R}^{S\times d}$ of $S = 584$ tokens with width $d = 16$, from which all three modalities are decodable. **DriveUniWAM** then models the evolution of z across sequences, and the DriveUniTok decoder maps predicted latents back to I, D, and p. In this division of labor, DriveUniTok determines which information the latent preserves, and DriveUniWAM determines how it evolves. We train DriveUniTok first, freeze it, and post-train DriveUniWAM against the NAVSIM PDM score.

3.2 DriveUniTok: Query Aligned Multimodal Tokenizer

DriveUniTok constructs the unified latent used by DriveUniWAM through three components: a query-based multimodal encoder that fuses the three modalities within each frame, a temporal compressor that reduces the T per-frame states of a sequence to one, and a shared decoder in which one set of mask tokens per target reads all three modalities back out of the compressed latent.

Multimodal encoder. Each input stream is first converted into features. RGB frames pass through a frozen DINOv3-L backbone (Siméoni et al., 2025), which yields a multilevel feature hierarchy. Rather than using a separate geometric backbone, we project the LiDAR sweep onto the image plane and process the resulting sparse depth map in the same feature space as RGB. The shared queries then extract appearance and geometric evidence through the same deformable-attention mechanism. The sparse depth map receives its own single-channel patch embedding, and the resulting tokens pass through the same frozen DINOv3 blocks with a learned modality embedding. Relative poses are mapped by a trainable ego state encoder into $N_E = 8$ ego tokens. The frozen backbone therefore supplies one feature hierarchy to both RGB and depth, while the depth patch embedding and the ego

state encoder only map their inputs into that shared token space. The driving-specific alignment itself is learned by the query layers described next.

The encoder uses one shared query set for all three streams. We initialize $N_I = 576$ image queries on the 18×32 patch grid and use the ego tokens as ego queries, together forming the $S = N_I + N_E$ tokens of the latent. The two groups differ in initialization and count rather than in modality, since they are updated jointly throughout the encoder and each ends up carrying evidence from all three inputs. Writing $Q \in \mathbb{R}^{S \times C}$ for the S queries of width C and F^{rgb}, F^{d} for the last four DINOv3 layers of the RGB and of the depth hierarchy, each encoder layer performs

$$Q \leftarrow \mathrm{SA}(Q), \qquad Q \leftarrow \mathrm{SA}\big(Q + \mathrm{DA}(Q, F^{\mathrm{rgb}})\big), \qquad Q \leftarrow \mathrm{SA}\big(Q + \mathrm{DA}(Q, F^{\mathrm{d}})\big), \tag{1}$$

with SA self-attention over the S queries and DA deformable cross-attention (Zhu et al., 2021) into a hierarchy. The same query sequence attends sequentially to RGB and depth features, while self-attention enables information exchange between the image-initialized and ego-initialized queries. This allows appearance, projected depth, and ego-pose information to interact before temporal compression. No modality is a privileged reference, and no fusion is deferred until after independent encoders have finished.

Temporal compression. Consecutive frames of a sequence are highly redundant, and keeping one latent per frame would make the downstream token sequence T times longer at quadratic attention cost. Each of the S queries produces one state per frame, and we compress those T states into one, independently for every query. Causal self-attention first processes the T states in order, then a learned pooling query gathers them into a single sequence-level state, which a Gaussian bottleneck (Kingma and Welling, 2014) maps to the d-dimensional code that forms z. Because depth and ego evidence are written into these shared queries rather than into token streams of their own, a sequence yields S tokens regardless of how many modalities it carries, roughly half of what separate per-modality representations would need. This gives DriveUniWAM a compact fixed-length latent in which appearance, geometry, and ego motion can be modeled together, while avoiding the duplicated token streams required by separate representations.

Multimodal decoder. A single latent step cannot be rendered directly, since the three targets are defined per frame. We therefore first project each token back to the decoder width, add learned frame queries for the T temporal positions, and expand the sequence representation to T frame-level memories with a temporal transformer. A shared Transformer (He et al., 2022; Zheng et al., 2025) then reads this memory through one set of learned mask tokens per target, each carrying its own positional embedding. The RGB and depth mask tokens yield patch features rendered by separate convolutional decoders into $\hat{I}$ and $\hat{D}$; the ego mask tokens feed the ego head that regresses $\hat{p}$. All three branches read the same memory, and only their mask tokens, positional embeddings, and output heads differ. This makes the multimodal alignment directly testable, since a modality absent from the shared latent could not be recovered by its branch.

Following VFMTok (Zheng et al., 2025), we add a training-only semantic target, because pixel reconstruction alone does not preserve the semantic content of the sampled foundation model features. A fourth set of mask tokens is read by the same shared Transformer into $\hat{\phi}$ and aligned by cosine similarity with the final layer patch tokens $\phi(I)$ of a separate frozen DINOv3-L applied to the ground truth frames.

Training objective. DriveUniTok is trained end to end with one term per readout,

$$\mathcal{L}_{\mathrm{tok}} = \lambda_{\mathrm{rgb}}\mathcal{L}_{\mathrm{rgb}} + \lambda_{\mathrm{depth}}\mathcal{L}_{\mathrm{depth}} + \lambda_{\mathrm{ego}}\mathcal{L}_{\mathrm{ego}} + \lambda_{\mathrm{sem}}\mathcal{L}_{\mathrm{sem}} + \lambda_{\mathrm{KL}}\mathcal{L}_{\mathrm{KL}}. \tag{2}$$

$\mathcal{L}_{\mathrm{rgb}}$ sums L_1, L_2, VGG LPIPS (Zhang et al., 2018), and ConvNeXt and DINOv3 perceptual distances. $\mathcal{L}_{\mathrm{depth}}$ is a confidence-weighted regression loss (Wang et al., 2024b) evaluated only where LiDAR projects to a valid pixel, plus a multiscale gradient term. $\mathcal{L}_{\mathrm{ego}} = \|\hat{p} - p\|_1$ and $\mathcal{L}_{\mathrm{sem}} = \mathbb{E}[1 - \cos(\hat{\phi}, \phi(I))]$. Appendix A gives every component and its weight. Sharing one decoder memory across these terms is what prevents the queries from degenerating into unrelated modality-specific subspaces while retaining the metric information driving requires.

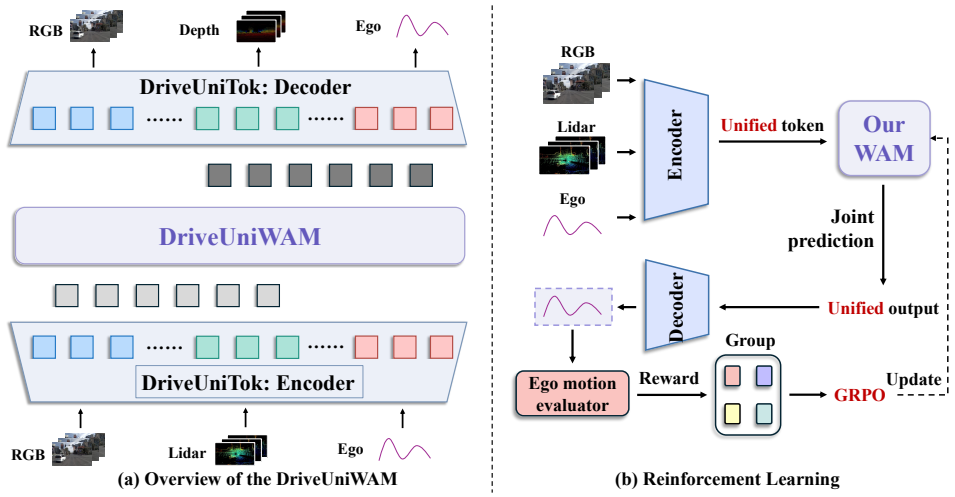

Figure 3: Training and policy optimization with DriveUniWAM. **(a) Supervised training.** DriveUniWAM forecasts the future latents with flow matching. **(b) Reinforcement learning.** Per scene, a group of sampled futures is decoded into trajectories and scored by the NAVSIM PDM reward, whose relative advantages drive the Flow-GRPO update. DriveUniTok stays frozen in both stages.

3.3 DriveUniWAM: Joint World-Action Modeling

DriveUniWAM follows the world-action model formulation introduced by DreamZero (Ye et al., 2026). Rather than forecasting observations and leaving action generation to a separate planner, it predicts future observations and actions within one generative process, so the imagined scene evolution and the executable action share a common dynamics representation. The central role of DriveUniTok is therefore to provide a latent in which the variables needed for world-action modeling are already jointly organized. Existing world-action models commonly inherit an RGB video latent and add action tokens beside it, leaving the model to discover the correspondence between appearance and motion during training. In DriveUniWAM that correspondence is already established by the frozen DriveUniTok latent, so the denoiser learns the evolution of an aligned driving state instead of simultaneously learning two prediction tasks and their correspondence.

DriveUniWAM uses the frozen DriveUniTok to encode the K observed and F future sequences. The history latents provide clean context, while the future latents are corrupted and reconstructed. We initialize the denoiser from the pretrained Wan2.2-TI2V-5B model (Team Wan et al., 2025) and adapt only its input and output projections to the DriveUniTok latent; a navigation embedding conditions it on the current vehicle state and route command. History queries attend only to history tokens, so the clean context cannot read noisy future targets, while every future query attends to the complete joint sequence, making all future tokens mutually visible during denoising. That visibility is what DriveUniTok makes reasonable, because each token already carries evidence from aligned appearance, geometry, and ego motion, so no artificial factorization between future observation and future action is needed.

Training uses flow matching (Lipman et al., 2023). A clean future latent z_0 is interpolated with Gaussian noise as $z_t = (1-t)z_0 + t\epsilon$ at a shared time t, and the denoiser regresses the velocity field $v_* = \epsilon - z_0$ under

$$\mathcal{L}_{\mathrm{IL}} = \mathbb{E}\left[w(t)\,\|v_\theta - v_*\|_2^2\right]. \tag{3}$$

Because appearance, geometry, and ego motion already share the tokens of z, a single shared denoiser predicts the velocity field over the complete unified latent, so the model learns future scenes and ego motion as one coherent continuation, as illustrated in Figure 3(a). The unified-latent loss, latent normalization, and supervised training schedule are described in Appendix C.1.

Reinforcement learning post-training. Flow matching fits recorded futures, whereas NAVSIM scores the consequences of a planned trajectory. We therefore post-train DriveUniWAM with Flow-GRPO (Liu et al., 2025) against the NAVSIM PDM reward (Dauner et al., 2024), as shown in

Table 1: Native-configuration comparison of video tokenizers on NAVSIM navtest. Comp. reports native temporal×height×width compression for grid tokenizers and tokens per clip for token-sequence methods; for DriveUniTok, it denotes the complete 584-token multimodal clip latent. The baselines use their official RGB-only preprocessing, whereas DriveUniTok additionally encodes projected depth and ego pose with a frozen DINOv3-L feature backbone. Results characterize fidelity–compression trade-offs under each tokenizer's native configuration.

