DVGT-WORLD: EXTENDING VISION-GEOMETRY-ACTION PARADIGMS TO 3D WORLD MODELS FOR AUTONOMOUS DRIVING

Anonymous authors
Paper under double-blind review

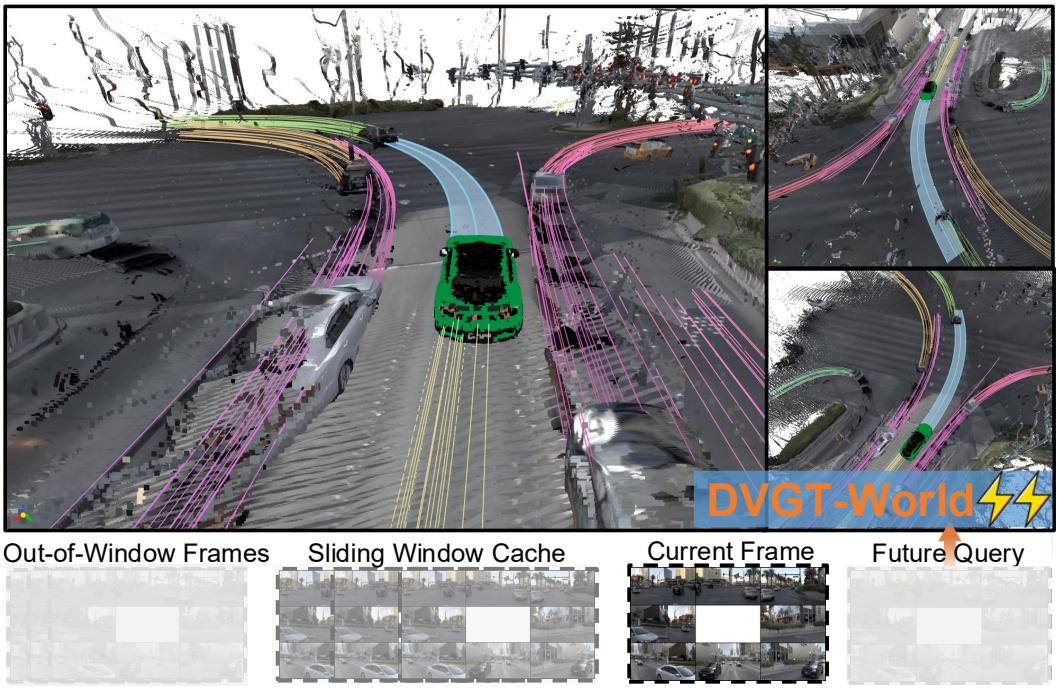

Figure 1: **DVGT-World is a streaming 3D world model for autonomous driving.** It uses multi-view images and a sliding-window cache to reconstruct current geometry and adopts future queries to predict 3D scene evolution, learning spatially grounded representations to support planning.

ABSTRACT

Autonomous driving requires not only understanding the current 3D scene but also predicting how it will evolve. Recent Vision-Geometry-Action (VGA) models have advanced end-to-end driving through explicit 3D geometry reconstruction. However, most VGA methods focus on reconstructing currently observed scenes, lacking explicit modeling of future 3D dynamics. While World-Action Models (WAMs) forecast scene evolution, their predictions in 2D image or latent spaces lack explicit 3D geometric supervision. To address this, we present **DVGT-World**, a framework that extends the VGA paradigm into a predictive 3D world model. By jointly supervising current geometry reconstruction and future geometry prediction, **DVGT-World** learns representations that capture both spatial structure and scene evolution, providing a geometric foundation for downstream planning. Specifically, we introduce learnable future queries conditioned on the target horizon into the geometry transformer to predict future dense pointmaps and ego poses from historical multi-view observations. After geometry pretraining, we fine-tune the model for planning, with future prediction disabled for inference efficiency. Experiments demonstrate accurate multi-view future geometry prediction and strong closed-loop planning performance, achieving 93.3 PDMS on NAVSIM v1 and 90.7 EPDMS on NAVSIM v2, which validates the benefit of modeling 3D scene evolution through future geometry prediction for downstream planning.

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.

1 INTRODUCTION

End-to-end autonomous driving has made substantial progress in mapping sensor observations to future trajectories (Hu et al., 2023; Jiang et al., 2023; Zhou et al., 2025b). To support scene understanding and planning, conventional methods typically use perception tasks such as object detection and map segmentation (Liao et al., 2025; Zheng et al., 2024c), while recent Vision-Language-Action (VLA) models leverage language reasoning to interpret driving contexts and guide decision-making (Zheng et al., 2024d; Zhou et al., 2025b; Li et al., 2025d). These approaches highlight the importance of learning informative scene representations for planning.

As vehicles operate in a 3D world, dense geometry provides rich spatial information about the surrounding environment, offering a foundation for reliable planning. Motivated by this, emerging Vision-Geometry-Action (VGA) models (Zuo et al., 2026; Lian et al., 2026) learn spatially grounded scene representations through dense geometry reconstruction. VGA models jointly predict dense pointmaps, ego poses, and future trajectories, using geometric supervision to learn fine-grained spatial features for planning. Furthermore, to support real-time driving, DVGT-2 (Zuo et al., 2026) adopts a streaming architecture that reuses cached historical features to avoid repeated computation, enabling efficient online inference of both geometry reconstruction and planning.

Despite these advances, existing VGA methods primarily learn geometric representations by reconstructing observed scenes, lacking explicit modeling of future 3D dynamics. In dynamic driving environments, safe planning requires not only understanding the current spatial relationships among roads, vehicles, and obstacles, but also predicting how these relationships will evolve (Zheng et al., 2024a; Li et al., 2025b). Recent World-Action Models (WAMs) learn scene evolution to support planning, but typically predict future 2D images or latents and lack explicit supervision of 3D geometric evolution (Xia et al., 2026; Liu et al., 2026; Xu et al., 2026). This motivates a 3D world model that learns scene dynamics through explicit supervision of future geometry.

To this end, we propose **DVGT-World**, which extends the streaming VGA paradigm into a predictive 3D world model. Building on DVGT-2, we append learnable future queries to the current multi-view image tokens for geometry prediction, with a time embedding specifying the target future frame. The geometry transformer processes the combined tokens through intra-view, cross-view, and temporal attention. Through temporal causal attention, future queries read current and cached historical features to predict dense pointmaps and ego poses. Current tokens do not attend to future queries and provide features for current geometry reconstruction and planning. The future branch can therefore be disabled during planning inference, avoiding its additional computational cost. We pretrain the model with joint supervision of current and future geometry, encouraging it to capture both the observed 3D structure and its evolution over time. We then fine-tune the model for planning, transferring the learned scene representations to trajectory prediction.

Experiments on NAVSIM demonstrate the effectiveness of **DVGT-World** in both future geometry prediction and closed-loop planning. **DVGT-World** achieves 93.3 PDMS on NAVSIM v1 (Dauner et al., 2024) and 90.7 EPDMS on NAVSIM v2 (Cao et al., 2025), outperforming the compared VLA, WAM, and VGA models. Geometry evaluations demonstrate accurate predictions across multiple views and future horizons, while controlled ablations confirm the benefit of future geometry supervision for planning. Together, these results show that extending geometry reconstruction to 3D world modeling provides an effective foundation for autonomous driving.

