# METIS: A GENERALIZABLE AND EFFICIENT 4D WORLD-ACTION MODEL FOR AUTONOMOUS DRIVING AND URBAN NAVIGATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

World-Action Models (WAMs) jointly model future observations and actions, but existing approaches often require explicit future observation generation during action inference, introducing substantial inference overhead. Moreover, future states are primarily modeled in semantic or 2D visual spaces, leaving their spatial structure insufficiently grounded for reliable planning. We introduce **Metis**, an end-to-end WAM that jointly learns future observation, action, and spatial geometry prediction during training, while requiring only action prediction at inference time. Metis employs specialized video generation and action experts coupled through asymmetric attention, allowing future video representations to attend to action representations while masking the reverse dependency. This design preserves supervision from future world modeling without requiring stochastic future video generation for action inference. We further introduce training-only spatial registers with complementary depth and occupancy supervision to enhance spatially grounded action representations without additional inference overhead. Extensive experiments demonstrate strong performance on NAVSIM-v2 and CityWalker, together with substantially reduced action-inference latency compared with full world-action generation. Zero-shot real-world deployment further confirms the practical feasibility of our approach.

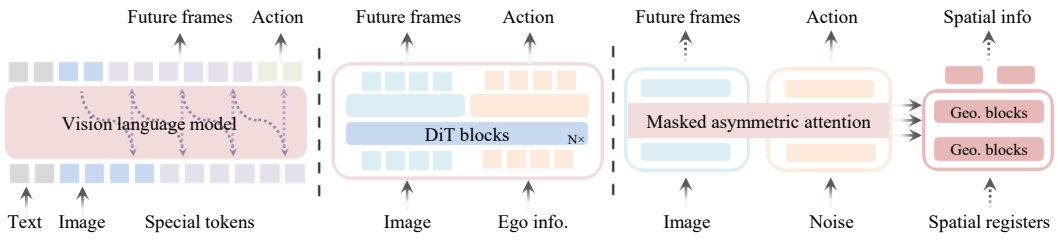
## 1 INTRODUCTION

Achieving safe and rational trajectory planning in Autonomous Driving (AD) and Urban Navigation (UN) requires policies that can not only react to current observations, but also anticipate how the environment evolves under agent interactions. This perspective has led to the emergence of World action models (WAMs), which integrate action generation with predictive modeling of future observations within a unified framework (Wang et al., 2024; Xia et al., 2026; Zhang et al., 2025c; Zhao et al., 2025). By modeling future outcomes, WAMs provide a natural way to capture physical dynamics and task-relevant temporal dependencies, offering a more expressive alternative to standard Vision-Language-Action (VLA) models (Li et al., 2025f; Zhou et al., 2025b; Hirose et al., 2026).

Recent advances in WAMs have led to diverse design paradigms. A natural extension of VLA-based methods toward WAMs is to augment Vision-Language Models (VLMs) (Bai et al., 2025; Chen et al., 2024c) with additional tokens for autoregressive future observation generation (Zhao et al., 2025; Li et al., 2025d), leveraging the rich prior knowledge of VLMs while modeling future environmental dynamics, as illustrated in Figure 1 (a). Another line of approaches (Zhang et al., 2025c; Xia et al., 2026; Liu et al., 2026a) builds upon video generation models (Wan et al., 2025; HaCohen et al., 2024), where intermediate representations are shared between video and action modules through tightly coupled designs, enabling joint modeling of future observations and actions, as illustrated in Figure 1 (b). Despite achieving impressive performance, existing methods still suffer from two key limitations. (1) *Inevitable inference latency*: Current methods require future observation generation during action prediction, introducing additional inference overhead and latency. (2) *Limited spatial grounding*: Future states are primarily represented in semantic or 2D visual spaces, where visually plausible predictions may fail to capture the underlying 3D spatial structure and geometric constraints required for safe driving. Therefore, we argue that WAMs should capture future



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.




(a) VLA-based WAM (b) Video gen.-based WAM (c) Metis (ours)

Figure 1: **Different WAM paradigms.** (a) VLA-based WAMs extend VLMs with autoregressive future observation prediction. (b) Video generation-based WAMs tightly couple future video and action prediction. (c) Our **Metis** enables action-only inference through asymmetric attention, while enhancing spatial grounding during training.

dynamics and spatial structure without explicitly generating future observations at inference time, enabling both efficient and safety-oriented planning.

To address these limitations, we propose **Metis**, an end-to-end WAM for AD and UN. As illustrated in Figure 1 (c), Metis jointly learns future observations, actions, and spatial geometry during training, while requiring only action prediction at inference time. Unlike prior methods (Liu et al., 2026a; Zhao et al., 2025; Zhang et al., 2025c) that couple action inference with explicit future observation generation, Metis treats future world prediction as training-time supervision and maintains an independent action prediction pathway for deployment. Meanwhile, spatial geometry is introduced only as a training-time auxiliary target, with spatial registers predicting complementary depth and occupancy cues from visual and action representations.

Specifically, Metis adopts a Mixture-of-Transformers (MoT) architecture with dedicated video generation and action prediction experts, allowing the two branches to retain task-specific representations while interacting through structured attention. To decouple action prediction from future video generation, we introduce asymmetric world action attention, where future video representations can attend to action representations, while the reverse dependency is masked. This directed information flow prevents stochastic future video representations from entering the action pathway, while preserving world model supervision to the action expert through backpropagation. To improve spatial grounding, we further treat spatial geometry as a training-time auxiliary prediction target, where spatial registers query current visual and future action representations to predict complementary depth and occupancy cues. Consequently, inference requires only action prediction, without explicit future world generation. Extensive experiments on NAVSIM-v2 and CityWalker demonstrate strong performance across AD and UN. Metis achieves 33.8/90.2 EPDMS on navhard/navtest, while reducing action-inference latency from 1.06s to 0.44s under the default 10-step setting; using 2-step inference further lowers the latency to 0.14s. Comprehensive ablations validate the proposed designs, while zero-shot deployment on a quadruped demonstrates their applicability to real-world navigation.

Our contributions are: (**i**) We propose **Metis**, an end-to-end WAM that jointly learns future observations, actions, and spatial geometry during training, while supporting action-only inference. (**ii**) We introduce asymmetric world action attention that removes future video dependency from action prediction while preserving world model supervision, together with training-only spatial registers that predict complementary depth and occupancy cues for spatial grounding. (**iii**) Metis achieves strong performance on NAVSIM-v2 and CityWalker while substantially improving inference efficiency. Extensive ablations and zero-shot quadruped deployment further validate the proposed designs.

## 2 Related Work

**Vision-language-action models.** Many VLA-based methods (Hwang et al., 2024; Jiang et al., 2024; Fu et al., 2025; Tian et al., 2024; Shao et al., 2024; Li et al., 2025a; Zhou et al., 2025a; Huang et al., 2026; Wang et al., 2025b) leverage the pretrained knowledge of VLMs to achieve strong performance in autonomous driving. Some work (Li et al., 2025f; Zhou et al., 2025b; Wang et al., 2025c; Zhang et al., 2025d; Luo et al., 2025) integrates reinforcement learning to enable safe driving via fast–slow reasoning. Furthermore, some methods (Zhao et al., 2025; Li et al., 2026b; Lu et al., 2025; Zeng et al., 2025; Li et al., 2026d; Zuo et al., 2026) extend VLA by incorporating structured prediction of future information. For instance, FSDrive (Zeng et al., 2025) and PWM (Zhao et al., 2025)

enable future image prediction in large models, while SGDrive (Li et al., 2026b) and DrivePI (Liu et al., 2025d) enable occupancy prediction, thereby significantly improving planning quality. In addition, VLA have also been widely explored in outdoor navigation tasks (Zhang et al., 2024b;a; Wei et al., 2025; 2026; Hirose et al., 2026; Zhong et al., 2025; Zhou et al., 2024; Song et al., 2024; Li et al., 2026a). NavFoM (Zhang et al., 2025b) trains a foundation model for navigation and autonomous driving using large-scale data. Abot-N0 (Chu et al., 2026) unifies navigation tasks in a single training framework and achieves autonomous navigation in urban environments.

**World action models.** Early end-to-end driving approaches (Li et al., 2025e;c; Wang et al., 2024; Gui et al., 2026; Yang et al., 2024a; 2025a; Kirby et al., 2026) have explored implicit world modeling to improve trajectory prediction, characterizing scene evolution either through structured representations (Li et al., 2025e; Zhang et al., 2025a) or latent spaces (Li et al., 2025c; Zheng et al., 2025; Wang et al., 2026a). Some methods further incorporate reinforcement learning to enhance closed-loop decision making. Recent advances in general purpose video generation models (Wan et al., 2025; Bruce et al., 2024; Ali et al., 2025) have facilitated their extension to downstream decision-making tasks, including autonomous driving and navigation. Several recent methods (Liu et al., 2026a; Li et al., 2026c; Bi et al., 2025) adapt pretrained video generation models for action planning by jointly predicting future observations and corresponding actions. In parallel, other approaches (Zhang et al., 2025c; Xia et al., 2026; Wang et al., 2025a; Chen et al., 2024b) develop dedicated video action architectures that jointly model visual and action information through tightly coupled diffusion transformers. Despite their strong performance, recent WAMs typically rely on explicit future observation generation during action inference, introducing additional inference overhead. Moreover, future states are predominantly modeled in semantic or 2D visual spaces, leaving the underlying 3D spatial structure and geometric constraints insufficiently captured for reliable planning. These limitations motivate more efficient and spatially grounded world-action modeling.

# 3 METHOD

## 3.1 PROBLEM FORMULATION AND NOTATION

In autonomous driving (AD) and urban navigation (UN), action planning is typically conditioned on the current visual observation $o_t$ and language instruction $l$. Existing WAMs (Li et al., 2026b; Zhang et al., 2025c; Huang et al., 2026; Zhao et al., 2025; Li et al., 2026c) jointly model future visual observations and actions during training, while still requiring explicit future observation generation alongside action prediction at inference time:

$$p_\theta(\mathbf{a}_{t:t+H}, \mathbf{v}_{t+1:t+N} \mid o_t, l), \quad (\mathbf{a}_{t:t+H}, \mathbf{v}_{t+1:t+N}) \sim p_\theta(\cdot \mid o_t, l), \tag{1}$$

where $\mathbf{a}_{t:t+H}$ denotes the planned action sequence over horizon $H$, and $\mathbf{v}_{t+1:t+N}$ represents future visual observations over horizon $N$. In contrast, Metis introduces spatial geometry as a training time auxiliary prediction target, while requiring only action prediction at inference time:

$$p_\theta(\mathbf{a}_{t:t+H}, \mathbf{v}_{t+1:t+N}, \mathbf{g}_{t:t+N} \mid o_t, l), \quad \mathbf{a}_{t:t+H} \sim p_\theta(\mathbf{a}_{t:t+H} \mid o_t, l), \tag{2}$$

where $\mathbf{g}_{t:t+N}$ denotes the sequence of future geometry targets, including depth and occupancy.