	Tokenizer configuration		NAVSIM navtest					
Method	Comp.	Dim.	$\text{LPIPS}_{\text{VGG}}\downarrow$	$\text{LPIPS}_{\text{DINO}}\downarrow$	PSNR↑	SSIM↑	$\text{rFVD}_{\text{I3D}}\downarrow$	$\text{rFVD}_{\text{VJ}}\downarrow$
Large-Scale Pretrained Video Tokenizers								
Wan2.1 VAE (Team Wan et al., 2025)	4×8×8	16	0.1151	0.0649	31.50	0.8819	**2.06**	1.0984
Wan2.2 VAE (Team Wan et al., 2025)	4×16×16	48	**0.1081**	0.0608	31.35	0.8775	2.98	0.7958
HunyuanVideo VAE (Kong et al., 2024)	4×8×8	16	0.1230	**0.0573**	**32.36**	**0.8924**	4.03	**0.6278**
CogVideoX VAE (Yang et al., 2025)	4×8×8	16	0.1401	0.0943	32.26	0.8909	5.09	0.9298
Cosmos-0.1 (CV4x8x8) (Agarwal et al., 2025)	4×8×8	16	0.2362	0.3449	30.00	0.8481	26.76	4.1983
Cosmos-0.1 (CV8x16x16) (Agarwal et al., 2025)	8×16×16	16	0.3665	0.7327	26.03	0.7357	341.22	18.3995
AToken (Lu et al., 2026)	4×16×16	48	0.1644	0.1105	29.84	0.8493	7.75	1.2303
Benchmark-Scale Video Tokenizers								
Open-MAGVIT2 (Luo et al., 2024)	4×8×8	18	0.1659	0.1308	25.76	0.7802	34.57	14.4352
OmniTokenizer VAE (Wang et al., 2024a)	4×8×8	8	0.2049	0.3367	**26.89**	**0.8455**	40.70	6.8585
LARP-L-long (Wang et al., 2025)	1024 tokens	16	**0.1459**	0.1332	26.79	0.8042	**24.74**	5.7300
DriveUniTok (Ours)	584 tokens	16	0.1865	**0.1063**	25.62	0.7400	56.68	**3.4546**

Figure 3(b). For each scene the policy samples a group of joint futures, the frozen decoder reads a trajectory out of each sampled latent, and the group rewards give relative advantages for a clipped policy update regularized toward the supervised reference model. Only LoRA adapters (Hu et al., 2022) in the denoiser are trained; DriveUniTok, its decoder, and the reference policy stay frozen, so any gain reflects policy optimization around a fixed decodable latent rather than a change to the representation. Appendix C.2 states the full reward, transition replay, reference BC objective, and hyperparameters.

4 Experiments

4.1 Experimental Setup

We conduct experiments on NAVSIM, a non-reactive planning benchmark derived from the OpenScene subset of nuPlan (Dauner et al., 2024). We train DriveUniTok on the NAVSIM training split and then freeze it before training DriveUniWAM on the same split.

We evaluate multimodal reconstruction, joint future prediction, and planning. DriveUniTok encodes front-camera RGB, projected LiDAR depth, and clip-relative ego poses into 584 tokens. Visual reconstruction uses VGG- and DINOv3-based LPIPS (Zhang et al., 2018; Siméoni et al., 2025), PSNR, SSIM (Wang et al., 2004), and I3D- and V-JEPA 2.1-based FVD (Unterthiner et al., 2019; Carreira and Zisserman, 2017; Mur-Labadia et al., 2026); depth uses MAE, AbsRel, and RMSE, and ego motion uses ADE and FDE. Planning uses no-at-fault collisions (NC), drivable-area compliance (DAC), ego progress (EP), time to collision (TTC), comfort, and PDMS.

4.2 Tokenizer Reconstruction

We first test whether the compact unified latent preserves useful visual and temporal information by comparing DriveUniTok with large-scale pretrained video VAEs and benchmark-scale tokenizers under a common NAVSIM navtest protocol. Baselines use official preprocessing and native temporal windows, and each source frame is evaluated at most once. Visual baselines encode RGB only, whereas DriveUniTok encodes RGB, depth, and ego pose with frozen DINOv3-L features. We report native grid compression and DriveUniTok's 584-token multimodal sequence separately to characterize fidelity–compression trade-offs under each tokenizer's native configuration. Section 4.3 further compares unified and separate tokenization under matched representation and downstream computation budgets. Results are given in Table 1; qualitative RGB and depth reconstructions are shown in Figures 4–6 in Appendix B.

Table 2: Unified vs. separate tokenization on NAVSIM. Scalars counts latent values per clip; Len. is the DriveUniWAM sequence length at $K{=}1, F{=}2$. Costs are in TFLOPs; DiT cost is per denoising step. Lower reconstruction and prediction errors and higher PDMS are better.

	Representation budget			Efficiency			Reconstruction			Future prediction			Planning
Representation	Tokens	Scalars	Len.	Tok. params (M)	Tok. cost	DiT/step cost	$\text{LPIPS}_{\text{DINO}}$	AbsRel	ADE	gFVD_{VJ}	AbsRel	ADE	PDMS
Separate, natural budget	1,160	18,560	3,480	957.8	4.6	34.2	0.1083	0.0723	**0.0702**	7.92	0.1194	0.3997	90.1
Separate, equal budget	584	9,344	1,752	956.6	4.3	16.3	0.1124	0.0766	0.0720	12.80	0.1850	0.4186	87.5
Unified (DriveUniTok)	584	9,344	1,752	847.3	4.6	16.3	**0.1063**	**0.0461**	0.0728	**6.61**	**0.1059**	**0.3816**	**91.0**

DriveUniTok attains the lowest $\text{LPIPS}_{\text{DINO}}$ (0.1063) and rFVD_{VJ} (3.4546) among benchmark-scale methods, improving over the respective second-best results by 18.7% and 39.7%. Large pretrained VAEs retain an advantage on pixel-oriented metrics, but do not encode geometry or ego motion. These results indicate that the compact multimodal latent preserves useful semantic and temporal information without allocating separate modality-specific token streams. We next evaluate whether this representation also benefits future prediction and planning when used by a WAM.

4.3 Benefits of Unified Tokenization

We next evaluate the performance and resource efficiency of unified tokenization against separate modality-specific designs for WAM prediction and planning. We compare DriveUniTok against two separate-stream controls that use the same sensors, data, DriveUniWAM backbone, and training protocol. The natural-budget control assigns a full query budget to each modality and measures the resource efficiency of the unified design. The equal-budget control uses 288 RGB, 288 depth, and 8 ego queries, matching DriveUniTok's 584 tokens, $d = 16$ latent width, 9,344 latent scalars, sequence length, and downstream computation. Both models use the same frozen DINOv3-L features, semantic supervision, mask-token reconstruction objective, optimization recipe, and downstream training protocol; their architectural difference combines latent organization, parameter sharing, and cross-modal connectivity. Further implementation details are provided in Appendix A. The complete comparison is reported in Table 2.

Compared with the natural-budget control, DriveUniTok uses roughly half as many tokens and latent scalars, halves the sequence length, and reduces per-step DiT computation by about 52%, while improving PDMS from 90.1 to 91.0. At equal representation and DiT budgets, it uses fewer tokenizer parameters, reduces gFVD_{VJ} by about 48%, future depth AbsRel by about 43%, and trajectory ADE by nearly 9%, and raises PDMS by 3.5 points to 91.0. These results support the unified design under both budget settings. Appendices D and F provide objective and input-modality ablations.

4.4 Planning Performance

Table 3 compares DriveUniWAM with end-to-end, vision-language-action, and world-action planners on NAVSIM navtest under their reported sensor configurations. DriveUniTok remains frozen during both imitation learning and reward post-training.

DriveUniWAM reaches 90.5 PDMS with camera-only input, exceeding DriveWAM by 0.4 points. Adding projected LiDAR raises PDMS to 91.0, a further gain of 0.5 points and a 0.9-point improvement over DriveWAM. Reward post-training further increases PDMS to 91.54. With the tokenizer fixed, this gain reflects policy optimization within the same decodable latent space. DriveUniWAM also obtains 90.3 EPDMS on NAVSIM-v2; additional results and qualitative trajectories are provided in Appendix E and Figures 7–16 in Appendix G.

4.5 Backbone and Tokenizer Generalization

Table 4 tests DriveUniTok across video diffusion backbones. Part I replaces the native VAE while keeping the backbone fixed; Part II compares different backbones using DriveUniTok. Replacing the native VAE improves PDMS by 2.5 points for Wan2.1-1.3B and 3.1 points for Wan2.2-5B. Across five backbones, the unified latent achieves 88.9–91.0 PDMS, supporting its utility across model families and scales.

Table 3: NAVSIM navtest performance. C and L denote camera and LiDAR. The best and second-best learned results, including ties, are shown in **bold** and underlined.

Method	Reference	Sensors	NC↑	DAC↑	EP↑	TTC↑	Comfort↑	PDMS↑
Reference								
Human Agent	–	–	100.0	100.0	87.5	100.0	99.9	94.8
Vision-Based End-to-End Methods								
UniAD (Hu et al., 2023b)	CVPR'23	6×C	97.8	91.9	78.8	92.9	**100.0**	83.4
TransFuser (Chitta et al., 2023)	TPAMI'23	3×C+L	97.7	92.8	79.2	92.8	**100.0**	84.0
ARTEMIS (Feng et al., 2025)	RA-L'25	3×C+L	98.3	95.1	81.4	94.3	**100.0**	87.0
WorldRFT (Yang et al., 2026)	AAAI'26	3×C	97.8	96.8	81.7	94.0	**100.0**	87.8
DiffusionDrive (Liao et al., 2025a)	CVPR'25	3×C+L	98.2	96.2	82.2	94.7	**100.0**	88.1
WoTE (Li et al., 2025b)	ICCV'25	3×C+L	98.5	96.8	81.9	94.9	99.9	88.3
SeerDrive (Zhang et al., 2025a)	NeurIPS'25	3×C+L	98.4	97.0	83.2	94.9	99.9	88.9
Vision-Language-Action Methods								
UniWorldVLA (Liu et al., 2026)	arXiv'26	1×C	98.7	96.7	83.2	96.1	**100.0**	89.4
DriveDreamer-Policy (Zhou et al., 2026)	arXiv'26	3×C	98.4	97.1	83.5	95.1	**100.0**	89.2
Vega (Zuo et al., 2026a)	arXiv'26	1×C	98.9	95.3	81.6	96.1	**100.0**	87.9
ImagiDrive (Li et al., 2026b)	ICRA'26	1×C	98.6	96.2	80.5	94.5	**100.0**	87.4
AutoVLA (Zhou et al., 2025)	NeurIPS'25	3×C	98.4	95.6	81.9	**98.0**	99.9	89.1
ReCogDrive (Li et al., 2026d)	ICLR'26	1×C	97.9	97.3	87.3	94.9	**100.0**	90.8
ExploreVLA (Sheng et al., 2026)	ECCV'26	1×C	98.8	98.4	83.5	96.5	99.9	90.4
DriveVLA-W0 (Li et al., 2026c)	ICLR'26	1×C	98.7	**99.1**	83.3	95.3	99.3	90.2
SGDrive (Li et al., 2026a)	CVPR'26	1×C	98.6	97.8	85.8	96.2	**100.0**	91.1
World-Action Models								
Epona (Zhang et al., 2025b)	ICCV'25	1×C	97.9	95.1	80.4	93.8	99.9	86.2
PWM (Zhao et al., 2025)	NeurIPS'25	1×C	98.6	95.9	81.8	95.4	**100.0**	88.1
DriveLaW (Xia et al., 2026)	CVPR'26	1×C	**99.0**	97.1	81.3	96.7	**100.0**	89.1
DriveWAM (Shi et al., 2026)	arXiv'26	1×C	98.3	98.1	84.3	95.2	**100.0**	90.1
DriveUniWAM (Camera-only)	–	1×C	98.9	97.7	84.0	96.8	99.9	90.5
DriveUniWAM (Ours)	–	1×C+L	**99.0**	97.7	85.6	96.3	**100.0**	91.0
DriveUniWAM + RL (Ours)	–	1×C+L	98.7	98.0	**87.4**	95.6	99.8	**91.54**

Table 4: Ablation of video backbones and latent representations on the NAVSIM navtest benchmark. Part I compares the native VAE and DriveUniTok with fixed backbones, while Part II evaluates DriveUniWAM across video DiT backbones using the same unified latent. Across both parts, best and second-best results are shown in **bold** and underlined; improvements over the native VAE appear in parentheses.