2 RELATED WORK

2.1 END-TO-END AUTONOMOUS DRIVING

Conventional end-to-end driving methods typically rely on auxiliary perception tasks (Hu et al., 2023; Jiang et al., 2023; Zheng et al., 2024b), such as object detection (Huang et al., 2021; Li et al., 2023), map segmentation (Li et al., 2024b; Liu et al., 2023), and occupancy prediction (Huang et al., 2023; 2024), to learn scene representations and predict trajectories. Such supervision primarily focuses on task-specific scene elements, with less emphasis on fine-grained scene geometry. Recently, Vision-Language-Action (VLA) models have leveraged large vision-language models to interpret driving contexts and connect semantic reasoning with trajectory prediction (Hwang et al., 2024; Zhou et al., 2025a; Li et al., 2025d). However, semantic reasoning alone does not explicitly con-

strain dense 3D geometry. Our work jointly supervises current geometry reconstruction and future geometry prediction to learn representations of 3D scene structure and evolution for robust planning.

2.2 Vision-Geometry-Action Models for Autonomous Driving

Inspired by visual geometry models (Wang et al., 2025a;c; Zuo et al., 2025) that reconstruct dense pointmaps from multi-view images, DVGT-2 (Zuo et al., 2026) introduced the Vision-Geometry-Action (VGA) paradigm. It jointly performs dense geometry reconstruction and trajectory planning, learning spatially grounded scene representations through geometric supervision. It employs a streaming geometry transformer that reuses cached historical features to avoid repeated computation during online inference. LiAuto-GeoX (Lian et al., 2026) distills a large geometry model into a compact model for efficient reconstruction and transfers the learned geometric representations to planning. Geometry supervision has also been incorporated into VLA and WAM frameworks through pointmap reconstruction or depth generation (Yao et al., 2026; Zhou et al., 2026). However, geometric supervision in these approaches primarily focuses on reconstructing observed scenes rather than predicting future geometry. Our work extends geometry pretraining to future dense pointmap and ego-pose prediction, explicitly learning 3D scene evolution to support downstream planning.

2.3 World-Action Models for Autonomous Driving

World-Action Models (WAM) learn scene evolution through future prediction to support trajectory planning. Latent world models (Li et al., 2024a; Zheng et al., 2025; Wang et al., 2026) predict future scene representations in latent feature space, while video world models (Xia et al., 2026; Liu et al., 2026) forecast future images or videos. However, latent feature or image prediction alone does not explicitly supervise future 3D geometry. Recent WAM methods incorporate geometric priors through feature distillation (Wang et al., 2026), or introduce future depth and semantic prediction (Xu et al., 2026). These advances highlight the value of geometric supervision for learning scene dynamics. However, these methods still fall short of modeling dense scene geometry reconstruction and prediction. Our work extends the streaming VGA framework to predict multi-view dense pointmaps and ego poses at specified future horizons. By jointly supervising current geometry reconstruction and future geometry prediction during pretraining, we learn representations of 3D scene structure and evolution for downstream planning.

3 Method

In this section, we detail the proposed **DVGT-World**. We first establish the formulation of 3D world modeling for the Vision-Geometry-Action (VGA) paradigm in Section 3.1. We then provide an overview of the model architecture and its data flow in Section 3.2. Subsequently, we elaborate on the specific implementations of the 3D world model for current geometry reconstruction and prediction in Sections 3.3 and 3.4, respectively. Section 3.5 details the two-stage training strategy.

3.1 3D World Modeling Formulation

As discussed in Section 2, existing VGA models primarily focus on reconstructing observed scene geometry. To extend this paradigm into a predictive 3D world model, we first define how to represent future scene geometry and its relation to the ego-vehicle.

Following the VGA formulation of DVGT-2, the current scene geometry is represented by multi-view local pointmaps $\mathbf{P}_t \in \mathbb{R}^{V\times H\times W\times 3}$ and a relative ego-pose $\mathbf{T}_{t-1\to t} \in \mathbb{R}^7$. Specifically, the pointmap aligns strictly with the V-view input images, where each pixel is assigned a 3D point coordinate (x, y, z) in the current ego-vehicle coordinate system. The relative ego-pose comprises a 3D translation and a 4D rotation quaternion of the ego-vehicle from the previous frame to the current frame. Notably, since pointmaps follow image pixel grids, their 3D sampling density is view-dependent: nearby surfaces generally occupy more pixels and are densely sampled, while distant regions project onto fewer pixels and are sparsely represented.

To model future scene geometry, we formulate 3D world modeling as horizon-conditioned future geometry prediction. Given observations up to the current frame, the model anticipates how the

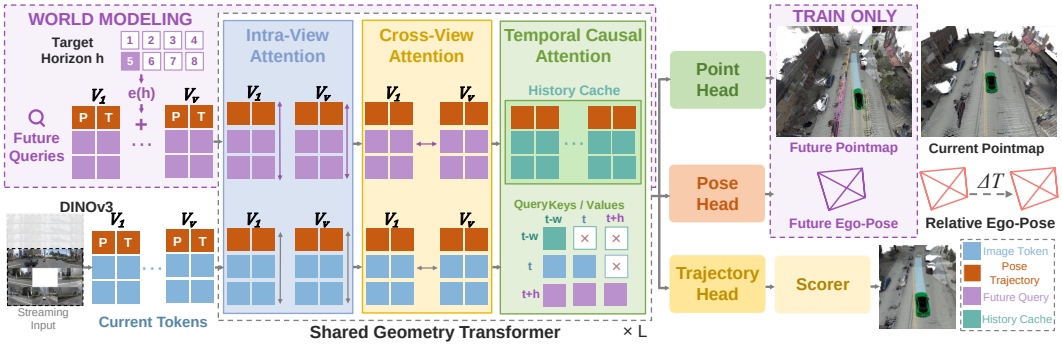

Figure 2: **Overall architecture of DVGT-World.** The image encoder first extracts multi-view current tokens. The geometry transformer integrates them with a sliding-window cache to reconstruct current pointmaps and ego-pose. During pretraining, horizon-conditioned future queries are appended to predict future geometry. Current features support trajectory planning, with the future queries are disabled during planning fine-tuning and inference.

local 3D scene evolves at different future horizons. Specifically, it predicts both the future ego-pose and a dense pointmap aligned with the future ego viewpoint:

$$(\mathbf{P}_{t+h}, \mathbf{T}_{t\to t+h}) = \mathcal{W}(\mathbf{I}_{1:t}, h).$$

The relative ego-pose describes the vehicle's future motion, while the pointmap represents scene geometry in the corresponding future ego-coordinate system. This future-ego-centric pointmap aligns geometric supervision with the spatial perspective relevant to trajectory planning, while prediction across horizons encourages the model to capture scene evolution, including the motion of dynamic objects and ego vehicle. Supervising these predictions during pretraining provides a spatially grounded representation for downstream planning.