## 3.2 MODEL ARCHITECTURE

To accommodate the heterogeneous distributions of video generation and action prediction, we adopt a Mixture-of-Transformers (MoT) architecture with two specialized experts, as illustrated in Figure 2. The video generation expert (VGE) captures spatiotemporal scene dynamics by leveraging the priors of a pretrained video model, while the action expert (AE) focuses on low-dimensional trajectory prediction. The two experts retain expert-specific projections and feed-forward layers, while their representations interact through a joint masked attention operation. During training, we further attach a spatial geometry module that predicts complementary depth and occupancy cues from intermediate world action representations, while being entirely removed at inference time.

We adopt Wan2.2-5B (Wan et al., 2025) as the backbone of the VGE and reuse its pretrained components, including a video VAE for visual encoding and a T5-based text encoder for language conditioning. The AE is implemented as a lightweight diffusion transformer with the same layer depth as the VGE. We further introduce an agent state encoder to incorporate ego states, which, together

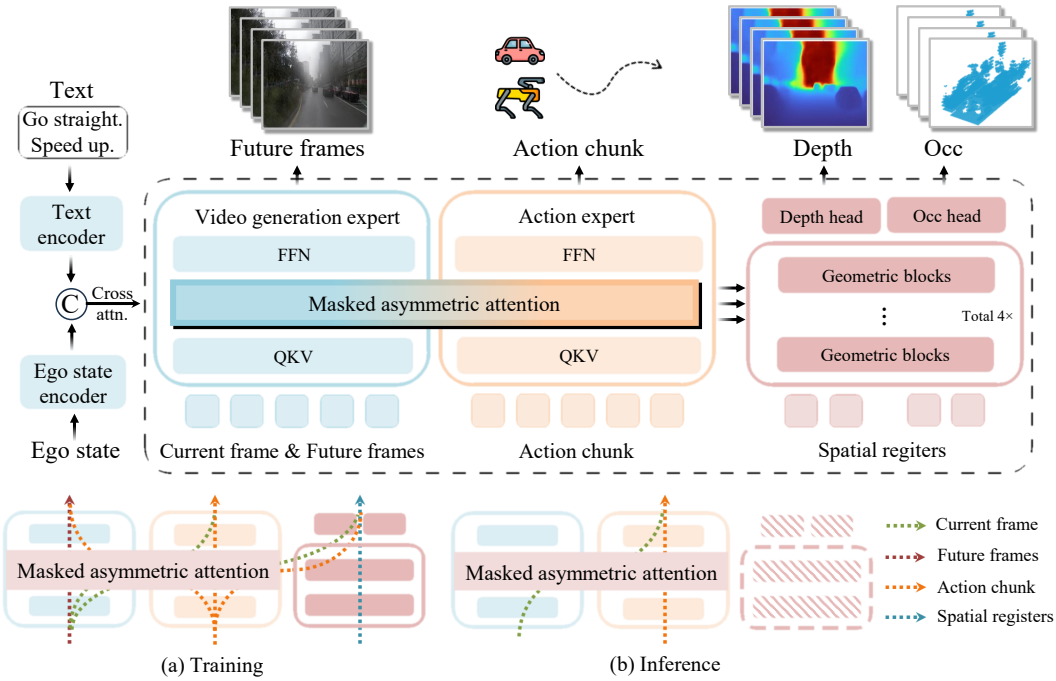



Figure 2: **Overview of Metis.** Top: training time joint world action learning with asymmetric attention and spatial geometry prediction. Bottom: action-only inference using the current observation, without future video or geometry generation.

with language instructions, provides conditional information for action prediction. For the spatial geometry module, the depth head is initialized from the pretrained geometry head of DepthAnythingV3 (Lin et al., 2025), while the occupancy head adopts a lightweight VAE-based decoder following (Liu et al., 2025b).

As illustrated in Figure 2, during training, the model operates on four types of representations: latent tokens of the current observation, noisy future video tokens, noisy action tokens, and learnable spatial registers. The video and action tokens are processed by their respective experts and interact through masked asymmetric attention, while the spatial registers query the current visual and future action representations for geometric prediction. At inference time, future video generation and the spatial-register branch are omitted, leaving only the current observation and action pathway for efficient action prediction.

### 3.3 Asymmetric World Action Attention

Low-latency action inference is essential for both AD and UN. However, existing WAMs (Zhao et al., 2025; Liu et al., 2026b;a) typically couple action prediction with future observation generation, introducing substantial inference overhead. To decouple these two processes, we introduce an asymmetric attention mechanism between the VGE and AE, which explicitly regulates cross-expert information flow, as illustrated in Figure 3 (a). Unlike conventional joint attention with bidirectional interaction between video and action tokens, our design imposes asymmetric visibility to support joint world action learning during training while enabling action-only inference.

Specifically, action tokens are prevented from attending to future video tokens, while future video tokens can attend to both the current visual observations and future action tokens. This asymmetric connectivity removes the forward dependency of action prediction on future video representations, while preserving the backward supervision from the video generation objective to the action expert. Let $\hat{\mathbf{a}}$ denote the predicted action sequence, $\mathbf{Z}_v^{\text{fut}}$ the future video representations, and $\theta_{\text{AE}}$ the parameters of the action expert. The resulting information flow can be summarized as:

$$\frac{\partial \hat{\mathbf{a}}}{\partial \mathbf{Z}_v^{\text{fut}}} = 0, \qquad \frac{\partial \mathcal{L}_{\text{video}}}{\partial \theta_{\text{AE}}} \neq 0. \tag{3}$$

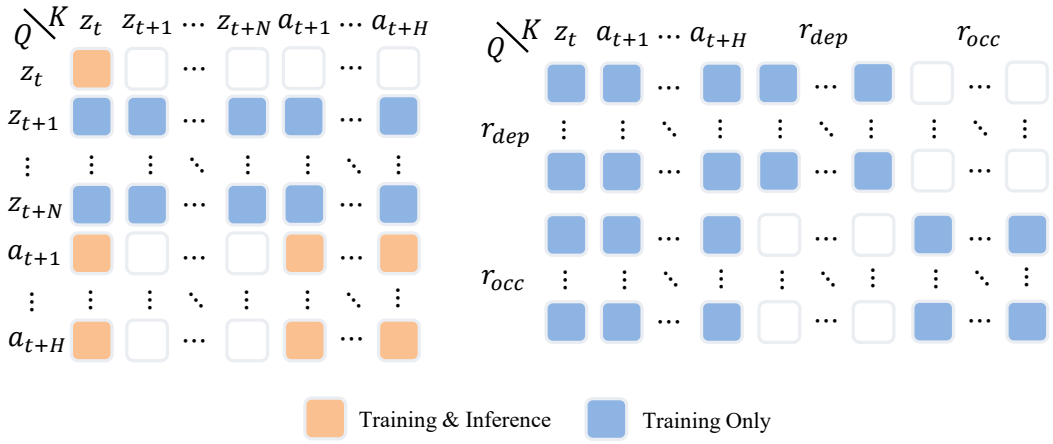



Figure 3: Attention mask designs in Metis. Left: asymmetric attention mask for world action interaction. Right: spatial register mask for training-time geometry prediction.

Consequently, Metis removes the forward dependency on future video representations while preserving backward supervision from world modeling, enabling efficient action-only inference.

### 3.4 Spatial Geometry Registers

To enhance the spatial awareness of world action representations without introducing additional inference overhead, we introduce training-only spatial registers as learnable geometry queries. Rather than injecting geometric tokens into the main world action stream, these registers query intermediate representations from both the current observation and future action tokens, allowing explicit geometric supervision to shape their internal representations.

We use $\mathbf{R}$ to denote the learnable spatial registers, which consist of two independent groups: depth registers $\mathbf{r}_{\text{dep}}$ and occupancy registers $\mathbf{r}_{\text{occ}}$. The former captures continuous geometric structure through dense depth modeling, while the latter characterizes discrete occupied or free-space structure through binary occupancy modeling. For the $\ell$-th selected MoT layer, let $\mathbf{R}^\ell$ denote the spatial registers, and let $\mathbf{Z}_o^\ell$ and $\mathbf{Z}_a^\ell$ denote the intermediate representations of the current observation and future actions, respectively. Here, $\mathbf{Z}_o^\ell$ is extracted from the VGE stream corresponding only to the current observation, excluding future-video representations. At each selected layer, the spatial registers are updated by a geometry block through cross-attention:

$$\mathbf{R}^{\ell+1} = \text{GeoBlock}^\ell\left(Q = \mathbf{R}^\ell, K = V = \left[\mathbf{R}^\ell; \mathbf{Z}_o^\ell; \mathbf{Z}_a^\ell\right]\right). \tag{4}$$

$\mathbf{r}_{\text{dep}}$ and $\mathbf{r}_{\text{occ}}$ emain mutually invisible to preserve their task-specific geometric representations and avoid interference between the two supervision signals. Following the same asymmetric information flow principle introduced above, the spatial registers can attend to the current observation and action representations, while the reverse interactions are masked. The complete cross-attention pattern is illustrated in Figure 3 (b).

After the final interaction layer, the two register groups are independently projected into their task-specific feature spaces and decoded by the corresponding geometry heads:

$$\hat{\mathbf{D}} = \mathcal{H}_{\text{dep}}\left(\mathcal{P}_{\text{dep}}(\mathbf{r}_{\text{dep}})\right), \qquad \hat{\mathbf{O}} = \mathcal{H}_{\text{occ}}\left(\mathcal{P}_{\text{occ}}(\mathbf{r}_{\text{occ}})\right). \tag{5}$$

Here, $\mathcal{P}_{\text{dep}}$ and $\mathcal{P}_{\text{occ}}$ denote the task-specific projection layers, while $\mathcal{H}_{\text{dep}}$ and $\mathcal{H}_{\text{occ}}$ denote the corresponding depth and occupancy prediction heads. The resulting $\hat{\mathbf{D}}$ and $\hat{\mathbf{O}}$ represent the predicted depth and binary occupancy sequences over the current and future time steps, respectively.

### 3.5 Training Objective

We jointly optimize action prediction, future video generation, and spatial geometry learning within a unified training framework. The action and video branches are trained with flow-matching objectives, while the spatial registers are supervised by complementary depth and occupancy objectives.