Video backbone	Tokenizer	NC↑	DAC↑	EP↑	TTC↑	PDMS↑
Part I: Visual Tokenizer Representation Ablation						
Wan2.1-T2V-1.3B (Team Wan et al., 2025)	Native VAE	98.1	94.7	80.9	94.5	86.5
	DriveUniTok	98.4	97.0	82.9	95.4	89.0 (+2.5)
Wan2.2-TI2V-5B (Team Wan et al., 2025)	Native VAE	98.4	95.5	82.6	94.8	87.9
	DriveUniTok	99.0	**97.7**	**85.6**	96.3	**91.0** (+3.1)
Part II: Scalability across Video DiT Backbones (with DriveUniTok)						
Wan2.1-T2V-1.3B (Team Wan et al., 2025)	DriveUniTok	98.4	97.0	82.9	95.4	89.0
CogVideoX-2B (Yang et al., 2025)	DriveUniTok	98.9	96.5	82.5	96.2	88.9
Cosmos-Predict2.5-2B (Agarwal et al., 2025)	DriveUniTok	99.0	96.2	83.3	96.1	89.1
Wan2.2-TI2V-5B (Team Wan et al., 2025)	DriveUniTok	99.0	**97.7**	**85.6**	96.3	**91.0**
HunyuanVideo-1.5-8.3B (Tencent Hunyuan Foundation Model Team, 2025)	DriveUniTok	**99.1**	**97.7**	84.0	**96.8**	90.5

5 CONCLUSION

We presented DriveUniTok and DriveUniWAM for joint future-scene and ego-trajectory prediction through a unified multimodal latent. On NAVSIM, the unified design improves prediction and planning with fewer tokenizer parameters and lower downstream computational cost than the natural-budget separate-stream control. DriveUniWAM achieves 91.0 PDMS with supervised training and 91.54 after reinforcement learning post-training.

AI USE STATEMENT

We used AI tools to assist with code development and debugging, manuscript language editing, and literature search and review. The tools were used as research assistance rather than as a substitute for author verification of technical claims. The authors checked the implementation, experimental results, citations, and final manuscript, and take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

To support reproducibility, we release an anonymous code repository at `https://anonymous.4open.science/r/DriveUniTok-B358/`. The repository provides the implementation needed to reproduce our DriveUniTok tokenizer, DriveUniWAM training, and NAVSIM evaluation. The main paper and Appendices A, C.1, and C.2 document the model, objectives, training procedures, and evaluation details needed to use the released code.

REFERENCES

Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., et al. Cosmos World Foundation Model Platform for Physical AI. *arXiv preprint arXiv:2501.03575*, 2025.

Agro, B., Sykora, Q., Casas, S., Gilles, T., and Urtasun, R. UnO: Unsupervised Occupancy Fields for Perception and Forecasting. *arXiv preprint arXiv:2406.08691*, 2024.

Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Komeili, M., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., et al. V-JEPA 2: Self-Supervised Video Models Enable Understanding, Prediction and Planning. *arXiv preprint arXiv:2506.09985*, 2025.

Bachmann, R., Kar, O. F., Mizrahi, D., Garjani, A., Gao, M., Griffiths, D., Hu, J., Dehghan, A., and Zamir, A. 4M-21: An Any-to-Any Vision Model for Tens of Tasks and Modalities. *Advances in Neural Information Processing Systems*, 2024.

Bu, Q., Yang, Y., Cai, J., Gao, S., Ren, G., Yao, M., Luo, P., and Li, H. UniVLA: Learning to Act Anywhere with Task-centric Latent Actions. *arXiv preprint arXiv:2505.06111*, 2025.

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., and Beijbom, O. nuScenes: A multimodal dataset for autonomous driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020.

Carreira, J., and Zisserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Chen, B., Chen, Y., Qiu, L., Bai, J., Ge, Y., and Ge, Y. UniT: Toward a Unified Physical Language for Human-to-Humanoid Policy Learning and World Modeling. *arXiv preprint arXiv:2604.19734*, 2026.

Chen, X., Wei, H., Zhang, P., Zhang, C., Wang, K., Guo, Y., Yang, R., Wang, Y., Xiao, X., Zhao, L., et al. villa-X: Enhancing Latent Action Modeling in Vision-Language-Action Models. *arXiv preprint arXiv:2507.23682*, 2025a.

Chen, Y., Ge, Y., Tang, W., Li, Y., Ge, Y., Ding, M., Shan, Y., and Liu, X. Moto: Latent Motion Token as the Bridging Language for Learning Robot Manipulation from Videos. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19752–19763, 2025b.

Chen, Y., Wang, Y., and Zhang, Z. DrivingGPT: Unifying Driving World Modeling and Planning with Multi-modal Autoregressive Transformers. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025c.

Chen, Z. and Zhao, H. BEV-VAE: Multi-view Image Generation with Spatial Consistency for Autonomous Driving. *arXiv preprint arXiv:2507.00707*, 2025.

Chitta, K., Prakash, A., Jaeger, B., Yu, Z., Renz, K., and Geiger, A. TransFuser: Imitation with Transformer-Based Sensor Fusion for Autonomous Driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12878–12895, 2023.

Dang, C., Ang, S., Li, Y., Tian, H., Wang, J., Li, G., Ye, H., Ma, J., Chen, L., and Wang, Y. DriveFine: Refining-Augmented Masked Diffusion VLA for Precise and Robust Driving. *arXiv preprint arXiv:2602.14577*, 2026.

Dauner, D., et al. NAVSIM: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems*, 2024.

Fan, S., Wu, K., Che, Z., Wang, X., Wu, D., Liao, F., Liu, N., Zhang, Y., Zhao, Z., Xu, Z., et al. XR-1: Towards Versatile Vision-Language-Action Models via Learning Unified Vision-Motion Representations. *arXiv preprint arXiv:2511.02776*, 2025.

Feng, R., Xi, N., Chu, D., Wang, R., Deng, Z., Wang, A., Lu, L., Wang, J., and Huang, Y. ARTEMIS: Autoregressive End-to-End Trajectory Planning with Mixture of Experts for Autonomous Driving. *arXiv preprint arXiv:2504.19580*, 2025.

Fu, Y., Chen, N., Zhao, J., Shan, S., Yao, G., Wang, P., Wang, Z., and Zhang, S. METIS: Multi-source Egocentric Training for Integrated Dexterous Vision-Language-Action Model. *arXiv preprint arXiv:2511.17366*, 2025.

Guo, M., Wu, S., and Fei, H. V-RAE: Rethinking Video Latent Spaces for Generation. *arXiv preprint arXiv:2608.13556*, 2026.

He, K., et al. Masked Autoencoders Are Scalable Vision Learners. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., Shotton, J., and Corrado, G. GAIA-1: A Generative World Model for Autonomous Driving. *arXiv preprint arXiv:2309.17080*, 2023a.

Hu, E. J., et al. LoRA: Low-Rank Adaptation of Large Language Models. *International Conference on Learning Representations*, 2022.

Hu, X., Yin, W., Jia, M., Deng, J., Guo, X., Zhang, Q., Long, X., and Tan, P. DrivingWorld: Constructing World Model for Autonomous Driving via Video GPT. *arXiv preprint arXiv:2412.19505v1*, 2024.

Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., Lu, L., Jia, X., Liu, Q., Dai, J., Qiao, Y., and Li, H. Planning-oriented Autonomous Driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023b.

Kang, X., Shi, Y., Liang, L., Gan, R., Liu, D., Zhang, P., Chen, D., Qin, X., Zheng, Y., Zheng, J., Wang, H., Zhan, X., and Su, H. X-Tokenizer: A Multimodal Action Tokenizer for Vision-Language-Action Pretraining. *arXiv preprint arXiv:2606.14752v1*, 2026.

Kingma, D. P., and Welling, M. Auto-Encoding Variational Bayes. *International Conference on Learning Representations*, 2014.

Kong, W., et al. HunyuanVideo: A Systematic Framework for Large Video Generative Models. *arXiv preprint arXiv:2412.03603*, 2024.

Lee, S., Wang, Y., Etukuru, H., Kim, H. J., Shafiullah, N. M. M., and Pinto, L. Behavior Generation with Latent Actions. *arXiv preprint arXiv:2403.03181*, 2024.

Li, J., Wu, J., Hu, D., Huang, X., Sun, B., Hao, Z., Lang, X., Zhu, X., and Zhang, L. SGDrive: Scene-to-Goal Hierarchical World Cognition for Autonomous Driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026a.

Li, J., Zhang, B., Jin, X., Deng, J., Zhu, X., and Zhang, L. ImagiDrive: A Unified Imagination-and-Planning Framework for Autonomous Driving. *IEEE International Conference on Robotics and Automation*, 2026b.

Li, X., Wu, C., Yang, Z., Xu, Z., Zhang, Y., Liang, D., Wan, J., and Wang, J. DriVerse: Navigation World Model for Driving Simulation via Multimodal Trajectory Prompting and Motion Alignment. *ACM Multimedia*, 2025a.

Li, Y., Shang, S., Liu, W., Zhan, B., Wang, H., Wang, Y., Chen, Y., Wang, X., An, Y., Tang, C., Hou, L., Fan, L., and Zhang, Z. DriveVLA-W0: World Models Amplify Data Scaling Law in Autonomous Driving. *International Conference on Learning Representations*, 2026c.

Li, Y., Wang, Y., Liu, Y., He, J., Fan, L., and Zhang, Z. End-to-End Driving with Online Trajectory Evaluation via BEV World Model. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025b.

Li, Y., Xiong, K., Guo, X., Li, F., Yan, S., Xu, G., Zhou, L., Chen, L., Sun, H., Wang, B., Ma, K., Chen, G., Ye, H., Liu, W., and Wang, X. ReCogDrive: A Reinforced Cognitive Framework for End-to-End Autonomous Driving. *International Conference on Learning Representations*, 2026d.

Liao, B., Chen, S., Yin, H., Jiang, B., Wang, C., Yan, S., Zhang, X., Li, X., Zhang, Y., Zhang, Q., and Wang, X. DiffusionDrive: Truncated Diffusion Model for End-to-End Autonomous Driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025a.

Liao, Z., Wei, P., Zhang, R., Chen, S., Wang, H., and Ren, Z. I^2-World: Intra-Inter Tokenization for Efficient Dynamic 4D Scene Forecasting. *arXiv preprint arXiv:2507.09144*, 2025b.

Lipman, Y., et al. Flow Matching for Generative Modeling. *International Conference on Learning Representations*, 2023.

Liu, J., et al. Flow-GRPO: Training Flow Matching Models via Online RL. *Advances in Neural Information Processing Systems*, 2025.

Liu, Q., Xu, H., Li, J., Sun, B., Hao, Z., She, D., Zhu, X., and Zhang, L. Uni-World VLA: Interleaved World Modeling and Planning for Autonomous Driving. *arXiv preprint arXiv:2603.27287*, 2026.

Lu, J., Clark, C., Lee, S., Zhang, Z., Khosla, S., Marten, R., Hoiem, D., and Kembhavi, A. Unified-IO 2: Scaling Autoregressive Multimodal Models with Vision, Language, Audio, and Action. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Lu, J., Clark, C., Zellers, R., Mottaghi, R., and Kembhavi, A. Unified-IO: A Unified Model for Vision, Language, and Multi-Modal Tasks. *International Conference on Learning Representations*, 2023.

Lu, J., et al. AToken: A Unified Tokenizer for Vision. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Luo, Z., Shi, F., Ge, Y., Yang, Y., Wang, L., and Shan, Y. Open-MAGVIT2: An Open-Source Project Toward Democratizing Auto-Regressive Visual Generation. *arXiv preprint arXiv:2409.04410*, 2024.

Mittal, S., Mousakhan, A., Galesso, S., et al. Orbis 2: A Hierarchical World Model for Driving. *arXiv preprint arXiv:2607.15898*, 2026.

Mur-Labadia, L., Muckley, M., Bar, A., Assran, M., Sinha, K., Rabbat, M., LeCun, Y., Ballas, N., and Bardes, A. V-JEPA 2.1: Unlocking Dense Features in Video Self-Supervised Learning. *arXiv preprint arXiv:2603.14482*, 2026.

Pertsch, K., Stachowicz, K., Ichter, B., Driess, D., Nair, S., Vuong, Q., Mees, O., Finn, C., and Levine, S. FAST: Efficient Action Tokenization for Vision-Language-Action Models. *arXiv preprint arXiv:2501.09747*, 2025.