3.2 Overall Architecture

DVGT-World unifies current geometry reconstruction, future geometry prediction, and ego-vehicle trajectory planning within a single end-to-end framework. To achieve this, the overall architecture comprises three core components: a visual encoder $\mathcal{E}$, a geometry transformer $\mathcal{G}$, and a planning module $\mathcal{P}$. At timestep t, the model processes current multi-view images $\mathbf{I}_t$ together with a historical cache $\mathbf{C}_{t-1}$ from the past w frames. A target horizon h specifies the future frame for geometry prediction, while current ego-vehicle states $\mathbf{s}_t$ are used for planning.

The visual encoder first extracts visual features $\mathbf{F}_t$ from the current images. The geometry transformer then fuses these features with the historical cache to simultaneously encode the current scene representation $\mathbf{Z}_t$ and forecast the future scene representation $\mathbf{Z}_{t+h}$ at the target timestep $t+h$:

$$\mathbf{F}_t = \mathcal{E}(\mathbf{I}_t), \qquad (\mathbf{Z}_t, \mathbf{Z}_{t+h}) = \mathcal{G}(\mathbf{F}_t, \mathbf{C}_{t-1}, h).$$

During streaming inference, the cache is updated with current scene features in a First-In-First-Out (FIFO) manner, retaining a fixed window of observed frames. For geometry reconstruction and prediction, both current and future scene representations contain visual and pose tokens:

$$\mathbf{Z} = [\mathbf{Z}^{\text{vis}}, \mathbf{Z}^{\text{pose}}].$$

These tokens are processed by their respective task heads to decode the physical predictions. Specifically, the visual tokens $\mathbf{Z}^{\text{vis}}$ and pose tokens $\mathbf{Z}^{\text{pose}}$ are fed into a point head $\mathcal{H}_{\text{point}}$ and a pose head $\mathcal{H}_{\text{pose}}$ to predict the dense pointmaps $\mathbf{P}$ and relative ego-poses $\mathbf{T}$:

$$\mathbf{P} = \mathcal{H}_{\text{point}}(\mathbf{Z}^{\text{vis}}), \qquad \mathbf{T} = \mathcal{H}_{\text{pose}}(\mathbf{Z}^{\text{pose}}).$$

For planning, we add learnable planning tokens to the current observation tokens in the geometry transformer. The resulting current planning features $\mathbf{Z}_t^{\text{plan}}$, together with the current ego states $\mathbf{s}_t$, are fed into a multi-trajectory planner to generate candidate trajectories and a scorer (Li et al., 2025e) to select the final trajectory $\mathbf{A}_t$:

$$\mathbf{A}_t = \mathcal{P}(\mathbf{Z}_t^{\text{plan}}, \mathbf{s}_t).$$

Future geometry prediction is disabled during planning fine-tuning and inference for efficiency.

By explicitly supervising both current geometry reconstruction and future geometry prediction, the model learns representations of 3D scene structure and evolution to support robust planning.

3.3 Streaming Geometry Reconstruction

To model the current 3D scene structure, **DVGT-World** inherits the streaming geometry architecture of DVGT-2 (Zuo et al., 2026). The visual features $\mathbf{F}_t$, together with learnable pose tokens, are fed into a geometry transformer. Planning tokens are additionally included during planning fine-tuning and inference. The transformer sequentially applies intra-view attention to model local structures within each camera view, cross-view attention to fuse multi-camera information, and temporal causal attention to aggregate information from current and past frames.

Following these spatio-temporal interactions, the transformer outputs visual tokens $\mathbf{Z}_t^{\text{vis}}$ and pose tokens $\mathbf{Z}_t^{\text{pose}}$ for current geometry reconstruction. During planning, the output planning tokens $\mathbf{Z}_t^{\text{plan}}$ provide scene features for trajectory prediction. To process sequential inputs, **DVGT-World** employs a window-based streaming mechanism. During streaming inference, the historical cache $\mathbf{C}_{t-1}$ stores intermediate features from the past w frames at each temporal attention layer. The cache is updated with the current frame's features in a First-In-First-Out (FIFO) manner, discarding the oldest entries to maintain the history window. This window-based streaming mechanism avoids redundant computations for past frames and enables efficient inference.

3.4 Horizon-conditioned Future Geometry Prediction

Since trajectory planning spans multiple future steps, the model needs to capture the 3D scene evolution over different horizons. To realize this, **DVGT-World** first appends a set of learnable future queries $\mathbf{Q}$ after the current observation tokens $\mathbf{F}_t$. These queries follow the same multi-view token layout as an observed frame. To support future prediction across varying time horizons, the target future horizon h is encoded via a time embedding and added to the future queries:

$$\mathbf{Q}_{t+h} = \mathbf{Q} + \text{Embed}(h).$$

The current visual tokens $\mathbf{F}_t$ and horizon-conditioned future queries $\mathbf{Q}_{t+h}$ are then processed jointly through the geometry transformer $\mathcal{G}$ to obtain the current and future scene representations:

$$(\mathbf{Z}_t, \mathbf{Z}_{t+h}) = \mathcal{G}([\mathbf{F}_t; \mathbf{Q}_{t+h}], \mathbf{C}_{t-1}).$$

Intra-view and cross-view attention operate separately at each timestep. Temporal causal attention allows future queries to attend to current tokens and cached historical features, while preventing observed tokens from attending to future queries.

The resulting future representations $\mathbf{Z}_{t+h}$ are then decoupled into future visual tokens $\mathbf{Z}_{t+h}^{\text{vis}}$ and future pose tokens $\mathbf{Z}_{t+h}^{\text{pose}}$ to predict future dense pointmaps and relative ego-poses, using the same prediction heads as current geometry reconstruction. Future geometry supervision during pretraining encourages the shared geometry transformer to learn 3D scene evolution, providing representations for downstream planning. Since current features do not depend on future queries, the future prediction branch can be disabled during planning fine-tuning and inference, avoiding the additional cost of future geometry prediction.

3.5 Training Details

We train **DVGT-World** in two stages. First, we pretrain the 3D world model on OpenScene using current geometry reconstruction and future geometry prediction. For each observation sequence, we randomly sample a target horizon $h \in \{1, \ldots, 8\}$. All observed frames are processed in parallel with temporal causal masking, allowing each frame to attend to its own and earlier observations. Following DVGT-2 (Zuo et al., 2026), the pointmap and pose objectives are defined as

$$\mathcal{L}_{\text{point}} = \mathcal{L}_{\text{reg}} + \mathcal{L}_{\text{conf}} + \mathcal{L}_{\text{normal}}, \qquad \mathcal{L}_{\text{pose}} = \mathcal{L}_{\text{trans}} + \mathcal{L}_{\text{rot}}, \tag{1}$$

where $\mathcal{L}_{\text{reg}}$ is the ℓ_2 point regression loss, $\mathcal{L}_{\text{conf}}$ is the confidence-aware regression loss, and $\mathcal{L}_{\text{normal}}$ enforces multi-scale surface-normal consistency. $\mathcal{L}_{\text{trans}}$ and $\mathcal{L}_{\text{rot}}$ apply ℓ_1 supervision to the relative translation and rotation quaternion, respectively. The overall geometry pretraining objective is

$$\mathcal{L}_{\text{world}} = \mathcal{L}_{\text{point}}^{\text{cur}} + \mathcal{L}_{\text{point}}^{\text{fut}} + \lambda_p \left(\mathcal{L}_{\text{pose}}^{\text{cur}} + \mathcal{L}_{\text{pose}}^{\text{fut}} \right), \tag{2}$$

where cur and fut denote reconstruction of observed frames and prediction of the sampled future frame, respectively, and λ_p weights the pose losses.