For action prediction, we supervise the velocity field conditioned on the current context:

$$\mathcal{L}_{\text{action}} = \mathbb{E}\left[\|u_\theta\left(\mathbf{a}^{(s)}_{t:t+H}, s \mid o_t, l\right) - \dot{\mathbf{a}}^{(s)}_{t:t+H}\|^2\right] \tag{6}$$

where $s \in [0,1]$ denotes the flow timestep, $\mathbf{a}^{(s)}_{t:t+H} = (1-s)\boldsymbol{\epsilon} + s\mathbf{a}_{t:t+H}$ with $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$, and $\dot{\mathbf{a}}^{(s)}_{t:t+H} = \mathbf{a}_{t:t+H} - \boldsymbol{\epsilon}$. For future video generation, we similarly supervise the velocity field conditioned on the current context and noisy action tokens:

$$\mathcal{L}_{\text{video}} = \mathbb{E}\left[\|u_\phi\left(\mathbf{z}^{(s)}_{t+1:t+N}, s \mid o_t, l, \mathbf{a}^{(s)}_{t:t+H}\right) - \dot{\mathbf{z}}^{(s)}_{t+1:t+N}\|^2\right], \tag{7}$$

where $\mathbf{z}^{(s)}_{t+1:t+N} = (1-s)\boldsymbol{\epsilon} + s\mathbf{z}_{t+1:t+N}$ denotes the noisy latent representation of future observations. For geometric register supervision, depth is optimized over the current and future states, while occupancy supervision is applied only to the current observation:

$$\mathcal{L}_{\text{dep}} = \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\frac{1}{|\Omega^t_{\text{D}}|}\sum_{i\in\Omega^t_{\text{D}}}\rho_{\text{SL1}}\left(\hat{d}_{t,i} - d_{t,i}\right), \mathcal{L}_{\text{occ}} = \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\frac{1}{|\Omega^t_{\text{O}}|}\sum_{i\in\Omega^t_{\text{O}}}\rho_{\text{BCE}}\left(\hat{o}_{t,i}, ot, i\right). \tag{8}$$

Here, $\mathcal{T}$ contains the future geometry supervision timestamps. $\Omega^t_{\text{D}}$ and $\Omega^t_{\text{O}}$ denote the valid depth pixels and occupancy voxels, respectively. $\rho_{\text{SL1}}$ and $\rho_{\text{BCE}}$ denote the Smooth-$L_1$ and binary cross-entropy functions. The overall training objective is formulated as

$$\mathcal{L} = \lambda_{\text{action}}\mathcal{L}_{\text{action}} + \lambda_{\text{video}}\mathcal{L}_{\text{video}} + \lambda_{\text{dep}}\mathcal{L}_{\text{dep}} + \lambda_{\text{occ}}\mathcal{L}_{\text{occ}}. \tag{9}$$

We set $\lambda_{\text{action}} = 1$, $\lambda_{\text{video}} = 4$, $\lambda_{\text{dep}} = 10$, and $\lambda_{\text{occ}} = 1$ by default.

# 4 Experiments

## 4.1 Experimental setup

**Implementation details.** We adopt Wan2.2-5B (Wan et al., 2025) as the VGE, while the AE follows the same transformer architecture with a reduced hidden dimension of $d_a = 1024$, resulting in approximately 1B parameters and a total model size of 6B. Unless otherwise specified, we use an input resolution of $320 \times 384$. For geometric supervision, we predict four future states at 1-second intervals over the 4-second horizon. The depth and occupancy groups employ $12 \times 10$ and $10 \times 10$ spatial register grids at each future timestep, respectively. Geometry blocks are inserted after MoT layers 11, 13, 15, and 17. For AD, we use only the front-facing camera and predict 8 waypoints over a 4-second horizon at 0.5-second intervals, with each waypoint parameterized as $(x, y, \theta)$. Future video and action sequences are temporally aligned at a 1:1 ratio. We train the model for 24k steps with an effective batch size of 256 using gradient accumulation. For UN, we predict 5 future waypoints, each represented as $(x, y)$, and train the model for 5k steps. For both tasks, we use 10 denoising steps during inference. All experiments are conducted on 8 NVIDIA H200 GPUs. More implementation details are provided in the Appendix A.1.

**NAVSIM-v2.** We train our model on the navtrain subset (1,192 scenarios) and evaluate performance on two benchmarks: `navhard` and `navtest` (Cao et al., 2025). `navhard` comprises 244 safety-critical real-world scenarios (Stage 1) and 4,164 3DGS-generated synthetic counterparts (Stage 2) for closed-loop evaluation. `navtest` includes 12,146 scenarios for assessing generalization. We also provide results on NAVSIM-v1 in the Appendix F.

**CityWalker.** The dataset (Liu et al., 2025c) contains 15 hours of teleoperation data collected across diverse urban areas in New York City, with 6 hours used for fine-tuning and 9 hours for testing. We evaluate performance under several challenging scenarios and utilize Maximum Average Orientation Error (MAOE) as the primary metric to assess human-like trajectory alignment, supplemented by the average L2 distance for spatial accuracy.

**Real-world experiments** We deploy our model on the Unitree Go2 quadruped for real-world navigation. We adapt the PD controller provided in (Wei et al., 2026) to provide velocity command to the quadruped. We conduct zero-shot real-world experiments in both indoor and outdoor environments, demonstrating the strong generalization ability of our **Metis**. Additional quantitative and qualitative results are provided in the Appendix B.1, with supplementary videos included separately.

Table 1: **Performance on the NAVSIM-v2 `navhard`.** S. is the per-stage score. For our method, † denotes $640 \times 768$ inputs; the default setting uses $320 \times 384$.

| Method | Reference | Stage | NC ↑ | DAC ↑ | DDC ↑ | TLC ↑ | EP ↑ | TTC ↑ | LK ↑ | HC ↑ | EC ↑ | S. ↑ | EPDMS ↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *Traditional E2E-based Planner* | | | | | | | | | | | | | |
| DiffusionDrive (Liao et al., 2025) | CVPR2025 | S1 | 96.8 | 86.0 | 98.8 | 99.3 | 84.0 | 95.8 | 96.7 | 97.6 | 79.6 | 66.7 | 27.5 |
| | | S2 | 80.1 | 72.8 | 84.4 | 98.4 | 85.9 | 76.6 | 46.4 | 96.3 | 72.8 | 40.5 | |
| GuideFlow (Liu et al., 2025a) | CVPR2026 | S1 | 96.6 | 80.5 | 96.3 | 99.3 | 82.3 | 94.9 | 91.5 | 97.7 | 67.8 | - | 27.1 |
| | | S2 | 87.3 | 76.7 | 88.8 | 99.2 | 84.3 | 85.1 | 49.7 | 93.1 | 44.5 | - | |
| *VLA-based Planner* | | | | | | | | | | | | | |
| DriveVLA-W0 (Li et al., 2025d) | ICLR2026 | S1 | 96.8 | 83.3 | 99.0 | 99.6 | 84.6 | 95.3 | 96.4 | 97.6 | 78.2 | - | 24.4 |
| | | S2 | 76.8 | 64.3 | 79.9 | 98.3 | 89.2 | 75.0 | 46.8 | 95.8 | 53.1 | - | |
| ReCogDrive (Li et al., 2025f) | ICLR2026 | S1 | 96.4 | 78.9 | 98.7 | 99.8 | 82.6 | 95.6 | 94.4 | 97.6 | 74.2 | 67.7 | 25.7 |
| | | S2 | 80.2 | 65.0 | 82.4 | 98.7 | 85.2 | 76.9 | 43.8 | 96.6 | 71.8 | 37.6 | |
| SGDrive (Li et al., 2026b) | CVPR2026 | S1 | 95.8 | 87.6 | 97.8 | 99.8 | 84.4 | 94.7 | 92.9 | 97.8 | 28.9 | 71.1 | 25.5 |
| | | S2 | 79.4 | 65.4 | 79.1 | 98.9 | 88.9 | 75.3 | 42.7 | 96.4 | 29.6 | 35.2 | |
| *WAM-based Planner* | | | | | | | | | | | | | |
| DriveLaW (Xia et al., 2026) | CVPR2026 | S1 | 97.3 | 89.1 | 99.2 | 99.6 | 84.3 | 97.1 | 96.2 | 97.8 | 67.6 | - | 30.6 |
| | | S2 | 82.5 | 67.6 | 83.5 | 98.1 | 84.8 | 78.5 | 45.8 | 96.4 | 57.3 | - | |
| **Metis (Ours)** | - | S1 | 96.3 | 89.8 | 99.3 | 99.3 | 84.6 | 95.1 | 97.8 | 97.6 | 79.6 | 77.6 | 33.8 |
| | | S2 | 80.3 | 75.7 | 85.1 | 98.1 | 86.7 | 77.3 | 48.4 | 94.8 | 73.0 | 43.1 | |
| **Metis † (Ours)** | - | S1 | 97.0 | 92.0 | 99.4 | 99.6 | 84.9 | 96.2 | 98.7 | 97.8 | 80.9 | **81.1** | **35.7** |
| | | S2 | 80.2 | 74.4 | 85.4 | 98.3 | 86.9 | 76.8 | 48.6 | 95.3 | 71.3 | **43.7** | |

Table 2: **Performance comparison on NAVSIM-v2** `navtest` with human penalty enabled. Results use the corrected official EPDMS implementation. C and L denote camera and LiDAR. For our method, † denotes $640 \times 768$ inputs; the default setting uses $320 \times 384$.

| Method | Sensors | NC↑ | DAC↑ | DDC↑ | TLC↑ | EP↑ | TTC↑ | LK↑ | HC↑ | EC↑ | EPDMS↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Human Agent | - | 100 | 100 | 99.8 | 100 | 87.4 | 100 | 100 | 98.1 | 90.1 | 94.5 |
| *Traditional E2E-based Planner* | | | | | | | | | | | |
| DriveSuprim (Yao et al., 2026) | 3xC | 98.4 | 98.6 | 99.6 | 99.8 | **90.5** | 97.8 | 97.0 | 98.3 | 78.6 | 87.1 |
| DiffusionDriveV2 (**?**) | 3xC+L | 97.7 | 96.6 | 99.2 | 99.8 | 88.9 | 97.2 | 96.0 | 97.8 | **91.0** | 87.5 |
| WoTE (Li et al., 2025e) | 3xC | 98.5 | 96.8 | 98.8 | 99.8 | 86.1 | 97.9 | 95.5 | 98.3 | 82.9 | 87.7 |
| *VLA-based Planner* | | | | | | | | | | | |
| ReCogDrive (Li et al., 2025f) | 1xC | 98.3 | 95.2 | 99.5 | 99.8 | 87.1 | 97.5 | 96.6 | 98.3 | 86.5 | 83.6 |
| SGDrive (Li et al., 2026b) | 1xC | 98.6 | 94.3 | 99.5 | **99.9** | 86.0 | 97.9 | 96.1 | 98.3 | 85.9 | 86.2 |
| Vega (Zuo et al., 2026) | 1xC | **98.9** | 95.3 | 99.4 | **99.9** | 87.0 | **98.4** | 96.1 | 98.3 | 76.3 | 86.9 |
| DriveFine (Dang et al., 2026) | 1xC | 98.7 | 97.3 | 99.5 | 99.8 | 88.7 | 97.8 | 97.7 | **98.4** | 83.8 | 89.7 |
| *WAM-based Planner* | | | | | | | | | | | |
| DriveVLA-W0 (Li et al., 2025d) | 1xC | 98.4 | 95.2 | 99.4 | **99.9** | 86.6 | 97.9 | 97.8 | 98.3 | 82.7 | 86.9 |
| PWM (Zhao et al., 2025) | 1xC | 98.8 | 95.9 | 99.4 | **99.9** | 86.4 | **98.4** | 97.6 | 98.3 | 85.3 | 88.2 |
| DriveLaW (Xia et al., 2026) | 1xC | 98.7 | 96.9 | 99.6 | 99.8 | 87.5 | 98.3 | 97.6 | 98.4 | 77.4 | 88.6 |
| EponaV2 (Xu et al., 2026) | 1xC | 98.5 | 97.4 | 99.5 | **99.9** | 87.9 | 98.1 | 97.7 | 98.2 | 77.4 | 88.9 |
| Latent-WAM (Wang et al., 2026a) | 3xC | 98.1 | 97.3 | 99.6 | 99.8 | 87.7 | 97.3 | 97.6 | 98.11 | 87.4 | 89.3 |
| **Metis (Ours)** | 1xC | 98.3 | 97.9 | **99.7** | 99.8 | 87.9 | 97.7 | 98.0 | **98.4** | 88.2 | 90.2 |
| **Metis † (Ours)** | 1xC | 98.6 | **98.1** | **99.7** | 99.8 | 87.8 | 98.0 | **98.1** | **98.4** | 88.2 | **90.8** |

### 4.2 Main results

**Navhard benchmark.** As shown in Table 1, Metis achieves strong overall performance on the challenging navhard benchmark. The default model reaches 33.8 EPDMS, substantially outperforming representative VLA-based methods such as ReCogDrive (25.7), as well as the WAM-based DriveLaW (30.6). Increasing the input resolution further improves EPDMS to 35.7. These results suggest that Metis remains effective under the more challenging long-tail and synthetic scenarios considered by navhard.