Sheng, Z., et al. ExploreVLA: Dense World Modeling and Exploration for End-to-End Autonomous Driving. *European Conference on Computer Vision*, 2026.

Shi, C., Xu, J., Shi, S., Sheng, K., Zhang, B., and Jiang, L. DriveWAM: Video Generative Priors Enable Scalable World-Action Modeling for Autonomous Driving. *arXiv preprint arXiv:2605.28544*, 2026.

Siméoni, O., et al. DINOv3. *arXiv preprint arXiv:2508.10104*, 2025.

Tang, A., He, T., Guo, J., Cheng, X., Song, L., and Bian, J. VidTok: A Versatile and Open-Source Video Tokenizer. *arXiv preprint arXiv:2412.13061*, 2024.

Team Wan et al. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv preprint arXiv:2503.20314*, 2025.

Tencent Hunyuan Foundation Model Team. HunyuanVideo 1.5 Technical Report. *arXiv preprint arXiv:2511.18870*, 2025.

Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., and Gelly, S. FVD: A New Metric for Video Generation. *International Conference on Learning Representations Workshop*, 2019. `https://openreview.net/forum?id=rylgEULtdN`.

van den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. *Advances in Neural Information Processing Systems*, 2017.

Wang, H., Suri, S., Ren, Y., Chen, H., and Shrivastava, A. LARP: Tokenizing Videos with a Learned Autoregressive Generative Prior. *International Conference on Learning Representations*, 2025.

Wang, J., Jiang, Y., Yuan, Z., Peng, B., Wu, Z., and Jiang, Y.-G. OmniTokenizer: A Joint Image-Video Tokenizer for Visual Generation. *Advances in Neural Information Processing Systems*, 2024a.

Wang, J., Zhang, Q., Yang, S., Luo, Y., Shen, Y., Wu, Z., Jiang, Y.-G., and Xu, Y. RepWAM: World Action Modeling with Representation Visual-Action Tokenizers. *arXiv preprint arXiv:2606.13674*, 2026.

Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., and Revaud, J. DUSt3R: Geometric 3D Vision Made Easy. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b.

Wang, X., et al. DriveDreamer: Towards Real-world-driven World Models for Autonomous Driving. *arXiv preprint arXiv:2309.09777*, 2023.

Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 2004.

Xia, T., Li, Y., Zhou, L., Yao, J., Xiong, K., Sun, H., Wang, B., Ma, K., Chen, G., Ye, H., Liu, W., and Wang, X. DriveLaW: Unifying Planning and Video Generation in a Latent Driving World. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Xu, J., Zhong, Z., Shu, Z., Jia, M., Li, M., Bian, J.-W., Zhang, Q., Zhang, K., Xie, J., Yang, J., and Yin, W. EponaV2: Driving World Model with Comprehensive Future Reasoning. *arXiv preprint arXiv:2605.14696*, 2026.

Yang, P., Lu, B., Xia, Z., Han, C., Gao, Y., Zhang, T., Zhan, K., Lang, X., Zheng, Y., and Zhang, Q. WorldRFT: Latent World Model Planning with Reinforcement Fine-Tuning for Autonomous Driving. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Yang, Z., Chen, L., Sun, Y., and Li, H. Visual Point Cloud Forecasting enables Scalable Autonomous Driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14673–14684, 2024.

Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al. CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer. *International Conference on Learning Representations*, 2025.

Yao, W., Li, Z., Lan, S., Wang, Z., Sun, X., Alvarez, J. M., and Wu, Z. DriveSuprim: Towards Precise Trajectory Selection for End-to-End Planning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026.

Ye, B., Zhang, B., and Zhao, H. DAP: A Discrete-token Autoregressive Planner for Autonomous Driving. *arXiv preprint arXiv:2511.13306*, 2025.

Ye, S., Ge, Y., Zheng, K., Gao, S., Yu, S., Kurian, G., Indupuru, S., Tan, Y. L., Zhu, C., Xiang, J., et al. World Action Models Are Zero-Shot Policies. *arXiv preprint arXiv:2602.15922*, 2026.

Ye, S., Jang, J., Jeon, B., Joo, S., Yang, J., Peng, B., Mandlekar, A., Tan, R., Chao, Y.-W., Lin, B. Y., et al. Latent Action Pretraining from Videos. *arXiv preprint arXiv:2410.11758*, 2024.

Yu, L., Lezama, J., Gundavarapu, N. B., Versari, L., Sohn, K., Minnen, D., Cheng, Y., Gupta, A., Gu, X., Hauptmann, A. G., Gong, B., Yang, M.-H., Essa, I., Ross, D. A., and Jiang, L. Language Model Beats Diffusion: Tokenizer is Key to Visual Generation. *International Conference on Learning Representations*, 2024.

Yuan, T., Dong, Z., Liu, Y., and Zhao, H. Fast-WAM: Do World Action Models Need Test-time Future Imagination? *arXiv preprint arXiv:2603.16666*, 2026.

Zhang, B., Song, N., Li, J., Zhu, X., Deng, J., and Zhang, L. Future-Aware End-to-End Driving: Bidirectional Modeling of Trajectory Planning and Scene Evolution. *Advances in Neural Information Processing Systems*, 2025a.

Zhang, K., Tang, Z., Hu, X., Pan, X., Guo, X., Liu, Y., Huang, J., Yuan, L., Zhang, Q., Long, X.-X., Cao, X., and Yin, W. Epona: Autoregressive Diffusion World Model for Autonomous Driving. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025b.

Zhang, R., Isola, P., Efros, A. A., Shechtman, E., and Wang, O. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Zhang, Y., Gong, S., Xiong, K., Ye, X., Tan, X., Wang, F., Huang, J., Wu, H., and Wang, H. BEVWorld: A Multimodal World Model for Autonomous Driving via Unified BEV Latent Space. *arXiv preprint arXiv:2407.05679v1*, 2024.

Zhao, G., Wang, Y., Wang, X., Zhu, Z., Yu, T., Huang, G., Zai, Y., Jiao, J., Xue, C., Wang, X., Yang, Z., Zhu, F., and Wang, X. UniDriveDreamer: A Single-Stage Multimodal World Model for Autonomous Driving. *arXiv preprint arXiv:2602.02002*, 2026a.

Zhao, Z., Fu, T., Wang, Y., Wang, L., and Lu, H. From Forecasting to Planning: Policy World Model for Collaborative State-Action Prediction. *Advances in Neural Information Processing Systems*, 2025.

Zhao, Z., Zhou, X., Xu, T., Sun, Z., Zhou, K., Wu, Y., Li, H., Liang, D., and Bai, X. SimWAM: A Simple World Action Model for End-to-End Autonomous Driving. *arXiv preprint arXiv:2608.07468*, 2026b.

Zheng, A., Wen, X., Zhang, X., Ma, C., Wang, T., Yu, G., Zhang, X., and Qi, X. Vision Foundation Models as Effective Visual Tokenizers for Autoregressive Image Generation. *arXiv preprint arXiv:2507.08441*, 2025.

Zheng, W., Chen, W., Huang, Y., Zhang, B., Duan, Y., and Lu, J. OccWorld: Learning a 3D Occupancy World Model for Autonomous Driving. *European Conference on Computer Vision (ECCV)*, 2024.

Zhou, Y., Wang, X., Shao, H., et al. DriveDreamer-Policy: A Geometry-Grounded World–Action Model for Unified Generation and Planning. *arXiv preprint arXiv:2604.01765*, 2026.

Zhou, Z., Cai, T., Zhao, S. Z., Zhang, Y., Huang, Z., Zhou, B., and Ma, J. AutoVLA: A Vision-Language-Action Model for End-to-End Autonomous Driving with Adaptive Reasoning and Reinforcement Fine-Tuning. *Advances in Neural Information Processing Systems*, 2025.

Zhu, X., et al. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *International Conference on Learning Representations*, 2021.

Zhuo, D., Zheng, W., Zuo, S., Yan, S., Hou, L., Zhou, J., and Lu, J. DriveTok: 3D Driving Scene Tokenization for Unified Multi-View Reconstruction and Understanding. *arXiv preprint arXiv:2603.19219*, 2026.

Zuo, S., Li, Y., Zheng, W., et al. Vega: Learning to Drive with Natural Language Instructions. *arXiv preprint arXiv:2603.25741*, 2026a.

Zuo, S., Xie, Z., Zheng, W., Xu, S., Li, F., Li, H., Chen, L., Yang, Z.-X., and Lu, J. DVGT-2: Vision-Geometry-Action Model for Autonomous Driving at Scale. *arXiv preprint arXiv:2604.00813*, 2026b.

A Tokenizer Implementation Details

Encoder and decoder configuration. Each input is resized to 288×512 and divided into patches of size 16, yielding an 18×32 grid. The encoder is initialized with $N_I = 576$ queries with spatial reference points and $N_E = 8$ queries with learned reference points, all with embedding width 1024. These query groups differ in initialization rather than modality: after joint attention, they form one unified sequence of $S = 584$ tokens, with every token able to carry evidence from RGB, depth, and ego motion. The DINOv3-L feature backbone is initialized from a pretrained checkpoint and kept frozen throughout tokenizer training; the separate DINOv3 semantic teacher is also frozen. The encoder contains two query layers; each layer applies joint self-attention over the S queries followed by deformable cross-attention (Zhu et al., 2021) over the RGB and depth feature hierarchies. We use four feature levels, four sampling points per level, and 16 attention heads. Reference points for the first query group are fixed at patch-grid centers, whereas the second query group uses learned reference points because it has no spatial prior.

Temporal compression and expansion each use two causal self-attention blocks. The T states of each unified query are temporally pooled into one sequence-level state and passed through a Variational Latent Bottleneck (VLB), which compresses the state to width $d = 16$. Thus, DriveUniTok produces one unified $z \in \mathbb{R}^{S\times 16}$ rather than separate image and ego latents. The decoder maps this latent back to width 1024 and applies 24 shared ViT blocks with 16 heads and full self-attention. Four sets of mask tokens read the shared decoder features; the RGB and depth branches use convolutional decoders with channel multipliers $[1, 1, 2, 2, 4]$ and two residual blocks per stage. We optimize all trainable tokenizer modules with AdamW at learning rate 10^{-4} and weight decay 0.01, clip gradients at norm 10, and use warmup followed by a cosine schedule. The query-aggregator parameter group uses a $0.1\times$ learning-rate multiplier. All DINOv3 parameters receive no gradient updates. This unified latent is the representation consumed by DriveUniWAM for joint future prediction and planning.

Equal-budget unified–separate comparison. We provide additional implementation details for the comparison in Table 2. The protocol matches the representation exposed to DriveUniWAM, allowing us to compare the downstream effectiveness of the unified and separate designs under matched representation and downstream computation budgets. DriveUniTok uses 584 unified tokens, initialized from 576 queries with spatial reference points and 8 queries with learned reference points. The equal-budget separate baseline uses 288 RGB queries, 288 projected-LiDAR depth queries, and 8 ego queries, also totaling 584 tokens. Both RGB and depth query sets use 12×24 grids. With latent width $d = 16$, both models therefore produce $584 \times 16 = 9{,}344$ latent scalars and the same downstream sequence length of $3 \times 584 = 1{,}752$ tokens, giving the same nominal self-attention budget.

Both arms use the same frozen DINOv3-L feature extractor, frozen semantic teacher, RGB/depth reconstruction objectives, semantic auxiliary loss, depth preprocessing, and mask-token reconstruction path. The semantic supervision configuration and loss weights are kept unchanged. Neither arm uses a separate masking stage; mask-token reconstruction is part of the same tokenizer training procedure for both models.