Table 1: **Planning results on NAVSIM v1 `navtest`**. C and L denote camera and LiDAR.

Method	Input	NC ↑	DAC ↑	TTC ↑	Comf. ↑	EP ↑	PDMS ↑
Perception-based Methods							
VADv2 (Chen et al., 2024)	C	97.2	89.1	91.6	100	76.0	80.9
UniAD (Hu et al., 2023)	C	97.8	91.9	92.9	100	78.8	83.4
Transfuser (Chitta et al., 2022)	C & L	97.7	92.8	92.8	100	79.2	84.0
DiffusionDrive (Liao et al., 2025)	C & L	98.2	96.2	94.7	100	82.2	88.1
WoTE (Li et al., 2025c)	C & L	98.5	96.8	94.9	99.9	81.9	88.3
DriveSuprim (Yao et al., 2025)	C & L	97.8	97.3	93.6	100	86.7	89.9
VLA-based Methods							
AdaThinkDrive (Luo et al., 2025)	C	98.5	94.4	94.9	100	79.9	86.2
AutoVLA (Zhou et al., 2025b)	C	98.4	95.6	98.0	99.9	85.9	89.1
DriveDreamer-Policy (Zhou et al., 2026)	C	98.4	97.1	95.1	100	83.5	89.2
ReCogDrive (Li et al., 2025d)	C	98.2	97.8	95.2	99.8	83.5	89.6
DriveVLA-W0 (Li et al., 2025b)	C	98.7	99.1	95.3	99.3	83.3	90.2
ExploreVLA (Sheng et al., 2026)	C	98.8	98.4	96.5	99.9	83.5	90.4
DriveMA (Zheng et al., 2026)	C	98.7	97.8	95.5	99.0	86.7	91.2
DynVLA (Shang et al., 2026)	C	98.6	98.7	95.5	100	86.8	91.7
WAM-based Methods							
LAW (Li et al., 2024a)	C	96.4	95.4	88.7	99.0	81.7	84.6
Epona (Zhang et al., 2025)	C	97.9	95.1	93.8	99.0	80.4	86.2
PWM (Zhao et al., 2026)	C	98.6	95.9	95.4	100	81.8	88.1
DriveLaW (Xia et al., 2026)	C	99.0	97.1	96.7	100	81.3	89.1
DriveWAM (Shi et al., 2026)	C	98.3	98.1	95.2	100	84.3	90.1
EponaV2 (Xu et al., 2026)	C	98.6	97.9	95.7	100	84.8	90.4
DriveVA (Liu et al., 2026)	C	99.2	97.5	98.7	100	83.5	90.9
VGA-based Methods							
DVGT-2 (Zuo et al., 2026)	C	98.7	97.9	95.8	100	84.3	90.3
LiAuto-GeoX (Lian et al., 2026)	C	98.6	97.5	95.2	100	85.8	90.6
DVGT-World	C	99.2	98.1	96.8	100	89.9	**93.3**

Table 2: **Planning results on NAVSIM v2 `navtest` split**.

Method	NC ↑	DAC ↑	DDC ↑	TL ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
Ego Status MLP	93.1	77.9	92.7	99.6	86.0	91.5	89.4	98.3	85.4	64.0
Transfuser (Chitta et al., 2022)	96.9	89.9	97.8	99.7	87.1	95.4	92.7	98.3	87.2	76.7
Hydra-MDP++ (Li et al., 2025a)	97.2	97.5	99.4	99.6	83.1	96.5	94.4	98.2	70.9	81.4
DriveSuprim (Yao et al., 2025)	97.5	96.5	99.4	99.6	88.4	96.6	95.5	98.3	77.0	83.1
ARTEMIS (Feng et al., 2025)	98.3	95.1	98.6	99.8	81.5	97.4	96.5	98.3	-	83.1
DiffusionDrive (Liao et al., 2025)	98.2	95.9	99.4	99.8	87.5	97.3	96.8	98.3	87.7	84.5
DriveVLA-W0 (Li et al., 2025b)	98.5	99.1	98.0	99.7	86.4	98.1	93.2	97.9	58.9	86.1
DriveLaW (Xia et al., 2026)	98.7	96.9	99.6	99.8	87.5	98.3	97.6	98.4	77.4	88.6
ExploreVLA (Sheng et al., 2026)	98.8	96.2	99.6	99.8	87.1	98.2	97.8	98.3	86.8	88.8
EponaV2 (Xu et al., 2026)	98.5	97.4	99.5	99.9	87.9	98.1	97.7	98.2	77.4	88.9
DVGT-2 (Zuo et al., 2026)	98.7	97.9	99.7	99.9	87.9	98.0	98.2	98.2	77.0	89.6
DVGT-World	99.2	98.1	99.5	99.6	92.4	98.8	97.3	97.8	73.3	**90.7**

In the second stage, we fine-tune the pretrained model for planning on NAVSIM `navtrain` using LoRA (Hu et al., 2021). We freeze the visual encoder and the original geometry-transformer parameters, while optimizing the LoRA adapters, planning tokens, and planning module. The planner is trained with a best-of-K ℓ_1 loss $\mathcal{L}_{\text{traj}}$ over K candidate trajectories. NAVSIM simulator provides supervision for collision avoidance, drivable-area compliance, time-to-collision, progress, comfort, and driving-direction compliance. We train the scorer with binary cross-entropy loss $\mathcal{L}_{\text{score}}$, with candidate trajectories detached before being passed to the scorer. The final planning objective is

$$\mathcal{L}_{\text{planning}} = \mathcal{L}_{\text{traj}} + \lambda_s \mathcal{L}_{\text{score}}, \tag{3}$$

where λ_s weights the scoring loss. The scorer learns the NAVSIM v1 sub-metrics from simulator evaluations of the candidate trajectories. For trajectory selection, it combines predicted NC and DAC multiplicatively with a weighted sum of TTC, EP, and comfort, using weights of 5, 5, and 2, respectively. The candidate with the highest score is selected as the final trajectory.