**Navtest benchmark.** As shown in Table 2, Metis achieves 90.2 EPDMS on the diverse navtest benchmark, while the high-resolution setting further improves the score to 90.8. It also outperforms representative VLA-based planners and prior WAM-based methods. Notably, the default low-resolution setting already achieves strong performance, suggesting that the effectiveness of Metis does not rely on increased input resolution. Beyond the overall score, Metis performs strongly in DAC, DDC, LK, and EC, suggesting balanced improvements across multiple planning criteria.

**CityWalker dataset.** As shown in Table 3, Metis achieves the best overall performance among fine-tuned methods, reducing the mean L2 error to 0.71m and MAOE to $11.8^\circ$. Compared with CityWalker, the strongest fine-tuned baseline, our method substantially improves both trajectory accuracy and orientation prediction across diverse navigation scenarios. Notably, despite ABot-N0

Table 3: **Performance comparison on the CityWalker dataset.** Percentages indicate scenario proportions. "Mean" averages the six scenarios, while "All" averages all samples. Best results are in bold, and "-" indicates unreported results.

| Method | Metric | Mean | Turn 8% | Crossing 12% | Detour 12% | Proximity 6% | Crowd 7% | Other 55% | All 100% |
|---|---|---|---|---|---|---|---|---|---|
| *Pretrained method* | | | | | | | | | |
| ABot-N0 (Chu et al., 2026) | ↓ L2 (m) | - | - | - | - | - | - | - | - |
| | ↓ MAOE (°) | **11.2** | 21.3 | **9.8** | 12.8 | **8.1** | 8.8 | **6.3** | **7.6** |
| *Fine-tuned method* | | | | | | | | | |
| GNM (Shah et al., 2023a) | ↓ L2 (m) | 1.22 | 2.36 | 1.36 | 1.42 | 0.88 | 0.76 | **0.55** | 0.74 |
| | ↓ MAOE (°) | 16.2 | 31.1 | 14.8 | 12.5 | 14.7 | 12.8 | 11.0 | 12.1 |
| ViNT (Shah et al., 2023b) | ↓ L2 (m) | 1.30 | 1.91 | 1.13 | 1.14 | 0.77 | **0.66** | 0.57 | 0.70 |
| | ↓ MAOE (°) | 16.5 | 31.1 | 15.4 | 12.9 | 14.8 | 13.3 | 11.6 | 12.6 |
| NoMaD (Sridhar et al., 2024) | ↓ L2 (m) | 1.39 | 2.49 | 1.56 | 1.55 | 1.06 | 0.95 | 0.76 | 0.74 |
| | ↓ MAOE (°) | 19.1 | 35.1 | 18.5 | 15.6 | 18.1 | 14.3 | 12.8 | 12.1 |
| CityWalker (Liu et al., 2025c) | ↓ L2 (m) | 1.11 | 1.27 | 1.00 | 1.15 | 1.06 | 1.12 | 1.06 | 1.07 |
| | ↓ MAOE (°) | 15.2 | 26.6 | 14.1 | 13.9 | 14.3 | 12.0 | 10.4 | 11.5 |
| **Metis (Ours)** | ↓ L2 (m) | **0.71** | **0.69** | **0.77** | **0.66** | **0.76** | 0.77 | 0.63 | **0.67** |
| | ↓ MAOE (°) | 11.8 | **19.5** | 11.9 | **9.1** | 11.4 | **8.0** | 7.8 | 9.8 |

Table 4: **Effect and diagnostics of asymmetric information flow.** Action dispersion is measured over 64 scenes with 8 samples per scene by resampling only the future-video noise while fixing the observation and AE denoising noise. Grad. Ratio denotes $\|\nabla_{\mathrm{AE}}\mathcal{L}_{\mathrm{video}}\|/\|\nabla_{\mathrm{AE}}\mathcal{L}_{\mathrm{action}}\|$ and is averaged over 20 validation batches.

| | | Planning Performance | | Information-Flow Diagnostics | | |
|---|---|---|---|---|---|---|
| Variant | Forward Flow | navtest ↑ | navhard ↑ | Traj. RMS (m) ↓ | Final RMS (m) ↓ | Grad. Ratio |
| *Joint* | Action ↔ Video | 88.9 | 31.7 | 0.0750 | 0.1449 | 0.0200 |
| *Isolated* | None | 89.5 | 32.4 | 0 | 0 | 0 |
| *Ours* | Action → Video | 90.2 | 33.8 | 0 | 0 | 0.0196 |

being pretrained on large-scale navigation data, Metis achieves lower MAOE in challenging Turn, Detour, and Crowd scenarios, demonstrating generalization to complex urban navigation.

## 5 Ablation Studies

**Effect of asymmetric information flow.** We compare three interaction patterns between the VGE and AE: bidirectional interaction (*Joint*), no cross-expert interaction (*Isolated*), and asymmetric interaction (*Ours*). As shown in Table 4, *Ours* consistently outperforms both *Joint* and *Isolated*, achieving 90.2 EPDMS on `navtest` and 33.8 on `navhard`. We further examine whether these interaction patterns exhibit the desired forward and backward information flow. When resampling only the future video noise while fixing the current observation and action noise, *Joint* produces noticeable variations in the predicted actions, whereas *Isolated* and *Ours* remain invariant. This confirms that our asymmetric mask blocks the forward dependency of action prediction on stochastic future video representations. Meanwhile, *Ours* yields a non-zero gradient ratio of 0.0196, confirming that the video objective still provides supervision to the AE, whereas *Isolated* completely blocks this gradient path. Together, these results show that asymmetric attention simultaneously decouples action inference from future video generation and preserves world-modeling supervision during training.

**Effect of spatial geometry registers.** Table 5 evaluates the contribution of depth and occupancy supervision. Without geometric registers, the model achieves 89.1 EPDMS on `navtest` and 32.7 on `navhard`. Introducing either depth or occupancy registers consistently improves planning performance, confirming the benefit of explicit geometric supervision. Combining both objectives further improves EPDMS to 90.2 and 33.8, respectively, while also achieving the best DAC and EP. These results indicate that dense depth structure and occupied/free-space geometry provide complementary spatial cues, jointly enhancing the geometric awareness of world action representations.

**Efficiency analysis.** As shown in Table 6, all methods are evaluated on a single NVIDIA RTX 4090 for a fair latency comparison. Our asymmetric attention enables action inference without explicit future video generation, reducing latency from 1.06 s to 0.44 s while maintaining the same 90.2 EPDMS. Further reducing action denoising from 10 to 2 steps lowers the latency to 0.14 s

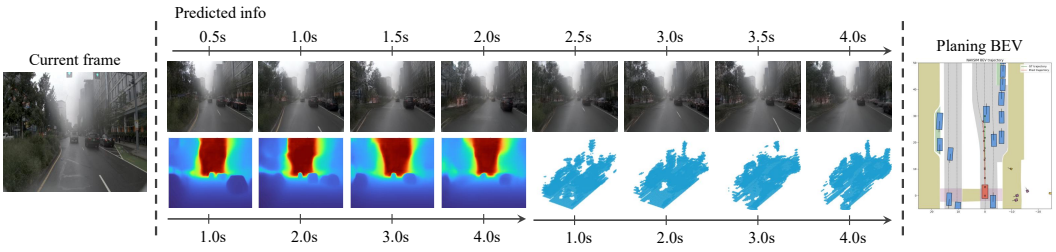



Figure 4: **Qualitative multi stream prediction and planning.** Left: the current observation. Middle: predicted future visual observations, depth maps, and occupancy representations over a 4-second horizon. Right: the corresponding planning result in BEV.

Table 5: Ablation of spatial geometry registers.

| Registers | | navtest | | | navhard |
|---|---|---|---|---|---|
| Depth | Occ. | DAC | EP | EPDMS | EPDMS |
| | | 96.9 | 86.8 | 89.1 | 32.7 |
| ✓ | | 97.5 | 87.8 | 89.8 | 33.4 |
| | ✓ | 97.6 | 87.4 | 89.7 | 33.2 |
| ✓ | ✓ | **98.1** | **87.9** | **90.2** | **33.8** |

Table 6: Inference efficiency comparison.

| Method | EPDMS ↑ | Latency (s) ↓ |
|---|---|---|
| DriveVLA-W0 (w/ video) (Li et al., 2025d) | 86.9 | 0.69 |
| PWM (w/ video) (Zhao et al., 2025) | 88.2 | 0.83 |
| Metis (w/ video) | **90.2** | 1.06 |
| Metis (10-step) | **90.2** | 0.44 |
| Metis (2-step) | 90.0 | **0.14** |

with only a marginal 0.2-point performance drop, resulting in a 7.6× speedup over full world-action inference. Notably, the spatial registers and geometry heads are used only during training and therefore introduce no additional inference overhead.

**Effect of future world-model supervision.** Table 7 isolates the contribution of future world modeling by disabling spatial geometry registers for all variants. *VAE+AE only* removes both the VGE and future-video objective, while Metis w/o video loss retains the VGE architecture without explicit future-video supervision; Metis w/ video loss further activates the world-modeling objective. Introducing the VGE already brings substantial gains, while enabling future-video supervision further improves performance, particularly on the more challenging navhard benchmark. These results show that the performance gain cannot be attributed solely to the pretrained VGE, as activating the future video objective provides further improvements. Together with the asymmetric attention ablation, this supports our design of leveraging future world modeling through asymmetric information flow while keeping action inference independent of future video generation.