The architectural difference is the location and connectivity of cross-modal interaction. DriveUniTok uses one shared query sequence, allowing RGB, depth, and ego queries to exchange information through self- attention before temporal compression and the Variational Latent Bottleneck. The separate baseline uses independent RGB, depth, and ego streams: RGB queries attend only to RGB features, depth queries attend only to depth features, and ego queries do not attend to visual features. Each stream has its own query set, encoder stack, temporal compressor, and Variational Latent Bottleneck. The separate design therefore removes cross-modal interaction in the tokenizer rather than merely assigning different output slices to a shared encoder.

The observation decoders use the same ViT-based decoder configuration, including the same depth, number of attention heads, and MLP ratio of 4.0. For the separate baseline, decoder context is restricted to the corresponding token ranges: RGB tokens occupy $[0, 288)$, depth tokens occupy $[288, 576)$, and ego tokens occupy $[576, 584)$. The unified model uses the full shared token sequence as decoder context. In the reported separate-stream configuration, ego reconstruction directly uses the aggregated ego tokens rather than the shared decoder ego readout.

We match the tokenizer training recipe across the two arms, including the data protocol, batch size, learning rate of 10^{-4}, precision, gradient accumulation, learning-rate schedule, and 33K tokenizer updates. We likewise match the downstream DriveUniWAM recipe, including the same data protocol, model configuration, 130K updates, learning rate, warm-up schedule, and flow-shift setting.

The equal-budget condition matches token count, latent-scalar count, downstream sequence length, nominal attention cost, sensors, training data, and optimization recipe. It does not imply exact equality of trainable parameter counts: the separate baseline contains independent copies of the query sets, encoder stacks, temporal compressors, and Variational Latent Bottlenecks, whereas DriveUniTok uses a shared aggregation pathway. The frozen DINOv3-L backbone and shared depth patch embedding are common to both models and are not treated as additional trainable capacity. The equal-budget condition therefore matches the representation and downstream DiT budgets while exposing the parameter-efficiency difference between the two tokenizer designs. Together with the natural-budget comparison, this evaluates whether the unified design remains advantageous under both lower-resource and matched-budget settings.

Loss components and weights. We use $\lambda_{\mathrm{rgb}} = \lambda_{\mathrm{depth}} = \lambda_{\mathrm{ego}} = \lambda_{\mathrm{sem}} = 1$ and $\lambda_{\mathrm{KL}} = 0.1$. Table 5 expands $\mathcal{L}_{\mathrm{tok}}$ into all of its component terms and lists the weight assigned to each term within its group.

Table 5: Components of the DriveUniTok training objective. Ω denotes the set of pixels where LiDAR projects to a valid depth.

Group	Term	Form	Weight
$\mathcal{L}_{\mathrm{rgb}}$	pixel L_1	$\lVert\hat{I} - I\rVert_1$	1.0
	pixel L_2	$\lVert\hat{I} - I\rVert_2^2$	1.0
	LPIPS	VGG feature distance (Zhang et al., 2018)	1.0
	perceptual	ConvNeXt feature distance	0.1
	perceptual	DINOv3 feature distance	1.0
$\mathcal{L}_{\mathrm{depth}}$	confidence	$\mathbb{E}_\Omega[r\kappa - \alpha \log \kappa]$, α=0.2	1.0
	regression	$\mathbb{E}_\Omega[r]$, $r = \lvert\hat{D} - D\rvert$	10.0
	gradient	multiscale gradient matching on Ω	0.1
$\mathcal{L}_{\mathrm{ego}}$	pose L_1	$\lVert\hat{p} - p\rVert_1$ on $(x, y, \text{heading})$	1.0
$\mathcal{L}_{\mathrm{sem}}$	cosine	$\mathbb{E}[1 - \cos(\hat{\phi}, \phi(I))]$	1.0
$\mathcal{L}_{\mathrm{KL}}$	Gaussian KL	unified VLB	0.1

For valid projected pixels Ω, the depth confidence term uses $r\kappa - \alpha \log \kappa$ with $r = \lVert\hat{D} - D\rVert_2$, $\alpha = 0.2$, and κ the predicted confidence. The regression term uses the same valid pixels, and both confidence and regression values are filtered at the 0.98 quantile. Depth is clipped to $[1, 150]$ m and normalized by an average scale of 60. Batches with fewer than 100 valid points skip the depth terms. The semantic target is the final-layer patch tokens of a frozen DINOv3-L applied to the ground-truth frames, which is a separate instance from the frozen encoder backbone.

B Qualitative Tokenizer Reconstructions

Each group compares the ground-truth RGB image, its reconstruction from the unified latent, the LiDAR-projected ground-truth depth, and the reconstructed depth at valid LiDAR pixels. Across nine scenes and five consecutive frames per scene, the RGB reconstructions preserve road geometry, vehicles, and salient scene details, while the depth reconstructions retain the spatial structure and distance patterns present in the sparse LiDAR measurements. The comparisons provide a qualitative

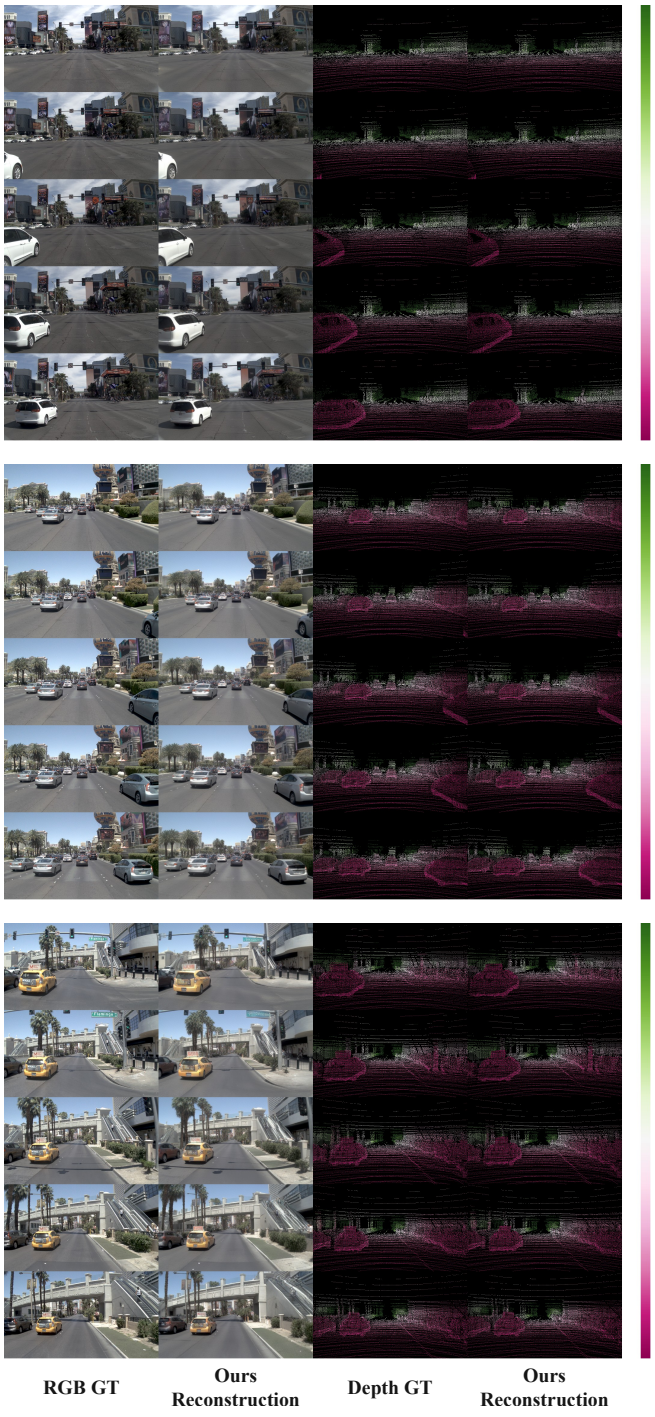

Figure 4: Qualitative tokenizer reconstructions for reconstruction group 1. From left to right, the columns show RGB ground truth, reconstructed RGB, LiDAR-projected depth ground truth, and reconstructed depth at valid LiDAR pixels.

complement to the reconstruction metrics in Table 1: the tokenizer preserves both appearance and metric 3D structure within the same compact representation.

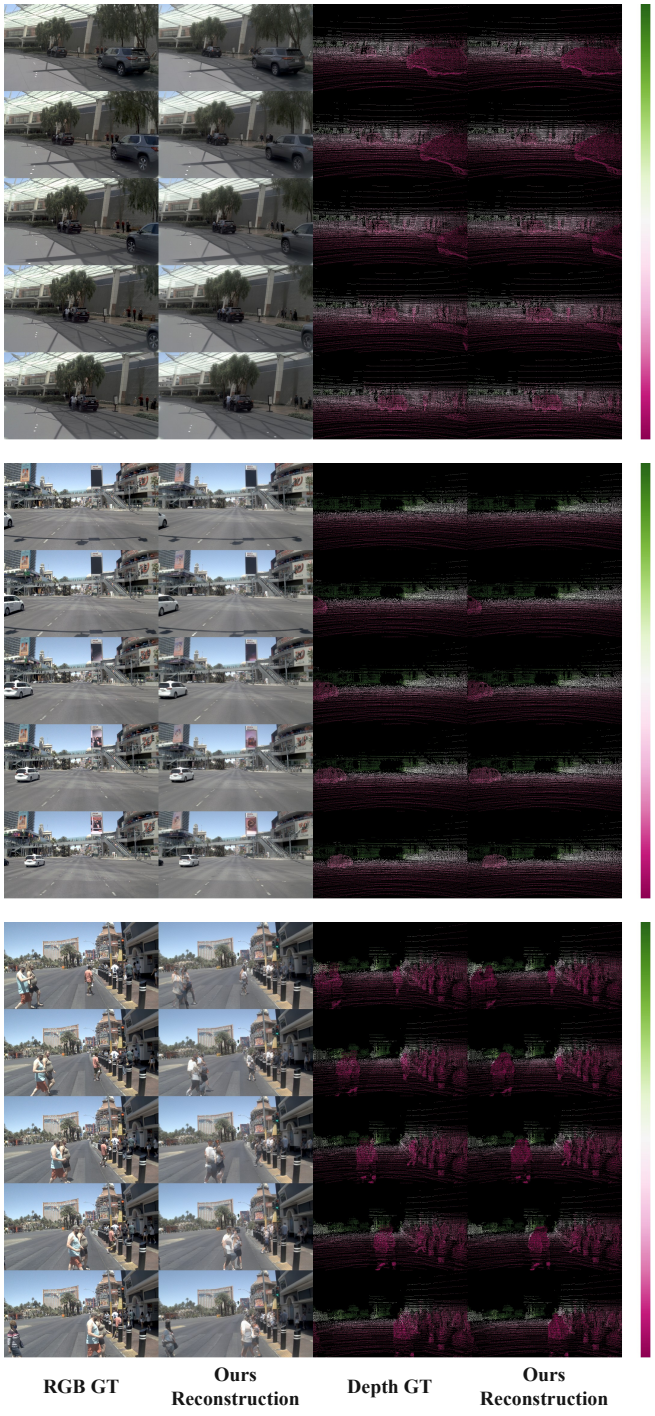

Figure 5: Qualitative tokenizer reconstructions for reconstruction group 2. From left to right, the columns show RGB ground truth, reconstructed RGB, LiDAR-projected depth ground truth, and reconstructed depth at valid LiDAR pixels.