Table 3: **Effect of future geometry supervision on NAVSIM v1 `navtest`.**

Future supervision (frame offsets)	NC ↑	DAC ↑	TTC ↑	Comf. ↑	EP ↑	PDMS ↑
No Future	98.2	97.5	94.5	100	85.1	89.9
Future +1	99.3	97.0	97.5	100	85.2	90.9
Future +1–2	99.4	97.9	97.6	100	86.7	92.2
Future +1–4	99.5	98.2	97.8	100	87.4	92.7
Future +1–8	99.3	98.0	97.4	100	88.8	**93.0**

Table 4: **Front-view future pointmap prediction on NAVSIM `navtest`.**

Method	Acc ↓					Comp ↓				
	1 s	2 s	3 s	4 s	Avg.	1 s	2 s	3 s	4 s	Avg.
GT Image + DVGT-2	0.396	0.390	0.386	0.382	0.389	0.433	0.428	0.426	0.422	0.428
GT-Pose + DVGT-2	0.838	1.212	1.478	1.652	1.295	0.742	1.197	1.791	2.671	1.600
WAN2.2 + DVGT-2	0.749	0.909	1.052	1.168	0.969	2.911	3.308	3.692	4.399	3.578
DVGT-World	**0.453**	**0.558**	**0.649**	**0.712**	**0.593**	**0.633**	**0.831**	**0.985**	**1.176**	**0.906**

4 Experiments

4.1 Experimental Setup

Datasets and Metrics. We pretrain **DVGT-World** on OpenScene with current and future geometry supervision, then fine-tune it for planning on NAVSIM `navtrain`. Following DVGT Zuo et al. (2025), pointmap supervision is derived from aligned and filtered MoGe-2 Wang et al. (2025b) depth estimates. On NAVSIM `navtest`, we evaluate closed-loop planning with PDMS Dauner et al. (2024) for v1 and EPDMS Cao et al. (2025) for v2. We evaluate future geometry using Accuracy (Acc), Completeness (Comp), and Chamfer distance for pointmaps, Abs Rel and $\delta < 1.25$ for depth, and AUC@30 for ego-pose estimation. Metric definitions are provided in Section B.

Training. We initialize **DVGT-World** from DVGT-2 Zuo et al. (2026) weights and pretrain it with geometry supervision on 32 GPUs for 100 epochs. We then fine-tune it for planning on 8 GPUs for 40 epochs. All ablation variants in Tab. 3 are pretrained with geometry supervision for 100 epochs and fine-tuned for planning for 20 epochs, using 8 GPUs in both stages.

Evaluation. For planning, the fine-tuned model predicts future ego trajectories from four observed frames across eight cameras and the current ego status. For geometry evaluation, the pretrained model predicts future pointmaps and ego poses at each target horizon from four observed frames.

4.2 Planning Performance

NAVSIM v1. As shown in Tab. 1, **DVGT-World** achieves 93.3 PDMS on NAVSIM v1, surpassing the compared VLA, WAM, and VGA models. Its NC is tied for the highest, and its TTC exceeds most baselines, indicating strong collision avoidance. DriveVA achieves higher TTC but substantially lower EP (83.5 vs. 89.9), while **DVGT-World** combines strong collision avoidance with the highest route progress score. It also achieves higher DAC than DVGT-2 and LiAuto-GeoX, indicating better drivable-area compliance. Compared with DVGT-2, **DVGT-World** extends geometry pretraining to future scene prediction and improves PDMS by 3.0 points. These results suggest that future geometry supervision helps learn scene representations for safe and efficient planning.

NAVSIM v2. As shown in Tab. 2, **DVGT-World** achieves 90.7 EPDMS on NAVSIM v2, the highest among the compared models. Compared with DVGT-2, **DVGT-World** improves collision avoidance (NC and TTC) and drivable-area compliance (DAC), while increasing EP from 87.9 to 92.4. EC decreases from 77.0 to 73.3 and LK from 98.2 to 97.3, which may reflect trade-offs between greater route progress, driving comfort, and lane keeping. Overall, the 1.1-point gain in EPDMS supports the benefit of future geometry pretraining for planning under the broader v2 criteria.



Figure 3: **Qualitative results of DVGT-World.**

Table 5: **Multi-view geometry reconstruction and prediction on NAVSIM `navtest`.**

Frame	Acc ↓	Comp ↓	Chamfer ↓	Abs Rel ↓	$\delta < 1.25$ ↑	AUC@30 ↑
History avg.	0.411	0.415	0.826	0.047	97.0	92.24
Future 1 s	0.497	0.663	1.160	0.117	89.8	95.89
Future 2 s	0.599	0.844	1.443	0.157	85.5	95.63
Future 3 s	0.766	1.066	1.831	0.192	81.8	94.06
Future 4 s	0.998	1.395	2.393	0.224	78.4	92.41
Future avg.	0.715	0.992	1.707	0.173	83.9	94.50

Visualizations. Fig. 3 presents qualitative results of joint geometry modeling and planning.

4.3 Effect of Future Geometry Supervision

We compare variants with and without future geometry supervision under the same training protocol. As shown in Tab. 3, the Future +1–8 variant improves PDMS from 89.9 to 93.0 over the No Future baseline, with better collision avoidance (NC and TTC) and greater route progress (EP). These gains suggest that predicting 3D scene evolution during pretraining helps learn representations for downstream planning. Future geometry prediction is disabled during planning, avoiding its additional computational overhead at inference.

We further study how the future prediction range during pretraining affects planning performance. For each training sample, we randomly select one target frame from the specified range and supervise its geometry prediction. Expanding the range from +1 to +1–8 frames improves PDMS from 90.9 to 93.0, with diminishing gains as the range grows. Extending +1–4 to +1–8 yields a further 0.3-point PDMS gain, with higher EP but slightly lower NC, DAC, and TTC. The highest PDMS is obtained when supervision spans the full eight-frame (4 s) planning horizon, suggesting that learning longer-term scene evolution can benefit trajectory planning.

4.4 Future Geometry Prediction

Baseline Comparison. We compare **DVGT-World** with two baselines for future geometry prediction: *GT-Pose + DVGT-2* transforms DVGT-2's current pointmap into future ego coordinates using ground-truth poses; *WAN2.2 + DVGT-2* reconstructs pointmaps from future images generated by WAN2.2 Wan et al. (2025), which is pretrained on OpenScene and fine-tuned on NAVSIM. *GT Image + DVGT-2* serves as an oracle reference by reconstructing pointmaps from ground-truth future images. Since our WAN2.2 baseline generates only front-view images, this comparison evaluates front-view pointmaps at 1–4 s on NAVSIM `navtest`. Further details are provided in Section C.

Table 6: Inference efficiency of VLA, WAM, and VGA methods on an NVIDIA A800 GPU.

Method	Resolution	Views	Latency (ms) ↓	Memory (GB) ↓	FLOPs (T) ↓
VLA-based methods					
DriveVLA-W0 (Li et al., 2025b)	144×256	1	1,390	19.5	209.9
DriveDreamer-Policy (Zhou et al., 2026)	288×512	8	465	10.3	35.4
DynVLA (Shang et al., 2026)	288×512	1	1,438	23.8	104.9
AutoVLA (Zhou et al., 2025b)	280×504	3	12,171	7.9	18.3
WAM-based methods					
DriveVA (Liu et al., 2026)	288×512	1	256	23.4	25.6
Epona (Zhang et al., 2025)	288×512	1	374	14.5	7.0
DriveLaW (Xia et al., 2026)	288×512	1	589	16.4	10.2
DriveWAM (Shi et al., 2026)	288×512	1	1,107	26.2	19.3
VGA-based methods					
DVGT-2 (Zuo et al., 2026)	288×512	8	256	15.8	16.5
DVGT-World (future on)	288×512	8	414	14.5	27.9
DVGT-World (future off)	288×512	8	**222**	**11.5**	**15.2**

As shown in Tab. 4, **DVGT-World** achieves lower Acc and Comp than both baselines at every horizon. At 1 s, its Acc approaches the oracle reference (0.453 vs. 0.396 m), indicating accurate prediction of future scene geometry. It also provides more complete scene coverage, with an average Comp of 0.906 m, compared with 1.600 m for GT-Pose + DVGT-2 and 3.578 m for WAN2.2 + DVGT-2. These gains show that **DVGT-World** predicts future scene structure more accurately and completely than either transforming observed geometry or reconstructing generated images, although a coverage gap to the oracle remains.