Table 7: **Effect of future world-model supervision.** Spatial geometry registers are disabled for all variants.

| Variant | VGE | Video Loss | navtest EPDMS ↑ | navhard EPDMS ↑ |
|---|---|---|---|---|
| VAE + AE only | ✗ | ✗ | 78.5 | 15.7 |
| Metis w/o video loss | ✓ | ✗ | 87.9 | 29.4 |
| Metis w/video loss | ✓ | ✓ | **89.1** | **32.7** |

**Qualitative analysis.** Figure 4 visualizes the predicted future video, depth, and occupancy sequences of Metis. Across the prediction horizon, the generated future observations exhibit coherent scene evolution and preserve the overall road layout, while the corresponding depth and occupancy predictions capture meaningful spatial structure and geometric consistency. These visualizations provide qualitative evidence that the learned representations encode both future scene dynamics and complementary geometric information that can support action prediction. Additional evaluations of future video, depth, and occupancy, together with analyses of hyperparameter sensitivity, inference efficiency, and richer spatial-temporal contexts, are provided in the Appendix.

## 6 Conclusion

In this paper, we presented **Metis**, an end-to-end WAM for AD and UN following a "joint training, decoupled inference" paradigm. Asymmetric world action attention enables joint learning of future observations and actions while allowing action-only inference, and training only spatial registers introduce complementary depth and occupancy supervision without additional inference overhead. Extensive experiments on NAVSIM-v2 and CityWalker, together with ablations and zero-shot real-world deployment, demonstrate strong planning performance and substantially improved inference efficiency. These results highlight a promising direction toward general and efficient WAM that exploit rich future and geometric supervision during training without requiring explicit world generation at inference time.

### AI USE STATEMENT

In this work, we used generative AI tools to assist with English language polishing, sentence rewriting, and code generation. The research ideas, methodology, and experimental design were developed by the authors. Generative AI was not used to generate experimental results; all experiments were conducted by the authors. AI-assisted code was manually reviewed, verified, and tested before use. All AI-assisted manuscript content was also carefully reviewed and revised by the authors to ensure technical correctness and consistency with the actual methodology and experimental results. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

### REPRODUCIBILITY STATEMENT

We have made substantial efforts to ensure the reproducibility of this work. The main paper provides the complete model formulation, training objectives, and evaluation settings. The Appendix further documents detailed implementation and training configurations, spatial geometry register design and target construction, inference settings, evaluation protocols, and additional ablation studies. We also provide detailed descriptions of the depth and occupancy supervision, denoising settings, efficiency measurements, and real-world evaluation setup. Together, these materials specify the key architectural, optimization, data processing, and evaluation details required to reproduce the reported results.

## REFERENCES

Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A unified latent action world model. *arXiv preprint arXiv:2512.13030*, 2025.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *ICML*, 2024.

Wei Cao, Marcel Hallgarten, Tianyu Li, Daniel Dauner, Xunjiang Gu, Caojun Wang, Yakov Miron, Marco Aiello, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. Pseudo-simulation for autonomous driving. In *CoRL*, 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*, 2024a.

Yuntao Chen, Yuqi Wang, and Zhaoxiang Zhang. Drivinggpt: Unifying driving world modeling and planning with multi-modal autoregressive transformers. *arXiv preprint*, 2024b.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *CVPR*, 2024c.

Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE TPAMI*, 2022.

Zedong Chu, Shichao Xie, Xiaolong Wu, Yanfen Shen, Minghua Luo, Zhengbo Wang, Fei Liu, Xiaoxu Leng, Junjun Hu, Mingyang Yin, et al. Abot-n0: Technical report on the vla foundation model for versatile embodied navigation. *arXiv preprint arXiv:2602.11598*, 2026.

Chenxu Dang, Sining Ang, Yongkang Li, Haochen Tian, Jie Wang, Guang Li, Hangjun Ye, Jie Ma, Long Chen, and Yan Wang. Drivefine: Refining-augmented masked diffusion vla for precise and robust driving. *arXiv preprint arXiv:2602.14577*, 2026.

Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *Advances in Neural Information Processing Systems (NeurIPS)*, 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. *IEEE RA-L*, 2025.

Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In *ICCV*, 2025.

Xingtai Gui, Meijie Zhang, Tianyi Yan, Wencheng Han, Jiahao Gong, Feiyang Tan, Cheng-zhong Xu, and Jianbing Shen. Bridging scene generation and planning: Driving with world model via unifying vision and motion representation. *arXiv preprint arXiv:2603.14948*, 2026.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Noriaki Hirose, Catherine Glossop, Dhruv Shah, and Sergey Levine. Omnivla: An omni-modal vision-language-action model for robot navigation. In *ICRA*, 2026.

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*, 2023.

Wenhui Huang, Songyan Zhang, Qihang Huang, Zhidong Wang, Zhiqi Mao, Collister Chua, Zhan Chen, Long Chen, and Chen Lv. Automot: A unified vision-language-action model with asynchronous mixture-of-transformers for end-to-end autonomous driving. *arXiv preprint arXiv:2603.14851*, 2026.

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, Bencheng Liao, Xingyu Zhang, Wei Yin, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Senna: Bridging large vision-language models and end-to-end autonomous driving. *arXiv preprint arXiv:2410.22313*, 2024.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

Ellington Kirby, Alexandre Boulch, Yihong Xu, Yuan Yin, Gilles Puy, Éloi Zablocki, Andrei Bursuc, Spyros Gidaris, Renaud Marlet, Florent Bartoccioni, Anh-Quan Cao, Nermin Samet, Tuan-Hung Vu, and Matthieu Cord. Driving on registers. In *CVPR*, 2026.

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

Jingyu Li, Zhe Liu, Wenxiao Wu, and Li Zhang. Mcnav: Memory-aware dynamic cognitive map for zero-shot goal-oriented navigation. *arXiv preprint arXiv:2605.19594*, 2026a.

Jingyu Li, Junjie Wu, Dongnan Hu, Xiangkai Huang, Bin Sun, Zhihui Hao, Xianpeng Lang, Xiatian Zhu, and Li Zhang. Sgdrive: Scene-to-goal hierarchical world cognition for autonomous driving. *arXiv preprint arXiv:2601.05640*, 2026b.

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*, 2025b.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026c.

Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. In *ICLR*, 2025c.

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*, 2025d.

Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. In *ICCV*, 2025e.

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*, 2025f.

Yongkang Li, Lijun Zhou, Sixu Yan, Bencheng Liao, Tianyi Yan, Kaixin Xiong, Long Chen, Hongwei Xie, Bing Wang, Guang Chen, et al. Unidrivevla: Unifying understanding, perception, and action planning for autonomous driving. *arXiv preprint arXiv:2604.02190*, 2026d.

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*, 2025.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Lin Liu, Caiyan Jia, Guanyi Yu, Ziying Song, JunQiao Li, Feiyang Jia, Peiliang Wu, Xiaoshuai Hao, and Yadan Luo. Guideflow: Constraint-guided flow matching for planning in end-to-end autonomous driving. *arXiv preprint arXiv:2511.18729*, 2025a.

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*, 2026a.

Qiqi Liu, Huan Xu, Jingyu Li, Bin Sun, Zhihui Hao, Dangen She, Xiatian Zhu, and Li Zhang. Uni-world vla: Interleaved world modeling and planning for autonomous driving. *arXiv preprint arXiv:2603.27287*, 2026b.

Ruixun Liu, Lingyu Kong, Derun Li, and Hang Zhao. Occvla: Vision-language-action model with implicit 3d occupancy supervision. *arXiv preprint arXiv:2509.05578*, 2025b.

Xinhao Liu, Jintong Li, Yicheng Jiang, Niranjan Sujay, Zhicheng Yang, Juexiao Zhang, John Abanes, Jing Zhang, and Chen Feng. Citywalker: Learning embodied urban navigation from web-scale videos. In *CVPR*, 2025c.

Zhe Liu, Runhui Huang, Rui Yang, Siming Yan, Zining Wang, Lu Hou, Di Lin, Xiang Bai, and Hengshuang Zhao. Drivepi: Spatial-aware 4d mllm for unified autonomous driving understanding, perception, prediction and planning. *arXiv preprint arXiv:2512.12799*, 2025d.

Hao Lu, Ziyang Liu, Guangfeng Jiang, Yuanfei Luo, Sheng Chen, Yangang Zhang, and Ying-Cong Chen. Uniugp: Unifying understanding, generation, and planing for end-to-end autonomous driving. *arXiv preprint arXiv:2512.09864*, 2025.

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. In *ICRA*, 2025.

Yuechen Luo, Fang Li, Qimao Chen, Shaoqing Xu, Jiaxin Liu, Ziying Song, Zhi-xin Yang, and Fuxi Wen. Unleashing vla potentials in autonomous driving via explicit learning from failures. In *CVPR*, 2026.

Chen Min, Dawei Zhao, Liang Xiao, Yiming Nie, and Bin Dai. Uniworld: Autonomous driving pre-training via world models. *arXiv preprint arXiv:2308.07234*, 2023.

Chen Min, Dawei Zhao, Liang Xiao, Jian Zhao, Xinli Xu, Zheng Zhu, Lei Jin, Jianshu Li, Yulan Guo, Junliang Xing, et al. Driveworld: 4d pre-trained scene understanding via world models for autonomous driving. In *CVPR*, 2024.

Dhruv Shah, Ajay Sridhar, Arjun Bhorkar, Noriaki Hirose, and Sergey Levine. Gnm: A general navigation model to drive any robot. In *ICRA*, 2023a.

Dhruv Shah, Ajay Sridhar, Nitish Dashora, Kyle Stachowicz, Kevin Black, Noriaki Hirose, and Sergey Levine. Vint: A foundation model for visual navigation. In *CoRL*, 2023b.

Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. Lmdrive: Closed-loop end-to-end driving with large language models. In *CVPR*, 2024.

Daeun Song, Jing Liang, Amirreza Payandeh, Amir Hossain Raj, Xuesu Xiao, and Dinesh Manocha. Vlm-social-nav: Socially aware robot navigation through scoring using vision-language models. *IEEE RA-L*, 2024.

Ajay Sridhar, Dhruv Shah, Catherine Glossop, and Sergey Levine. Nomad: Goal masked diffusion policies for navigation and exploration. In *ICRA*, 2024.

Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*, 2024.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Jiahao Wang, Zhenpei Yang, Yijing Bai, Yingwei Li, Yuliang Zou, Bo Sun, Abhijit Kundu, Jose Lezama, Luna Yue Huang, Zehao Zhu, et al. Drive&gen: Co-evaluating end-to-end driving and video generation models. In *IROS*, 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*, 2026a.

Linhan Wang, Zichong Yang, Chen Bai, Guoxiang Zhang, Xiaotong Liu, Xiaoyin Zheng, Xiao-Xiao Long, Chang-Tien Lu, and Cheng Lu. Drive-jepa: Video jepa meets multimodal trajectory distillation for end-to-end driving. *arXiv preprint arXiv:2601.22032*, 2026b.

Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. Omnidrive: A holistic vision-language dataset for autonomous driving with counterfactual reasoning. In *CVPR*, 2025b.

Yan Wang, Wenjie Luo, Junjie Bai, Yulong Cao, Tong Che, Ke Chen, Yuxiao Chen, Jenna Diamond, Yifan Ding, Wenhao Ding, et al. Alpamayo-r1: Bridging reasoning and action prediction for generalizable autonomous driving in the long tail. *arXiv preprint arXiv:2511.00088*, 2025c.

Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In *CVPR*, 2024.

Meng Wei, Chenyang Wan, Xiqian Yu, Tai Wang, Yuqiang Yang, Xiaohan Mao, Chenming Zhu, Wenzhe Cai, Hanqing Wang, Yilun Chen, et al. Streamvln: Streaming vision-and-language navigation via slowfast context modeling. *arXiv preprint arXiv:2507.05240*, 2025.

Meng Wei, Chenyang Wan, Jiaqi Peng, Xiqian Yu, Yuqiang Yang, Delin Feng, Wenzhe Cai, Chenming Zhu, Tai Wang, Jiangmiao Pang, and Xihui Liu. Ground slow, move fast: A dual-system foundation model for generalizable vision-language navigation. In *ICLR*, 2026.

Yi Wei, Linqing Zhao, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *ICCV*, 2023.

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

Katharina Winter, Mark Azer, and Fabian B Flohr. Bevdriver: Leveraging bev maps in llms for robust closed-loop driving. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 20379–20385. IEEE, 2025.

Tianze Xia, Yongkang Li, Lijun Zhou, Jingfeng Yao, Kaixin Xiong, Haiyang Sun, Bing Wang, Kun Ma, Hangjun Ye, Wenyu Liu, et al. Drivelaw: Unifying planning and video generation in a latent driving world. In *CVPR*, 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.

Jiazhi Yang, Shenyuan Gao, Yihang Qiu, Li Chen, Tianyu Li, Bo Dai, Kashyap Chitta, Penghao Wu, Jia Zeng, Ping Luo, et al. Generalized predictive model for autonomous driving. In *CVPR*, 2024a.

Jiazhi Yang, Kashyap Chitta, Shenyuan Gao, Long Chen, Yuqian Shao, Xiaosong Jia, Hongyang Li, Andreas Geiger, Xiangyu Yue, and Li Chen. Resim: Reliable world simulation for autonomous driving. *arXiv preprint arXiv:2506.09981*, 2025a.

Pengxuan Yang, Ben Lu, Zhongpu Xia, Chao Han, Yinfeng Gao, Teng Zhang, Kun Zhan, XianPeng Lang, Yupeng Zheng, and Qichao Zhang. Worldrft: Latent world model planning with reinforcement fine-tuning for autonomous driving. In *AAAI*, 2026.

Yu Yang, Jianbiao Mei, Yukai Ma, Siliang Du, Wenqing Chen, Yijie Qian, Yuxiang Feng, and Yong Liu. Driving in the occupancy world: Vision-centric 4d occupancy forecasting and planning via world models for autonomous driving. In *AAAI*, 2025b.

Zetong Yang, Li Chen, Yanan Sun, and Hongyang Li. Visual point cloud forecasting enables scalable autonomous driving. In *CVPR*, 2024b.

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. In *AAAI*, 2026.

Quanfu Yu, Xian Wu, Hao Xu, and Liulong Ma. Hyworldvla: A vision-language-action model with hybrid world modeling for autonomous driving. *arXiv preprint arXiv:2607.20988*, 2026.

Chengran Yuan, Zhanqi Zhang, Jiawei Sun, Shuo Sun, Zefan Huang, Christina Dao Wen Lee, Dongen Li, Yuhang Han, Anthony Wong, Keng Peng Tee, et al. Drama: An efficient end-to-end motion planner for autonomous driving with mamba. *arXiv preprint arXiv:2408.03601*, 2024.

Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, Xing Wei, and Ning Guo. Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving. *arXiv preprint arXiv:2505.17685*, 2025.

Bozhou Zhang, Nan Song, Jingyu Li, Xiatian Zhu, Jiankang Deng, and Li Zhang. Future-aware end-to-end driving: Bidirectional modeling of trajectory planning and scene evolution. In *NeurIPS*, 2025a.

Jiazhao Zhang, Kunyu Wang, Shaoan Wang, Minghan Li, Haoran Liu, Songlin Wei, Zhongyuan Wang, Zhizheng Zhang, and He Wang. Uni-navid: A video-based vision-language-action model for unifying embodied navigation tasks. *arXiv preprint arXiv:2412.06224*, 2024a.

Jiazhao Zhang, Kunyu Wang, Rongtao Xu, Gengze Zhou, Yicong Hong, Xiaomeng Fang, Qi Wu, Zhizheng Zhang, and He Wang. Navid: Video-based vlm plans the next step for vision-and-language navigation. *RSS*, 2024b.

Jiazhao Zhang, Anqi Li, Yunpeng Qi, Minghan Li, Jiahang Liu, Shaoan Wang, Haoran Liu, Gengze Zhou, Yuze Wu, Xingxing Li, et al. Embodied navigation foundation model. *arXiv preprint arXiv:2509.12129*, 2025b.

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 *ICCV*, 2025c.

Songyan Zhang, Wenhui Huang, Zhan Chen, Chua Jiahao Collister, Qihang Huang, and Chen Lv. Openread: Reinforced open-ended reasoning for end-to-end autonomous driving with llm-as-critic. *arXiv preprint arXiv:2512.01830*, 2025d.

Zhida Zhao, Talas Fu, Yifan Wang, Lijun Wang, and Huchuan Lu. From forecasting to planning: Policy world model for collaborative state-action prediction. In *NeurIPS*, 2025.

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*, 2024.

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 *ICCV*, 2025.

Yufeng Zhong, Chengjian Feng, Feng Yan, Fanfan Liu, Liming Zheng, and Lin Ma. Robotron-nav: A unified framework for embodied navigation integrating perception, planning, and prediction. In *ICCV*, 2025.

Gengze Zhou, Yicong Hong, Zun Wang, Xin Eric Wang, and Qi Wu. Navgpt-2: Unleashing navigational reasoning capability for large vision-language models. In *ECCV*, 2024.

Xin Zhou, Dingkang Liang, Sifan Tu, Xiwu Chen, Yikang Ding, Dingyuan Zhang, Feiyang Tan, Hengshuang Zhao, and Xiang Bai. Hermes: A unified self-driving world model for simultaneous 3d scene understanding and generation. In *ICCV*, 2025a.

Zewei Zhou, Tianhui Cai, Yun Zhao, Seth Z.and 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. *NeurIPS*, 2025b.

Sicheng Zuo, Yuxuan Li, Wenzhao Zheng, Zheng Zhu, Jie Zhou, and Jiwen Lu. Vega: Learning to drive with natural language instructions. *arXiv preprint arXiv:2603.25741*, 2026.

# APPENDIX

## CONTENTS

**A Additional implementation details** **16**

A.1 Training and inference settings . . . . . 16

A.2 Spatial geometry configuration . . . . . 17

A.3 Evaluation metrics . . . . . 18

**B Real-world deployment** **18**

B.1 Real-world evaluation . . . . . 18

**C Additional ablation studies** **20**

C.1 Loss weight sensitivity . . . . . 20

C.2 Denoising steps and efficiency trade-off . . . . . 20

**D Evaluation of world and geometry prediction** **20**

D.1 Future video prediction . . . . . 20

D.2 Future depth prediction . . . . . 21

D.3 Future occupancy prediction . . . . . 21

**E Extension to richer observation contexts** **21**

E.1 Multi-view observations on navtest. . . . . 21

E.2 Historical observations on navtest. . . . . 22

**F Additional NAVSIM-v1 results** **22**

**G Additional qualitative results and failure case** **22**

**H Extended related work** **23**

## A ADDITIONAL IMPLEMENTATION DETAILS

### A.1 TRAINING AND INFERENCE SETTINGS

We adopt Wan2.2-TI2V-5B (Wan et al., 2025) as the video generation expert (VGE), which consists of 30 transformer layers with a hidden dimension of 3072 and 24 attention heads. The action expert (AE) follows the same layer depth with a reduced hidden dimension of 1024 and an FFN dimension of 4096. Both branches are optimized under the flow-matching formulation with 1,000 training timesteps and a scheduler shift of 5.0.

Unless otherwise specified, we train the model for 24k steps with a learning rate of $1 \times 10^{-4}$ and a weight decay of $1 \times 10^{-2}$. The per-device batch size is 16 with 2 gradient accumulation steps, and we use 8 data-loading workers. No image augmentation is applied. During inference, we use 10 denoising steps by default. All experiments are conducted on 8 NVIDIA H200 GPUs.

Table 8: **Training status of model components.** The depth head is initialized from Depth Anything V3 and jointly fine-tuned with the remaining trainable modules.

| Component | Training Status |
|---|---|
| Video VAE | Frozen |
| T5 text encoder | Frozen |
| VGE transformer | Trainable |
| Action Expert | Trainable |
| Ego-state / action encoders | Trainable |
| Depth spatial registers | Trainable |
| Occupancy spatial registers | Trainable |
| Geometry projection layers | Trainable |
| Depth prediction head | Trainable |
| Occupancy prediction head | Trainable |

### A.2 Spatial geometry configuration

We supervise both depth and occupancy over four future timestamps at 1-second intervals within the 4-second prediction horizon. Spatial geometry blocks are inserted after the 11th, 13th, 15th, and 17th layers of the MoT backbone.

**Depth configuration.** For each future timestep, we employ a $10 \times 12$ perspective register grid aligned with the $320 \times 384$ depth target, such that each register corresponds to a $32 \times 32$ image region. This results in 120 registers per timestep and 480 depth registers in total, with a hidden dimension of 1024. At the selected backbone layers, the depth registers perform self-attention with spatiotemporal positional encoding, followed by cross-attention to the current-observation Video representations and noisy Action states. The register states from the selected layers are projected to the feature dimension required by the pretrained DA3 geometry head, which decodes four $320 \times 384$ future metric depth maps.

To construct dense metric depth targets, LiDAR points are transformed into the `CAM_F0` coordinate system and projected onto the image plane using the calibrated camera intrinsics. A pixel-level z-buffer is applied to retain the nearest valid measurement, and the resulting sparse metric depths are aggregated using the median within the same $10 \times 12$ grid. In parallel, a frozen Depth Anything V3 (Lin et al., 2025) predicts dense monocular depth from the corresponding RGB frame. The prediction is resized to $320 \times 384$, aggregated using the same grid partition, and aligned to metric scale using a robust per-frame median ratio computed from valid LiDAR-DA3 correspondences. Frames with fewer than 12 valid correspondence cells are discarded. The resulting metric depth targets are restricted to $[1, 60]$ m and encoded using normalized log-depth for training. Thus, DA3 provides dense spatial completion, while LiDAR supplies metric calibration.

**Occupancy configuration.** For each future timestep, we employ a $10 \times 10$ BEV register grid, resulting in 100 registers per timestep and 400 occupancy registers in total. At the selected backbone layers, the occupancy registers perform BEV self-attention with 2D positional encoding and cross-attend to the same current-observation Video representations and noisy Action states. The resulting register features are projected to 256-dimensional BEV features and aggregated across the selected layers. For each future horizon, the BEV features are progressively upsampled to $100 \times 100$, followed by a $1 \times 1$ convolution that predicts 16 vertical occupancy logits at each BEV location, producing a $100 \times 100 \times 16$ occupancy volume.