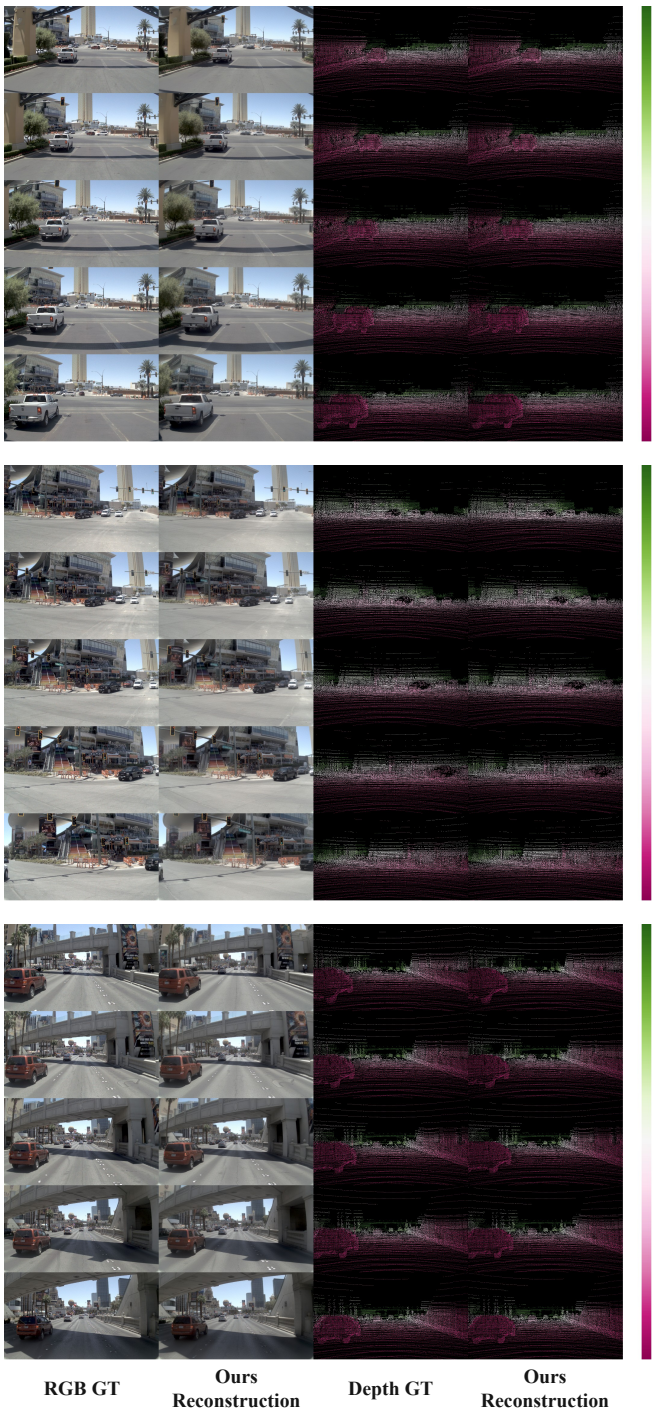

Figure 6: Qualitative tokenizer reconstructions for reconstruction group 3. From left to right, the columns show RGB ground truth, reconstructed RGB, LiDAR-projected depth ground truth, and reconstructed depth at valid LiDAR pixels.

C DriveUniWAM Training Details

C.1 Supervised Flow Matching

The supervised model uses one history clip and two future clips, corresponding to the $K{=}1, F{=}2$ setting. Each clip contains five frames sampled at 2 Hz. With shared boundary frames, the history and future clips cover frame indices $[0, 1, 2, 3, 4]$, $[4, 5, 6, 7, 8]$, and $[8, 9, 10, 11, 12]$, respectively. Each clip's ego pose is expressed relative to its first frame. When clips are concatenated, the poses of each subsequent clip are transformed into the coordinate frame anchored at the endpoint of the preceding clip, so the shared boundary frames produce a continuous trajectory. The unified latent is normalized per channel using statistics estimated on the training split. We sample Gaussian noise ϵ and a shifted flow time σ with shift 3.0, then construct

$$z_\sigma = (1-\sigma)z_0 + \sigma\epsilon, \qquad v_* = \epsilon - z_0. \tag{4}$$

The history clip remains clean, while the two future clips share the same σ and are noised independently. A single shared denoiser predicts the velocity field over the complete unified latent:

$$\mathcal{L}_{\mathrm{SFT}} = \mathbb{E}\left[w(\sigma)\,\|v_\theta - v_*\|_2^2\right]. \tag{5}$$

The loss is averaged over the complete unified token sequence. The navigation condition has eight values: four current-frame kinematic values and four route-command values. Only the Wan DiT parameters are optimized; DriveUniTok and its decoder remain frozen. The reported training uses AdamW with learning rate 10^{-4}, betas $(0.9, 0.95)$, weight decay 0.01, 1,000 warmup updates, cosine decay, gradient clipping at 1.0, and 130,000 optimizer updates.

C.2 Reinforcement Learning Post-Training

We post-train DriveUniWAM with Flow-GRPO (Liu et al., 2025) against the NAVSIM PDM reward (Dauner et al., 2024). For each scene the policy samples a group of $G = 8$ joint futures over 10 stochastic Flow-SDE denoising steps, and the frozen DriveUniTok decoder reads a trajectory out of each sampled latent for NAVSIM to score. For rewards $\{R_i\}_{i=1}^{G}$ the group relative advantage is

$$A_i = \frac{R_i - \mathrm{mean}_j(R_j)}{\mathrm{std}_j(R_j) + \varepsilon}. \tag{6}$$

Let $x_{i,k-1}$ denote the sampled next state from $x_{i,k}$ in the denoising chain. With the transition likelihood ratio $r_{i,k}(\theta) = \exp[\log p_\theta(x_{i,k-1} \mid x_{i,k}) - \log p_{\mathrm{old}}(x_{i,k-1} \mid x_{i,k})]$ we optimize the clipped objective

$$\mathcal{L}_{\mathrm{policy}} = -\mathbb{E}_{i,k}\left[\gamma^{N-k-1}\min\left(r_{i,k}(\theta)A_i, \mathrm{clip}(r_{i,k}(\theta), 1-\delta, 1+\delta)A_i\right)\right], \tag{7}$$

where N is the number of denoising steps, $\gamma = 0.6$ weights transitions by their contribution to the final sample, and $\delta = 0.02$. Since the trajectory is decoded from the unified latent, the likelihood ratio is computed over the transitions of the complete token sequence. A frozen reference model supplies a behavior cloning regularizer over the unified latent transition mean:

$$\mathcal{L}_{\mathrm{ref}} = \frac{\|\mu_\theta^z - \mu_{\mathrm{ref}}^z\|_2^2}{2\sigma^2}, \tag{8}$$

This regularizer keeps the policy close to the reference model on the decodable unified-latent manifold. The final objective is $\mathcal{L}_{\mathrm{RL}} = \mathcal{L}_{\mathrm{policy}} + \lambda_{\mathrm{ref}}\mathcal{L}_{\mathrm{ref}}$, with $\lambda_{\mathrm{ref}} = 0.1$. We train rank-16, scale-32 LoRA adapters (Hu et al., 2022) on the attention projections for four inner epochs per rollout at learning rate 5×10^{-5}. DriveUniTok, its decoder, and the reference policy remain frozen throughout post-training.

D Training Objective Ablation

This ablation isolates the masking and semantic-supervision objectives from the representation comparison. During tokenizer training on NAVSIM, intra-modal and cross-modal masks are sampled randomly for each training example. Intra-modal masking removes patches or tokens within an observed modality and reconstructs them from the remaining evidence in that modality. Cross-modal

Table 6: Training-objective ablation on NAVSIM navtest. Each row removes the named objectives from the full recipe. All variants share the architecture, 584×16 latent, and complete camera, depth, and ego inputs at evaluation. Best values, including ties, are bold.

	RGB				Depth		Ego	
Variant	$\text{LPIPS}_{\text{DINO}} \downarrow$	PSNR↑	SSIM↑	$\text{rFVD}_{\text{VJ}} \downarrow$	AbsRel↓	RMSE↓	ADE↓	FDE↓
w/o masked reconstruction and semantic supervision	0.1540	24.55	0.68	8.40	0.0695	9.30	0.1450	0.2670
w/o intra-modal masking	0.1215	25.20	0.72	4.85	0.0520	8.00	0.0900	0.1650
w/o cross-modal masking	0.1298	25.02	0.72	5.60	0.0565	8.30	0.1040	0.1910
w/o semantic supervision	0.1195	25.55	**0.74**	4.28	0.0480	7.72	0.0775	0.1420
Full training objective	**0.1063**	**25.62**	**0.74**	**3.45**	**0.0461**	**7.57**	**0.0728**	**0.1329**

Table 7: Performance on the NAVSIM-v2 navtest benchmark. Human Agent is included as a reference. DriveUniWAM (Ours) denotes the imitation-trained DriveUniWAM with a frozen DriveUniTok tokenizer. TLC denotes traffic-light compliance, HC denotes history comfort, EC denotes extended comfort, and EPDMS is the NAVSIM-v2 aggregate score. The best and second-best learned results, including ties, are shown in **bold** and underlined.

Method	Reference	NC↑	DAC↑	DDC↑	TLC↑	EP↑	TTC↑	LK↑	HC↑	EC↑	EPDMS↑
Reference											
Human Agent	–	100.0	100.0	99.8	100.0	87.4	100.0	100.0	98.1	90.1	94.5
Vision-Based End-to-End Methods											
TransFuser (Chitta et al., 2023)	TPAMI'23	96.90	89.90	97.80	99.70	87.10	95.40	92.70	98.30	87.20	76.7
DriveSuprim (Yao et al., 2026)	AAAI'26	97.50	96.50	99.40	99.60	88.40	96.60	95.50	98.30	77.00	83.1
DiffusionDrive (Liao et al., 2025a)	CVPR'25	98.20	95.90	99.40	99.80	87.50	97.30	96.80	98.30	**87.70**	84.5
Vision-Language-Action Methods											
ReCogDrive (Li et al., 2026d)	ICLR'26	98.30	95.20	99.50	99.80	87.10	97.50	96.60	98.30	86.50	83.6
SGDrive (Li et al., 2026a)	CVPR'26	98.60	94.30	99.50	**99.90**	86.00	97.90	96.10	98.30	85.90	86.2
DriveFine (Dang et al., 2026)	arXiv'26	**98.70**	97.30	99.50	99.80	**88.70**	97.80	97.70	**98.40**	83.80	89.7
DriveVLA-W0 (Li et al., 2026c)	ICLR'26	98.50	**99.10**	98.00	99.70	86.40	98.10	93.20	97.90	58.90	86.1
World-Action Models											
DVGT-2 (Zuo et al., 2026b)	arXiv'26	**98.70**	97.90	**99.70**	**99.90**	87.90	98.00	**98.20**	**98.40**	77.00	89.6
EponaV2 (Xu et al., 2026)	arXiv'26	98.50	97.40	99.50	**99.90**	87.90	98.10	97.70	98.20	77.40	88.9
DriveLaW (Xia et al., 2026)	CVPR'26	**98.70**	96.90	99.60	99.80	87.50	98.30	97.60	**98.40**	77.40	88.6
DriveUniWAM (Ours)	–	98.63	98.11	99.60	99.88	88.11	**98.51**	97.83	98.18	80.65	**90.3**

masking hides one target modality and reconstructs it from the others. Semantic supervision refers to the training-only semantic feature reconstruction branch; when it is removed, the RGB DINO perceptual objective is retained, so the ablation affects only the semantic readout. All variants use the same unified architecture, latent budget, NAVSIM training split, optimization budget, and full-input NAVSIM navtest evaluation protocol. The results are summarized in Table 6.

This ablation complements the equal-budget comparison by examining how the training objectives affect the quality of the unified latent itself. The full recipe achieves the best results across all reported RGB, depth, and ego metrics. Compared with omitting masked reconstruction and semantic supervision, it reduces $\text{LPIPS}_{\text{DINO}}$ from 0.1540 to 0.1063, rFVD_{VJ} from 8.4000 to 3.4546, depth AbsRel from 0.0695 to 0.0461, and trajectory ADE from 0.1450 to 0.0728. These results indicate that both the cross-modal reconstruction objective and the auxiliary semantic target contribute to the quality of the unified latent used by DriveUniWAM.

E NAVSIM-v2 Planning Results

We further evaluate the planning model under the NAVSIM-v2 protocol, which includes additional scene- and motion-compliance criteria. Table 7 shows that DriveUniWAM reaches 90.3 EPDMS while maintaining high scores across the individual metrics. This result supports the effectiveness of the unified latent representation for WAM planning: it preserves scene evidence, semantic structure, and the metric ego state needed to evaluate future action sequences.