Multi-view Prediction. Tab. 5 further evaluates this predictive capability across all eight views and the full 4 s planning horizon. At 1 s, pointmap prediction remains close to historical reconstruction in Acc (0.497 vs. 0.411). At 4 s, depth prediction still achieves 78.4% on $\delta < 1.25$, while ego-pose AUC@30 remains above 92 at every evaluated horizon. These results demonstrate the model's ability to predict both scene geometry and ego motion beyond the observed frames. Together with the planning ablations in Tab. 3, they suggest that learning to predict 3D scene evolution provides a geometric foundation for planning. Results for all eight future frames are provided in Section D.

4.5 Efficiency Comparison

We compare publicly available VLA, WAM, and VGA methods on the same NVIDIA A800 GPU. All measurements use a batch size of 1, 10 warm-up runs, and 100 synchronized timed runs. We use 288×512 images from 8 views when supported; otherwise, we report the supported resolution and view count in Tab. 6. All VGA models use streaming inference. As shown in Tab. 6, **DVGT-World** without future prediction achieves the highest efficiency among the compared methods, including latency, memory, and FLOPs. Enabling future queries increases latency to 414 ms, showing the additional cost of future prediction. Together with the planning results, these results show that **DVGT-World** achieves strong planning performance with high inference efficiency.

5 Conclusion

In this paper, we present **DVGT-World**, which extends the streaming Vision-Geometry-Action paradigm from observed geometry reconstruction to predictive 3D world modeling. We introduce horizon-conditioned future queries into the geometry transformer to predict dense pointmaps and ego poses from historical multi-view observations. Through joint supervision of current and future geometry, **DVGT-World** learns representations of 3D scene structure and evolution for downstream planning. The causal attention design allows the future prediction branch to be disabled during planning, avoiding its additional inference cost. Experiments on NAVSIM demonstrate accurate multi-view future geometry prediction and strong closed-loop planning performance, validating the benefit of future geometry supervision for planning. We hope this work encourages further exploration of predictive 3D world models as a geometric foundation for autonomous driving.

AI USE STATEMENT

In this work, we used generative AI tools to assist with manuscript revision, figure preparation, and experimental results integration. We have also used generative AI tools for LaTeX template migration and formatting. The authors reviewed and revised the AI-assisted content. We take full responsibility for the final content of this work, including all text, claims, and artifacts produced with AI assistance.

REFERENCES

Wei Cao, Marcel Hallgarten, Tianyu Li, Daniel Dauner, Xunjiang Gu, Caojun Wang, Yakov Miron, Marco Aiello, Hongyang Li, Igor Gilitschenski, et al. Pseudo-simulation for autonomous driving. *arXiv preprint arXiv:2506.04218*, 2025.

Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024.

Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *TPAMI*, 45 (11):12878–12895, 2022.

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

Renju Feng, Ning Xi, Duanfeng Chu, Rukang Wang, Zejian Deng, Anzheng Wang, Liping Lu, Jinxiang Wang, and Yanjun Huang. Artemis: Autoregressive end-to-end trajectory planning with mixture of experts for autonomous driving. *arXiv preprint arXiv:2504.19580*, 2025.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *CVPR*, pp. 17853–17862, 2023.

Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *CVPR*, pp. 9223–9232, 2023.

Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. In *ECCV*, pp. 376–393. Springer, 2024.

Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, et al. Emma: End-to-end multimodal model for autonomous driving. *arXiv preprint arXiv:2410.23262*, 2024.

Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *ICCV*, pp. 8340–8350, 2023.

Kailin Li, Zhenxin Li, Shiyi Lan, Yuan Xie, Zhizhong Zhang, Jiayi Liu, Zuxuan Wu, Zhiding Yu, and Jose M Alvarez. Hydra-mdp++: Advancing end-to-end driving via expert-guided hydra-distillation. *arXiv preprint arXiv:2503.12820*, 2025a.

Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. *arXiv preprint arXiv:2406.08481*, 2024a.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, et al. Drivevla-w0: World models amplify data scaling law in autonomous driving. *arXiv preprint arXiv:2510.12796*, 2025b.

Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. *arXiv preprint arXiv:2504.01941*, 2025c.

Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *AAAI*, volume 37, pp. 1477–1485, 2023.

Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, et al. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. *arXiv preprint arXiv:2506.08052*, 2025d.

Zhenxin Li, Wenhao Yao, Zi Wang, Xinglong Sun, Joshua Chen, Nadine Chang, Maying Shen, Zuxuan Wu, Shiyi Lan, and Jose M Alvarez. Generalized trajectory scoring for end-to-end multimodal planning. *arXiv preprint arXiv:2506.06664*, 2025e.

Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *TPAMI*, 2024b.

Jiawei Lian, Haoyi Sun, Yang Wu, Lifu Mu, Siyuan Wang, Le Hui, Ning Mao, Tao Wei, Pan Zhou, Kun Zhan, et al. Liauto-geox: Efficient grounded driving transformer. *arXiv preprint arXiv:2606.05774*, 2026.

Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *CVPR*, pp. 12037–12047, 2025.

Mengmeng Liu, Diankun Zhang, Jiuming Liu, Jianfeng Cui, Hongwei Xie, Guang Chen, Hangjun Ye, Michael Ying Yang, Francesco Nex, and Hao Cheng. Driveva: Video action models are zero-shot drivers. *arXiv preprint arXiv:2604.04198*, 2026.

Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pp. 2774–2781. IEEE, 2023.

Yuechen Luo, Fang Li, Shaoqing Xu, Zhiyi Lai, Lei Yang, Qimao Chen, Ziang Luo, Zixun Xie, Shengyin Jiang, Jiaxin Liu, et al. Adathinkdrive: Adaptive thinking via reinforcement learning for autonomous driving. *arXiv preprint arXiv:2509.13769*, 2025.

Shuyao Shang, Bing Zhan, Yunfei Yan, Yuqi Wang, Yingyan Li, Yasong An, Xiaoman Wang, Jierui Liu, Lu Hou, Lue Fan, et al. Dynvla: Learning world dynamics for action reasoning in autonomous driving. *arXiv preprint arXiv:2603.11041*, 2026.

Zihao Sheng, Xin Ye, Jingru Luo, Sikai Chen, and Liu Ren. Explorevla: Dense world modeling and exploration for end-to-end autonomous driving. *arXiv preprint arXiv:2604.02714*, 2026.