Binary occupancy targets are constructed directly from LiDAR observations. Each LiDAR sweep is transformed into the current ego coordinate system and voxelized within a fixed region of interest of $[0, -25, -4, 50, 25, 4]$ m using a $0.5$ m voxel size. The resulting ground-plane region covers $50\,\mathrm{m} \times 50\,\mathrm{m}$, spanning $[-25, 25]$ m laterally and $[0, 50]$ m longitudinally. A voxel is labeled as occupied if it contains at least one LiDAR point. Since no explicit ray tracing is performed, voxels without LiDAR returns are treated as pseudo-empty candidates rather than explicitly observed free space. During training, all valid occupied voxels are retained, while pseudo-empty voxels are sampled with a negative-to-positive ratio of 2:1, capped at 32,768 negative voxels per timestep. A route-

corridor validity mask is further applied to determine valid supervision regions. The occupancy representation is class-agnostic and is constructed independently of the monocular depth prediction.

The depth and occupancy branches use independent parameters and remain mutually isolated throughout training. Action-conditioning dropout is set to 0.2. Although the register branches do not write back to the video or action streams in the forward pass, their geometry objectives back-propagate through the queried representations during training. Both branches are entirely removed during action inference.

**Urban navigation training details.** For the CityWalker experiments, we initialize the video generation expert from the pretrained Wan2.2-TI2V-5B model and fine-tune Metis on the CityWalker training split. Since CityWalker does not provide LiDAR observations, occupancy supervision is disabled. We retain the depth branch and construct dense depth supervision using a frozen Depth Anything 3 (DA3) model applied to the corresponding RGB observations, without LiDAR-based metric calibration. Except for the task-specific action representation and training schedule described in Sec. 4.1, the remaining model architecture and training procedure follow the autonomous-driving setting. The models used for the different navigation evaluations should also be distinguished. The zero-shot Metis result in Fig. 9 directly uses the checkpoint trained on the autonomous-driving task, without any CityWalker fine-tuning. In contrast, the fine-tuned CityWalker result is obtained by training on the CityWalker training split. The Unitree Go2 deployment uses the CityWalker-fine-tuned checkpoint, without additional training on the deployment environments.

### A.3 Evaluation metrics

We evaluate autonomous driving performance on the navhard and navtest benchmarks of NAVSIMv2 (Cao et al., 2025). navhard follows a two-stage protocol, consisting of 244 safety-critical real-world scenarios in Stage 1 and 4,164 corresponding 3DGS-generated scenarios in Stage 2. In contrast, navtest contains 12,146 real-world scenarios and evaluates generalization across diverse driving conditions. Both benchmarks adopt the Extended Predictive Driver Model Score (EPDMS) (Li et al., 2025b):

$$\mathrm{EPDMS} = \left( \prod_{m \in \mathcal{M}*\mathrm{pen}} S_m \right) \left( \frac{\sum *m \in \mathcal{M} * \mathrm{avg} w_m S_m}{\sum *m \in \mathcal{M} * \mathrm{avg} w_m} \right), \tag{10}$$

where $\mathcal{M} * \mathrm{pen}$ contains the penalty metrics No-at-fault Collisions (NC), Drivable Area Compliance (DAC), Driving Direction Compliance (DDC), and Traffic Light Compliance (TLC), while $\mathcal{M}_{\mathrm{avg}}$ contains Time-to-Collision (TTC), Ego Progress (EP), Lane Keeping (LK), History Comfort (HC), and Extended Comfort (EC). The navhard protocol further incorporates two-stage aggregation, reactive traffic simulation, and human-failure-aware penalty handling. We additionally report results on NAVSIMv1 navtest (Dauner et al., 2024) using the Predictive Driver Model Score (PDMS):

$$\mathrm{PDMS} = \left( \prod_{m \in \mathrm{NC,DAC}} S_m \right) \left( \frac{\sum_{m \in \mathrm{EP,TTC,C}} w_m S_m}{\sum_{m \in \mathrm{EP,TTC,C}} w_m} \right), \tag{11}$$

where C denotes the comfort metric.

## B Real-world deployment

### B.1 Real-world evaluation

We further evaluate Metis in closed-loop obstacle-avoidance experiments under two real-world conditions: indoor nighttime and outdoor daytime. Figure 5 and Figure 6 present four representative examples, where the quadruped robot successfully plans collision-free paths around obstacles under different appearances and illumination conditions. These results qualitatively demonstrate the ability of Metis to transfer its learned planning capability to diverse real-world environments. We additionally conduct 15 trials under each condition and report the success rates in Table 9. Metis achieves a success rate of 53.3% indoors and 66.7% outdoors, providing further evidence of its real-world deployment capability.

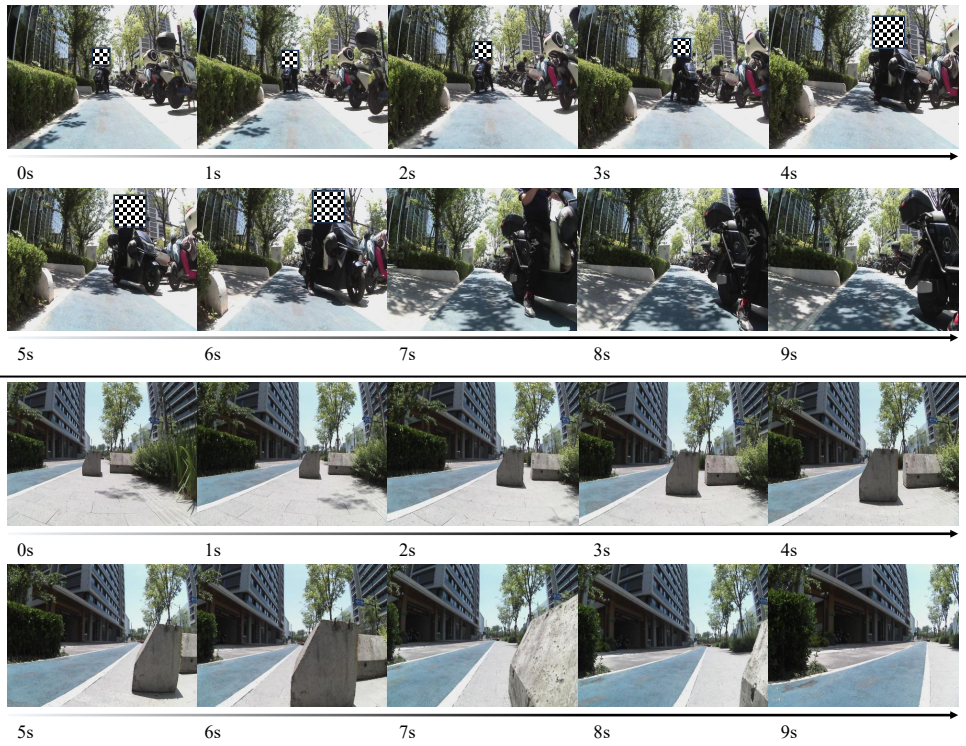



Figure 5: Qualitative results of real-world outdoor deployment under a zero-shot setting. Sensitive biometric information (e.g., human faces) has been anonymized.

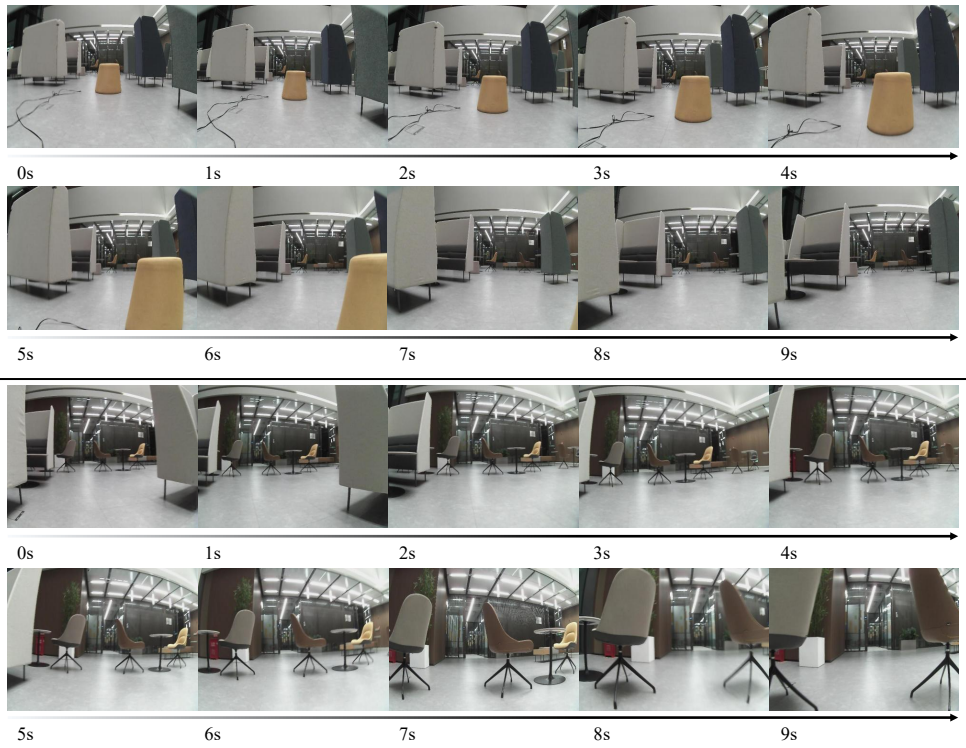



Figure 6: Qualitative results of real-world indoor deployment under a zero-shot setting (without any task-specific training)

Table 9: **Real-world closed-loop evaluation.** Success rates are reported over 15 trials for each scenario.

| **Method** | **Indoor** | **Outdoor** |
|---|---|---|
| Metis (Ours) | 53.3% (8/15) | 66.7% (10/15) |

Table 11: Performance efficiency trade-off with different denoising steps on navtest.

| Steps | DAC ↑ | EC ↑ | EPDMS ↑ | Latency (s) ↓ | FPS ↑ |
|---|---|---|---|---|---|
| 1 | 96.8 | 87.1 | 89.1 | **0.10** | **10.00** |
| 2 | 97.5 | 87.4 | 90.0 | 0.14 | 7.14 |
| 5 | 97.6 | 87.7 | 90.1 | 0.26 | 3.84 |
| 10 | **97.9** | **88.2** | **90.2** | 0.44 | 2.27 |

# C Additional ablation studies

## C.1 Loss weight sensitivity

Table 10: **Sensitivity to loss weights on navtest.** We vary one loss weight at a time while keeping the others fixed at their default values: $\lambda_{\text{action}} = 1$, $\lambda_{\text{video}} = 4$, $\lambda_{\text{dep}} = 10$, and $\lambda_{\text{occ}} = 1$.