Table 8: Modality sensitivity for reconstruction and joint future generation on NAVSIM navtest. C, L, and T denote camera, LiDAR/depth, and ego-trajectory modalities. Input and reconstruction Output follow the original tokenizer masking setup; the DriveUniWAM block evaluates future generation under the same input condition. Best results are highlighted in **bold**.

		Tokenizer Reconstruction				Joint Future Generation			
Input	Output	$\text{LPIPS}_{\text{DINO}}\downarrow$	$\text{rFVD}_{\text{VJ}}\downarrow$	Depth AbsRel↓	Ego ADE↓	$\text{LPIPS}_{\text{DINO}}\downarrow$	$\text{gFVD}_{\text{VJ}}\downarrow$	Depth AbsRel↓	Traj. ADE↓
C	C+L+T	0.1671	8.8645	0.1274	1.0512	0.1170	12.1153	0.2185	3.1991
L	L+T	–	–	0.1202	0.3350	–	–	0.2934	4.5854
T	T	–	–	–	0.2978	–	–	–	2.3534
C+L	C+L+T	0.1070	3.6430	0.0463	0.0876	0.0985	8.8587	0.1669	2.6611
C+L+T	**C+L+T**	**0.1063**	**3.4546**	**0.0461**	**0.0728**	**0.0955**	**6.6076**	**0.1059**	**0.3816**

F Modality Sensitivity

We finally vary the available input modalities to examine how different sources of evidence contribute to the unified latent, reconstruction, and joint future generation. Table 8 reports evaluation-time modality sensitivity under five input conditions, with the Input and Output configurations following the tokenizer masking setup. The DriveUniWAM block uses the same input condition as the tokenizer evaluation, allowing the effect of each modality to be tracked from representation quality to downstream prediction. Because these variants mask modalities at evaluation time rather than retraining with modality dropout, we do not interpret them as a complete sensor-failure robustness study.

Using all three input modalities yields the strongest overall reconstruction and future-generation results, indicating that the unified latent benefits from jointly integrating complementary visual, geometric, and motion evidence.

G Qualitative NAVSIM Planning Visualizations

The qualitative comparisons cover both vision-language-action and world-action modeling paradigms. Each group contains a front-view observation followed by predictions from AutoVLA and ReCogDrive (VLA methods), PWM (a world-action baseline), DriveUniWAM (Ours), and DriveUniWAM with reinforcement learning post-training. AutoVLA produces reasonable paths in straight-road scenes with simple geometry, but can under-steer at intersections and turns, resulting in large turning radii or endpoints that deviate from the ground-truth route. ReCogDrive generally makes better use of road topology, while PWM captures the broad scene-conditioned motion trend but can retain local curvature or endpoint errors in more demanding cases. DriveUniWAM further improves trajectory shape, turning curvature, and endpoint placement. After reinforcement learning post-training, the predicted paths show greater stability in complex intersections, turns, merges, and scenes with an explicit navigation direction: the model adjusts its direction earlier and better matches the ground-truth curvature, position, and final heading. Overall, the visualizations show that DriveUniWAM produces feasible trajectories across both comparison paradigms, while reinforcement learning post-training further improves their agreement with driving feasibility.



Figure 7: Qualitative NAVSIM navtest visualizations for scene group 1. From left to right, the columns show the front-view observation, AutoVLA, ReCogDrive, PWM, DriveUniWAM (Ours), and DriveUniWAM with reinforcement learning post-training. AutoVLA and ReCogDrive are vision-language-action methods, whereas PWM is a world-action baseline.

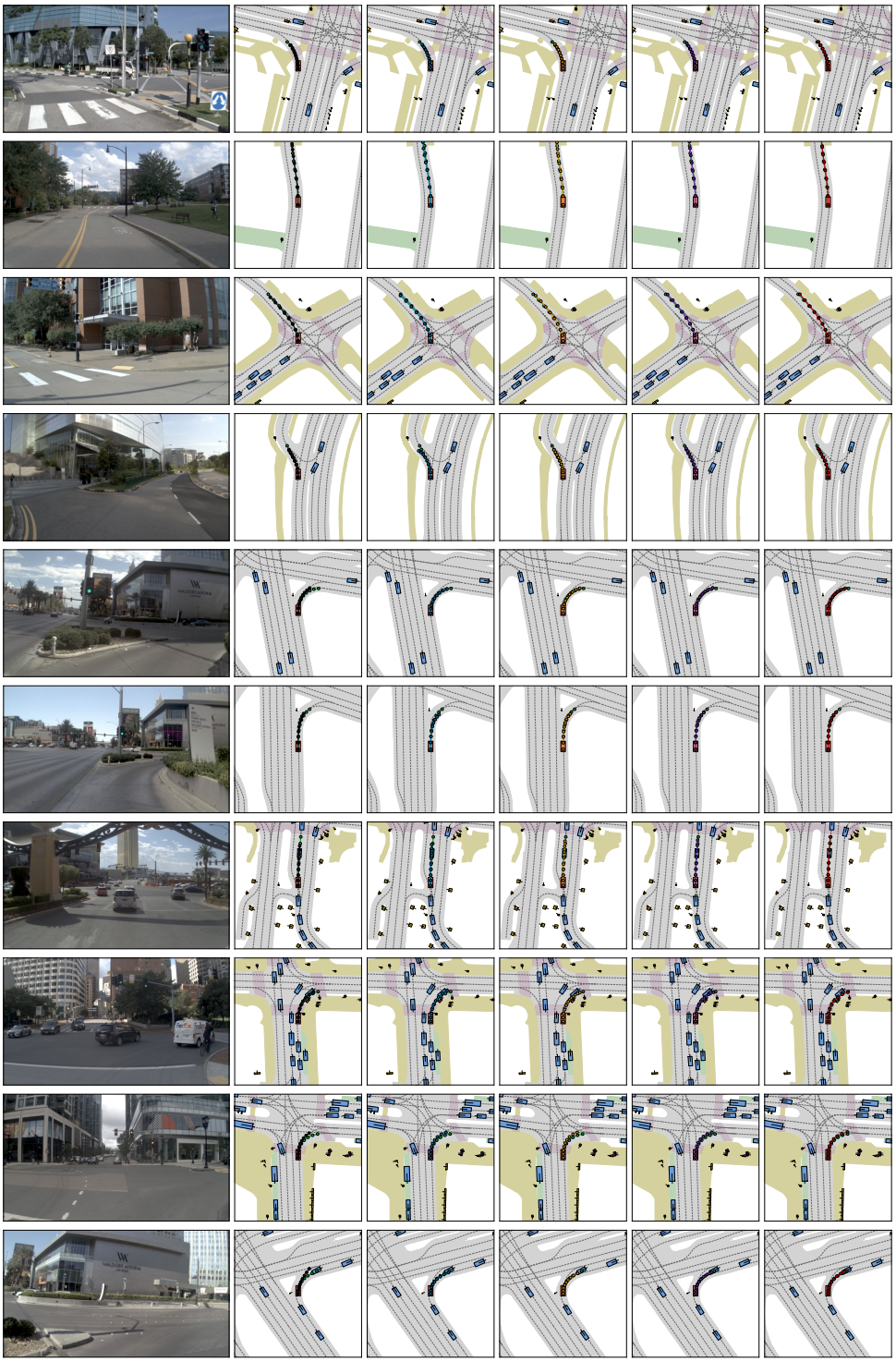

Figure 8: Qualitative NAVSIM navtest visualizations for scene group 2. The columns follow the ordering defined in Fig. 7.



Figure 9: Qualitative NAVSIM navtest visualizations for scene group 3. The columns follow the ordering defined in Fig. 7.



Figure 10: Qualitative NAVSIM navtest visualizations for scene group 4. The columns follow the ordering defined in Fig. 7.



Figure 11: Qualitative NAVSIM navtest visualizations for scene group 5. The columns follow the ordering defined in Fig. 7.



Figure 12: Qualitative NAVSIM navtest visualizations for scene group 6. The columns follow the ordering defined in Fig. 7.

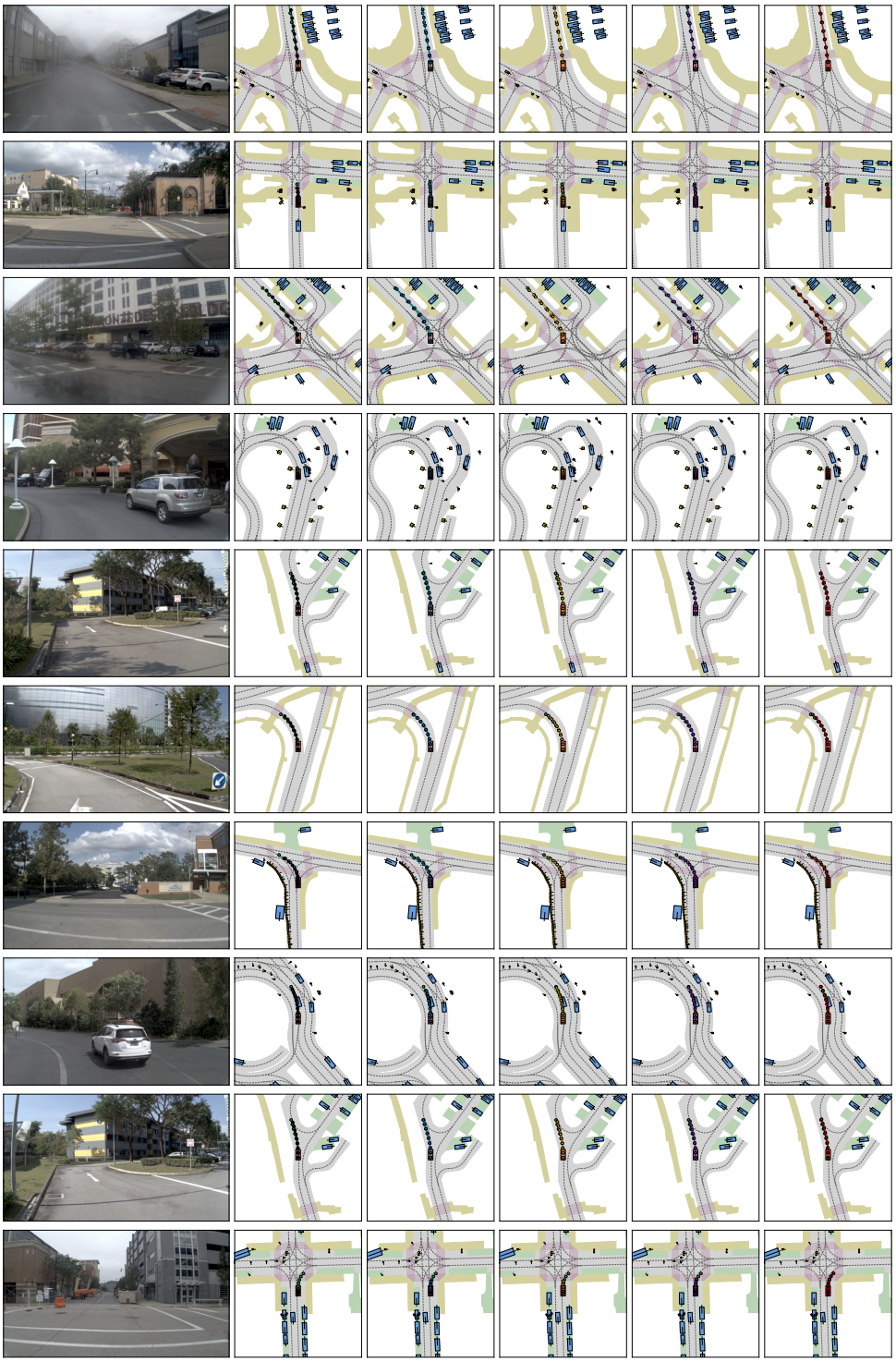

Figure 13: Qualitative NAVSIM navtest visualizations for scene group 7. The columns follow the ordering defined in Fig. 7.