Chen Shi, Jinrui Xu, Shaoshuai Shi, Kehua Sheng, Bo Zhang, and Li Jiang. Drivewam: Video generative priors enable scalable world-action modeling for autonomous driving. *arXiv preprint arXiv:2605.28544*, 2026.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

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 *CVPR*, pp. 5294–5306, 2025a.

Linbo Wang, Yupeng Zheng, Qiang Chen, Shiwei Li, Yichen Zhang, Zebin Xing, Qichao Zhang, Xiang Li, Deheng Qian, Pengxuan Yang, et al. Latent-wam: Latent world action modeling for end-to-end autonomous driving. *arXiv preprint arXiv:2603.24581*, 2026.

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025b.

Yifan Wang, Jianjun Zhou, Haoyi Zhu, Wenzheng Chang, Yang Zhou, Zizun Li, Junyi Chen, Jiangmiao Pang, Chunhua Shen, and Tong He. π^3: Permutation-equivariant visual geometry learning. *arXiv preprint arXiv:2507.13347*, 2025c.

Tianze Xia, Yongkang Li, Lijun Zhou, Jingfeng Yao, Kaixin Xiong, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, et al. Drivelaw: Unifying planning and video generation in a latent driving world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39701–39712, 2026.

Jiawei Xu, Zhizhou Zhong, Zhijian Shu, Mingkai Jia, Mingxiao Li, Jia-Wang Bian, Qian Zhang, Kaicheng Zhang, Jin Xie, Jian Yang, et al. Eponav2: Driving world model with comprehensive future reasoning. *arXiv preprint arXiv:2605.14696*, 2026.

Jin Yao, Dhruva Dixith Kurra, Tom Lampo, Zezhou Cheng, Danhua Guo, and Burhan Yaman. Vlga: Vision-language-geometry-action models for autonomous driving. *arXiv preprint arXiv:2606.12396*, 2026.

Wenhao Yao, Zhenxin Li, Shiyi Lan, Zi Wang, Xinglong Sun, Jose M Alvarez, and Zuxuan Wu. Drivesuprim: Towards precise trajectory selection for end-to-end planning. *arXiv preprint arXiv:2506.06659*, 2025.

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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27220–27230, 2025.

Zhida Zhao, Talas Fu, Yifan Wang, Lijun Wang, and Huchuan Lu. From forecasting to planning: Policy world model for collaborative state-action prediction. *Advances in Neural Information Processing Systems*, 38:134585–134611, 2026.

Weicheng Zheng, Yixin Huang, Qiao Sun, Derun Li, and Hang Zhao. Drivema: Driving vision-language-action models with verifiable meta-actions. *arXiv preprint arXiv:2605.31271*, 2026.

Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *ECCV*, pp. 55–72. Springer, 2024a.

Wenzhao Zheng, Ruiqi Song, Xianda Guo, Chenming Zhang, and Long Chen. Genad: Generative end-to-end autonomous driving. In *ECCV*, pp. 87–104. Springer, 2024b.

Wenzhao Zheng, Junjie Wu, Yao Zheng, Sicheng Zuo, Zixun Xie, Longchao Yang, Yong Pan, Zhihui Hao, Peng Jia, Xianpeng Lang, et al. Gaussianad: Gaussian-centric end-to-end autonomous driving. *arXiv preprint arXiv:2412.10371*, 2024c.

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

Yupeng Zheng, Pengxuan Yang, Zebin Xing, Qichao Zhang, Yuhang Zheng, Yinfeng Gao, Pengfei Li, Teng Zhang, Zhongpu Xia, Peng Jia, et al. World4drive: End-to-end autonomous driving via intention-aware physical latent world model. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 28632–28642. IEEE, 2025.

Xingcheng Zhou, Xuyuan Han, Feng Yang, Yunpu Ma, and Alois C Knoll. Opendrivevla: Towards end-to-end autonomous driving with large vision language action model. *arXiv preprint arXiv:2503.23463*, 2025a.

Yang Zhou, Xiaofeng Wang, Hao Shao, Letian Wang, Guosheng Zhao, Jiangnan Shao, Jiagang Zhu, Tingdong Yu, Zheng Zhu, Guan Huang, et al. Drivedreamer-policy: A geometry-grounded world-action model for unified generation and planning. *arXiv preprint arXiv:2604.01765*, 2026.

Zewei Zhou, Tianhui Cai, Seth Z Zhao, Yun Zhang, Zhiyu Huang, Bolei Zhou, and Jiaqi Ma. Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. *arXiv preprint arXiv:2506.13757*, 2025b.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Shengyin Jiang, Long Chen, Zhi-Xin Yang, and Jiwen Lu. Dvgt: Driving visual geometry transformer. *arXiv preprint arXiv:2512.16919*, 2025.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Hanbing Li, Long Chen, Zhi-Xin Yang, and Jiwen Lu. Dvgt-2: Vision-geometry-action model for autonomous driving at scale. *arXiv preprint arXiv:2604.00813*, 2026.

A Implementation Details

Architecture. We use a frozen DINOv3 (Siméoni et al., 2025) ViT-L visual encoder and a geometry transformer with 24 blocks, a feature dimension of 1024, and 16 attention heads, following DVGT-2 (Zuo et al., 2026). Current and future geometry share the DPT pointmap head and pose regression head. The target horizon is encoded by a learned embedding table with eight entries. For planning, we add eight learnable tokens per view and project their output features to 256 dimensions. Based on planning features, the planner produces 64 candidate trajectories, each containing eight (x, y, yaw) waypoints. The scorer selects the final trajectory with the highest predicted driving score.

Input Sampling. Images are processed at 512×288 resolution, with sequences sampled at 2 Hz. During geometry pretraining, we sample 3–8 camera views and at least 4 observed frames per sequence, with the sequence length and batch size constrained by a budget of 40 observed images per GPU. One future target horizon is randomly sampled from the specified prediction range for each sequence. Planning fine-tuning uses 4 observed frames from 8 cameras and a per-GPU batch size of 4. Pointmap supervision follows the aligned and filtered MoGe-2 (Wang et al., 2025b) depth annotations used by DVGT (Zuo et al., 2025) and DVGT-2 (Zuo et al., 2026).

Training. We initialize the model from DVGT-2 weights and use NVIDIA H20 GPUs for all training stages. The main model is pretrained for 100 epochs on 32 GPUs and fine-tuned for 40 epochs on 8 GPUs; all ablations use 8 GPUs for 100 pretraining epochs and 20 fine-tuning epochs. We use AdamW with a peak learning rate of 10^{-4} and weight decay of 0.05 in both stages. The learning rate adopts a linear warm-up and then follows a cosine decay to 10^{-8}. We use bfloat16 precision and gradient norm clipping at 1.0. During planning fine-tuning, LoRA adapters with rank 16 and $\alpha = 32$ are applied to the query, key, and value projections of the geometry transformer. The pose and scoring loss weights are $\lambda_p = 5$ and $\lambda_s = 1$, respectively.

B Evaluation Metrics

Closed-loop Planning. We use the official default scoring configurations of NAVSIM v1 (Dauner et al., 2024) and v2 (Cao et al., 2025), reporting PDMS and EPDMS, respectively. NC and TTC evaluate at-fault collisions and near-term collision risk, while DAC measures drivable-area compliance. EP measures normalized progress along the route, and Comf. evaluates motion smoothness. PDMS combines NC and DAC multiplicatively with a weighted average of TTC, EP, and Comf., using weights of 5, 5, and 2, respectively. EPDMS additionally evaluates driving-direction compliance (DDC), traffic-light compliance (TL), and lane keeping (LK). It assesses comfort through history comfort (HC), which accounts for the transition from observed motion to the planned trajectory, and extended comfort (EC), which measures consistency between plans from successive observations. All planning scores are reported on a 0–100 scale, with higher values indicating better performance.

Pointmap Evaluation. Given valid predicted and ground-truth pointmaps $\mathcal{P}$ and $\mathcal{G}$, Accuracy (Acc) and Completeness (Comp) measure nearest-neighbor distances in the two directions:

$$\mathrm{Acc} = \frac{1}{|\mathcal{P}|} \sum_{\mathbf{p} \in \mathcal{P}} \min_{\mathbf{g} \in \mathcal{G}} \|\mathbf{p} - \mathbf{g}\|_2, \qquad \mathrm{Comp} = \frac{1}{|\mathcal{G}|} \sum_{\mathbf{g} \in \mathcal{G}} \min_{\mathbf{p} \in \mathcal{P}} \|\mathbf{g} - \mathbf{p}\|_2. \tag{4}$$

Chamfer distance is their sum, $\mathrm{Chamfer} = \mathrm{Acc} + \mathrm{Comp}$. All three metrics are reported in meters, with lower values indicating better results. The baseline comparison in Tab. 4 evaluates pointmaps in the corresponding future ego coordinate system. For the multi-view results in Tabs. 5 and 7, predicted and ground-truth pointmaps are transformed into the first observed frame's ego coordinate system using their respective ego poses. We apply the reference validity masks and compute distances separately for each view and frame, averaging within each sample and then across samples.

Ray Depth Estimation. Following DVGT-2 (Zuo et al., 2026), ray depth is the Euclidean distance from each point to the ego-vehicle center at the corresponding timestep. For the predicted and ground-truth ray depths $\hat{d}_u$ and d_u at valid pixels $u \in \Omega$, we compute

$$\mathrm{Abs\ Rel} = \frac{1}{|\Omega|} \sum_{u \in \Omega} \frac{|\hat{d}_u - d_u|}{d_u}, \qquad \delta_{1.25} = \frac{1}{|\Omega|} \sum_{u \in \Omega} \mathbb{I}\left[\max\left(\frac{\hat{d}_u}{d_u}, \frac{d_u}{\hat{d}_u} \right) < 1.25 \right], \tag{5}$$

where $\mathbb{I}[\cdot]$ is the indicator function. We average these metrics across samples and report $\delta_{1.25}$ as $\delta < 1.25$ in percent. Lower Abs Rel and higher $\delta < 1.25$ indicate better depth estimates.

Ego-Pose Estimation. We compute relative rotation and translation-direction angular errors and take their maximum as the pose error. AUC@30 measures the normalized area under the accuracy–threshold curve up to $30°$, reported as a percentage with higher values indicating better performance. Historical reconstruction is evaluated over pairs of observed frames, while each future pose is evaluated relative to the first observed frame. The resulting AUC scores are averaged across samples.

Table 7: **Multi-view geometry prediction at all eight future frames on NAVSIM v1 `navtest`.** The history row summarizes four input frames. Future avg. averages all eight future horizons.

Frame	Acc ↓	Comp ↓	Chamfer ↓	Abs Rel ↓	$\delta < 1.25$ ↑	AUC@30 ↑
History avg.	0.411	0.415	0.826	0.047	97.0	92.24
Future +1	0.463	0.573	1.036	0.093	92.2	94.41
Future +2	0.497	0.663	1.160	0.117	89.8	95.89
Future +3	0.540	0.751	1.291	0.137	87.6	96.14
Future +4	0.599	0.844	1.443	0.157	85.5	95.63
Future +5	0.673	0.946	1.619	0.175	83.7	94.90
Future +6	0.766	1.066	1.831	0.192	81.8	94.06
Future +7	0.873	1.218	2.091	0.209	80.2	93.21
Future +8	0.998	1.395	2.393	0.224	78.4	92.41
Future avg.	0.676	0.932	1.608	0.163	84.9	94.58

C Future Geometry Baselines

Evaluation Protocol. The comparison in Tab. 4 evaluates front-view pointmaps at 1–4 s, since our WAN2.2 baseline generates only front-view images. All methods use the same 4 observed front-view frames as historical input, whereas the multi-view evaluation in Tab. 5 uses all 8 camera views. The three baselines share the same pretrained DVGT-2 checkpoint, and the metrics follow Section B. Of the 12,146 `navtest` samples, 12,095 provide complete front-view history and future frames together with valid WAN2.2 predictions. All rows are reported on their common subset of 11,539 samples, on which every baseline yields valid predictions at all four horizons.

GT-Pose + DVGT-2. DVGT-2 reconstructs the current front-view pointmap from the 4 observed frames. The predicted 3D points are transformed from the current ego coordinate system to each future ego coordinate system using the ground-truth relative ego-poses and filtered to those that project inside the future front-view image with positive depth.

WAN2.2 + DVGT-2. We adapt Wan2.2-TI2V-5B (Wan et al., 2025) for front-view driving video prediction. The generator is pretrained on OpenScene and fine-tuned on NAVSIM, and is conditioned on 13 history frames of the 10 Hz front-view video, ego states, navigation commands, and past trajectories, without access to future trajectories. The history frames are encoded by the VAE without added noise as conditioning latents, and the remaining conditions are each projected into the DiT through an MLP. The model generates 40 future frames at 10 Hz and a resolution of 672×384. Training uses 8 H20 GPUs for 100K iterations with a learning rate of 10^{-4}. At test time, we subsample the future frames at 2 Hz (0.5–4.0 s) from each generated video, rescale them to the native annotation resolution of 1920×1080, and process them with the same image pipeline to match the DVGT input resolution. The frozen DVGT-2 then reconstructs future pointmaps from the 4 observed frames and the 8 generated future frames.

GT Image + DVGT-2. As an oracle reference, DVGT-2 reconstructs pointmaps from the 4 observed frames together with the 8 ground-truth future front-view images. The model receives no future poses or pointmaps; the ground truth is used only for evaluation. This row serves as an oracle reference for the upper bound of future geometry prediction performance when ground-truth future observations are available.

D Additional Geometry Results

Tab. 7 reports future geometry prediction at all 8 horizons, sampled every 0.5 s over 4 s. Pointmap accuracy and completeness gradually decrease as the prediction horizon grows: Acc and Comp increase from 0.463 m and 0.573 m at 0.5 s to 0.998 m and 1.395 m at 4 s, respectively. Depth estimation follows the same trend, with $\delta < 1.25$ decreasing from 92.2% to 78.4%. Despite the increasing difficulty of longer-horizon prediction, ego-pose AUC@30 remains above 92% at every horizon. These intermediate-frame results complement the 1–4 s evaluation in Tab. 5 and show how prediction quality evolves across the full planning horizon.