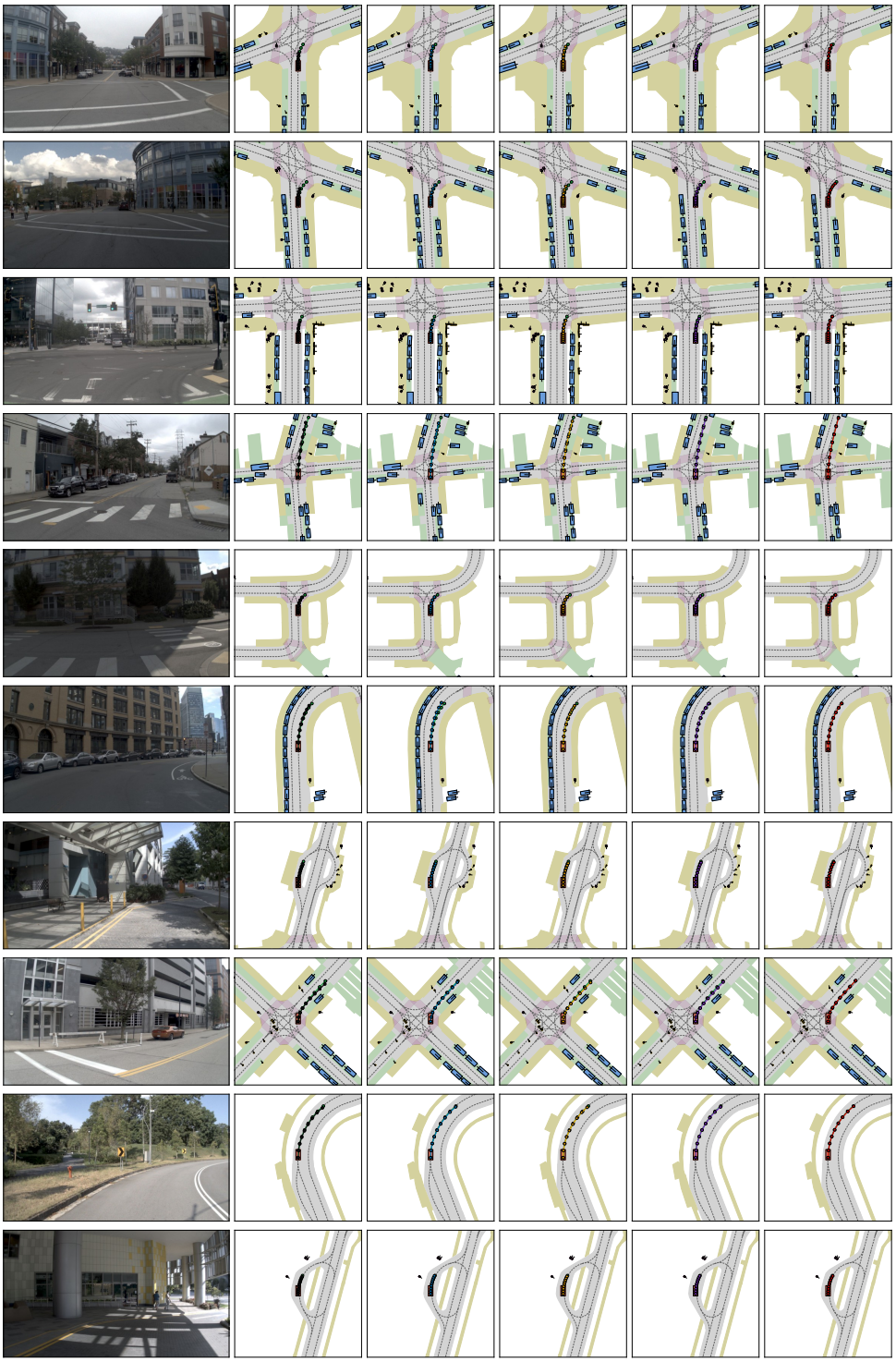

Figure 14: Qualitative NAVSIM navtest visualizations for scene group 8. The columns follow the ordering defined in Fig. 7.

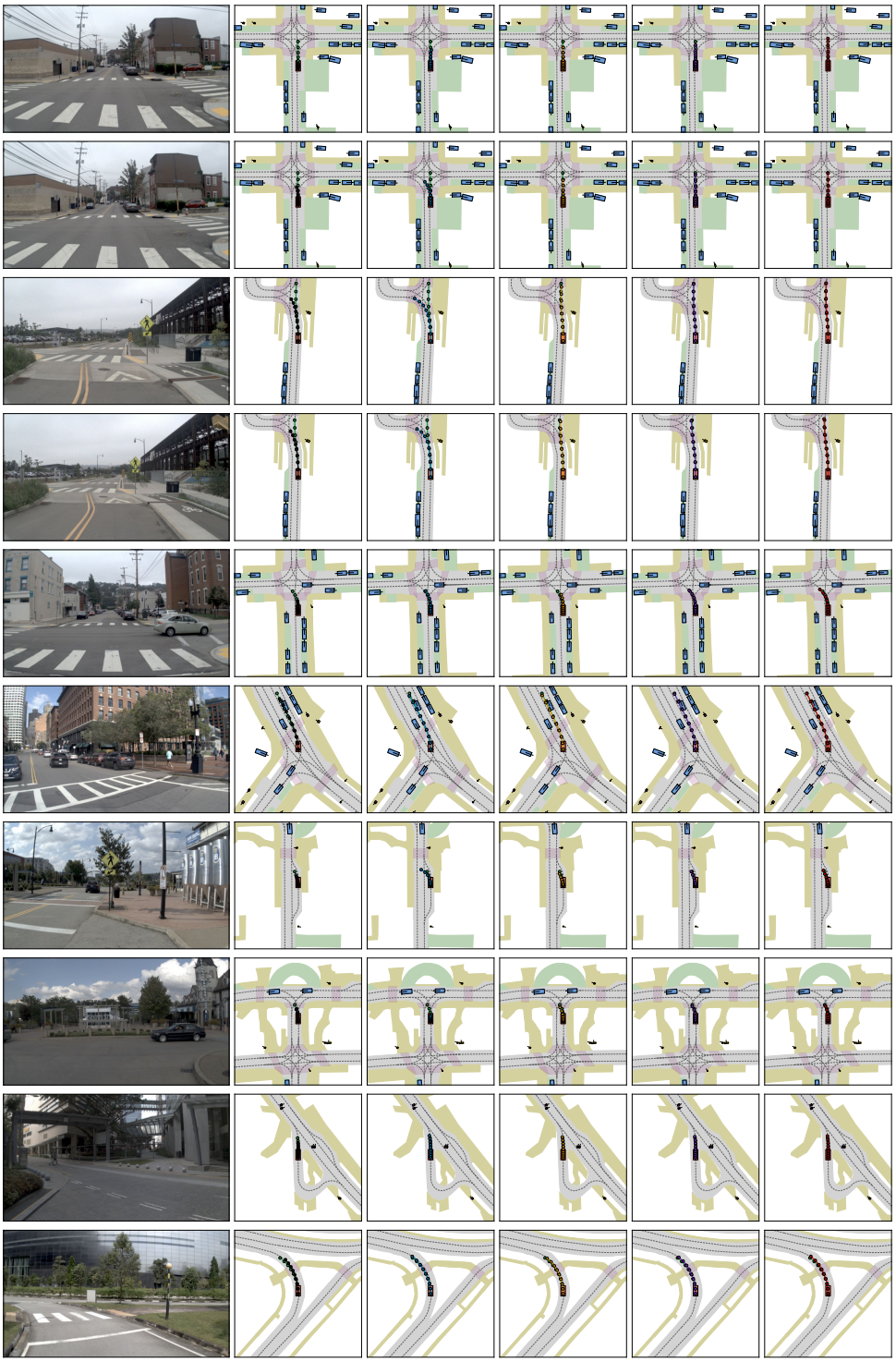

Figure 15: Qualitative NAVSIM navtest visualizations for scene group 9. The columns follow the ordering defined in Fig. 7.

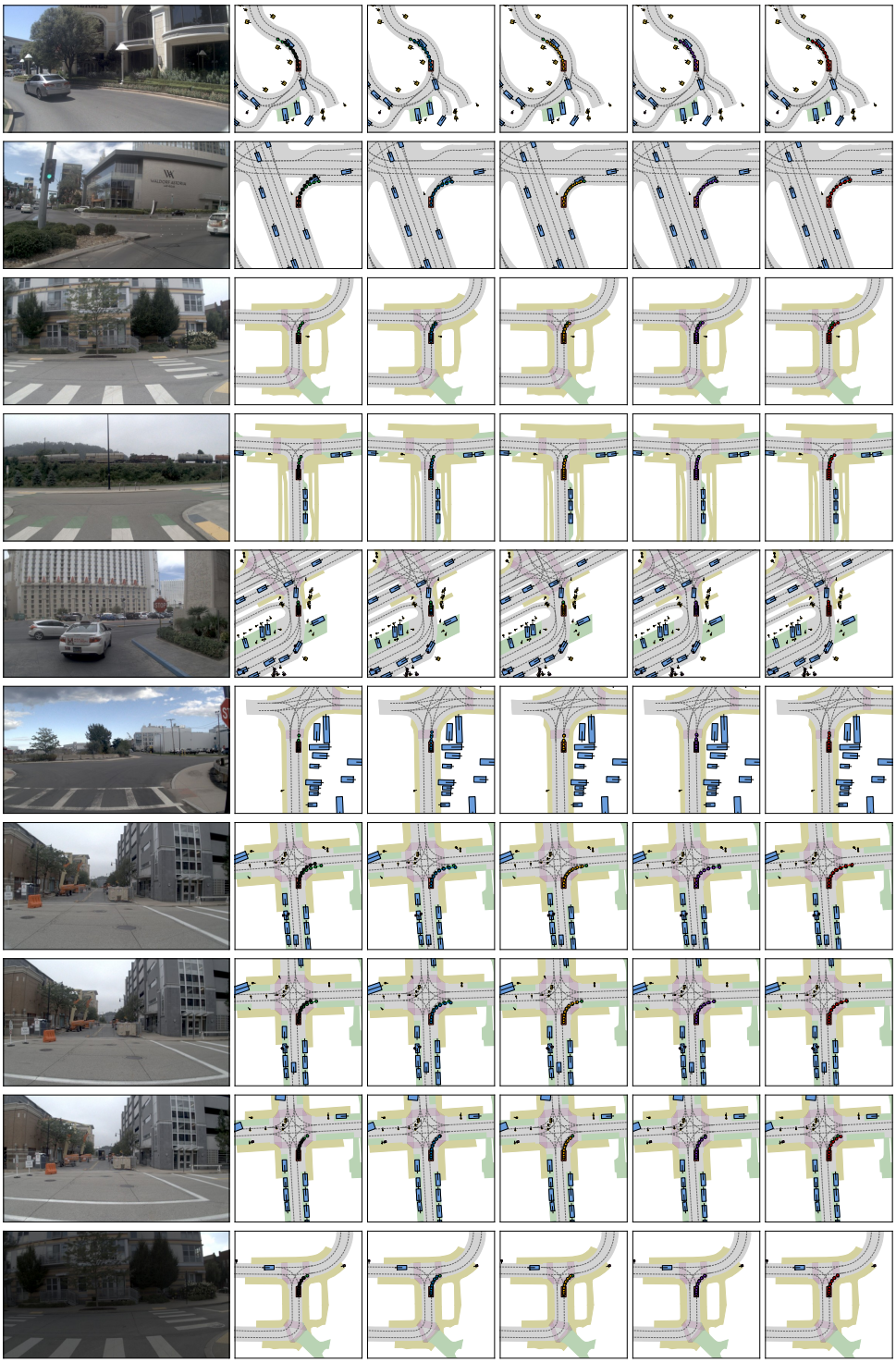

Figure 16: Qualitative NAVSIM navtest visualizations for scene group 10. The columns follow the ordering defined in Fig. 7.