$\lambda_{\text{video}}$

| Value | EPDMS ↑ |
|---|---|
| 2 | 89.6 |
| **4** | 90.2 |
| 8 | 90.0 |

$\lambda_{\text{dep}}$

| Value | EPDMS ↑ |
|---|---|
| 5 | 90.1 |
| **10** | 90.2 |
| 15 | 89.7 |

$\lambda_{\text{occ}}$

| Value | EPDMS ↑ |
|---|---|
| 0.5 | 89.5 |
| **1** | 90.2 |
| 2 | 89.9 |

**Sensitivity to loss weights.** Table 10 studies the sensitivity to the video, depth, and occupancy loss weights by varying one coefficient at a time. The overall performance remains stable across different settings, indicating that Metis is not sensitive to the exact choice of loss weights. We therefore adopt $\lambda_{\text{video}} = 4$, $\lambda_{\text{dep}} = 10$, and $\lambda_{\text{occ}} = 1$ as the default configuration.

## C.2 Denoising steps and efficiency trade-off

Table 11 reveals a clear performance–efficiency trade-off with respect to the number of action denoising steps. Increasing the number of steps consistently improves DAC, EC, and EPDMS, but the performance gains gradually saturate beyond two steps. The default 10-step setting achieves the best overall performance with 90.2 EPDMS, whereas reducing the process to only 2 steps retains a competitive 90.0 EPDMS while decreasing inference latency from 0.44,s to 0.14,s, corresponding to a $3.1\times$ speedup. We therefore adopt 10 steps for benchmark evaluation to maximize planning quality, while the 2-step configuration provides a favorable operating point for latency-sensitive real-world deployment.

# D Evaluation of world and geometry prediction

We further evaluate the quality of the future world and geometric predictions learned during training. These auxiliary predictions are not required at inference time; instead, the following experiments verify that they provide meaningful supervision for learning future dynamics and spatial geometry.

## D.1 Future video prediction

We first evaluate the quality of future video prediction using standard image- and video-generation metrics. We additionally report the results of PWM (Zhao et al., 2025) for reference. Since the two methods are evaluated under their respective evaluation settings, the results should be interpreted

Table 12: **Future video prediction quality.** "–" denotes unavailable results. Results are reported under their respective evaluation settings. PWM uses an input resolution of $128 \times 224$, while Metis-uses $320 \times 384$.

| Method | PSNR ↑ | SSIM ↑ | LPIPS ↓ | FVD ↓ |
|---|---|---|---|---|
| PWM (Zhao et al., 2025) | **21.57** | – | 0.23 | 85.95 |
| Metis | 19.75 | 0.538 | **0.20** | **56.5** |

Table 13: **Future depth prediction over different horizons.** Mean denotes the average over the four future timestamps.

| Metric | 1s | 2s | 3s | 4s | Mean |
|---|---|---|---|---|---|
| AbsRel ↓ | 0.108 | 0.131 | 0.152 | 0.172 | 0.141 |
| RMSE ↓ | 5.44 | 6.15 | 6.74 | 7.24 | 6.39 |
| $\delta_1$ ↑ | 0.886 | 0.856 | 0.831 | 0.806 | 0.845 |

as indicative rather than strictly controlled comparisons. As shown in Table 12, Metis achieves an FVD of 56.5, indicating meaningful future scene modeling. Compared with the reported PWM results, our method obtains a substantially lower FVD, while PWM achieves a higher PSNR. Since the two methods use different input resolutions and evaluation settings, these results are provided as a reference rather than a strictly controlled comparison.

### D.2 Future depth prediction

We evaluate future depth prediction at four timestamps over the 4-second prediction horizon. We report AbsRel, RMSE, and $\delta_1$, which measure relative depth error, absolute metric error, and threshold accuracy, respectively. Table 13 shows a gradual degradation as the prediction horizon increases, which is expected due to the growing uncertainty of future scene evolution. Nevertheless, the model maintains an AbsRel of 0.172 and a $\delta_1$ accuracy of 0.806 even at the 4-second horizon. The smooth temporal degradation indicates that the depth registers capture meaningful future geometric structure rather than only short-term appearance cues.

### D.3 Future occupancy prediction

We further evaluate class-agnostic future occupancy prediction using IoU, precision, recall, and F1 score. We exclude the current-frame prediction and report only the four future timestamps for consistency with future depth evaluation. As shown in Table 14, occupancy prediction remains stable at short horizons and gradually becomes more challenging toward 4 seconds. The model achieves a mean IoU of 0.494 and F1 score of 0.651 over the four future timestamps, while maintaining a recall of 0.733 at 4 seconds. These results indicate that the occupancy registers preserve useful occupied/free-space structure over long-term prediction horizons, providing complementary geometric supervision to dense depth prediction.

## E Extension to richer observation contexts

Although our main experiments focus on the standard single-view, single-frame setting, Metis can be naturally extended to richer spatial and temporal observation contexts. As these extensions are not the primary focus of this work, we provide preliminary results with multi-view and historical observations in the supplementary material to demonstrate the flexibility of our framework.

### E.1 Multi-view observations on navtest.

We first examine whether Metis can benefit from richer spatial observations beyond the monocular setting used in our main experiments. Following common multi-view input construction, we concatenate the left-front, front, and right-front camera images and resize the resulting panorama to

Table 14: **Future occupancy prediction over different horizons.** Mean denotes the average over the four future timestamps.

| Metric | 1s | 2s | 3s | 4s | Mean |
|---|---|---|---|---|---|
| IoU ↑ | 0.542 | 0.520 | 0.483 | 0.432 | 0.494 |
| Precision ↑ | 0.612 | 0.596 | 0.559 | 0.502 | 0.567 |
| Recall ↑ | 0.812 | 0.785 | 0.760 | 0.733 | 0.773 |
| F1 ↑ | 0.696 | 0.676 | 0.642 | 0.592 | 0.651 |

Table 15: **Effect of multi-view observations.**

| **Input** | **Frames** | **navtest** EPDMS ↑ | **navhard** EPDMS ↑ |
|---|---|---|---|
| Front-only | 1 | 90.2 | 33.8 |
| Multi-view | 3 | **90.5** | **34.4** |

Table 16: **Effect of historical observations.**

| **Input** | **Views** | **navtest** EPDMS ↑ | **navhard** EPDMS ↑ |
|---|---|---|---|
| Current only | 1 | 90.2 | 33.8 |
| Current + history | 4 | **90.7** | **34.7** |

$192 \times 768$. As shown in Table 15, extending the input from one to three views improves EPDMS from 90.2 to 90.5 on navtest and from 33.8 to 34.4 on navhard. These consistent gains demonstrate that Metis can readily incorporate broader spatial context without modifying the underlying world-action modeling framework.

### E.2 Historical observations on navtest.

We further investigate the effect of temporal context by augmenting the current observation with historical frames. As shown in Table 16, using four frames improves EPDMS from 90.2 to 90.7 on navtest and from 33.8 to 34.7 on navhard. The larger improvement on navhard suggests that temporal context is particularly beneficial in challenging scenarios, where historical observations provide additional cues about scene dynamics and agent motion. Together, these results show that Metis is not restricted to the default single-view, single-frame setting and can naturally accommodate richer spatial and temporal observations.

## F Additional NAVSIM-v1 results

As shown in Table 17, Metis remains competitive on the NAVSIM-v1 navtest benchmark, achieving 89.9 PDMS with the default input resolution and 90.3 PDMS under the high-resolution setting. Compared with prior WAM-based approaches, our method consistently improves over representative baselines such as Epona, PWM, DriveVLA-W0, UniWorldVLA, and DriveLAW, while using only a single front-facing camera. These results further demonstrate the robustness of our framework across different NAVSIM evaluation protocols.

## G Additional qualitative results and failure case

**Qualitative results.** For the AD task, Metis produces reasonable planning behaviors across diverse scenarios, including lane following and turning, as illustrated in Figure 7. In more complex open-world scenarios, the model can still complete the intended maneuver, although the predicted trajectory may not perfectly follow fine-grained route descriptions, as shown in Figure 8. This suggests a promising direction for incorporating stronger language understanding and reasoning capabilities from large-scale multimodal models to further improve instruction-grounded planning. For the UN task, as shown in Figure 9, Metis demonstrates stronger zero-shot generalization than Epona across diverse navigation environments.

We further visualize future video and spatial geometry predictions in a challenging turning scenario, as shown in Figure 10. Across the 1-4s prediction horizon, the generated frames exhibit coherent scene evolution along the turning trajectory. Meanwhile, the predicted depth and occupancy preserve the major road layout and surrounding geometric structures, showing clear consistency with their

Table 17: **Performance comparison on NAVSIM-v1 navtest**. * indicates training with reinforcement learning; For our method, † denotes $640 \times 768$ inputs; the default setting uses $320 \times 384$.

| Method | Sensors | NC↑ | DAC↑ | EP↑ | TTC↑ | C↑ | PDMS↑ |
|---|---|---|---|---|---|---|---|
| Human Agent | - | 100 | 100 | 87.5 | 100 | 99.9 | 94.8 |
| *Traditional E2E-based Planner* | | | | | | | |
| VADv2-$\mathcal{V}_{8192}$ (Chen et al., 2024a) | 3xC+L | 97.2 | 89.1 | 76.0 | 91.6 | **100.0** | 80.9 |
| UniAD (Hu et al., 2023) | 6xC | 97.8 | 91.9 | 78.8 | 92.9 | **100.0** | 83.4 |
| Bevdriver (Winter et al., 2025) | 4xC+L | 97.7 | 92.5 | 78.7 | 92.9 | **100.0** | 83.8 |
| LTF (Chitta et al., 2022) | 3xC | 97.4 | 92.8 | 79.0 | 92.4 | 100 | 83.8 |
| TransFuser (Chitta et al., 2022) | 3xC+L | 97.7 | 92.8 | 79.2 | 92.8 | **100.0** | 84.0 |
| PARA-Drive (Weng et al., 2024) | 6xC | 97.9 | 92.4 | 79.3 | 93.0 | 99.8 | 84.0 |
| LAW (Li et al., 2025c) | 1xC | 96.4 | 95.4 | 81.7 | 88.7 | 99.9 | 84.6 |
| World4Drive (Zheng et al., 2025) | 3xC | 97.4 | 94.3 | 79.9 | 92.8 | **100.0** | 85.1 |
| DRAMA (Yuan et al., 2024) | 3xC+L | 98.0 | 93.1 | 80.1 | 94.8 | **100.0** | 85.5 |
| Hydra-MDP++ (Li et al., 2025b) | 3xC+L | 97.6 | 96.0 | 80.4 | 93.1 | **100.0** | 86.6 |
| ARTEMIS (Feng et al., 2025) | 3xC+L | 98.3 | 95.1 | 81.4 | 94.3 | **100.0** | 87.0 |
| WorldRFT* (Yang et al., 2026) | 3xC | 97.5 | 96.0 | 80.9 | 94.0 | **100.0** | 87.0 |
| DiffusionDrive (Liao et al., 2025) | 3xC+L | 98.2 | 96.2 | 82.2 | 94.7 | **100.0** | 88.1 |
| WorldDrive (Gui et al., 2026) | 1xC | 98.4 | 96.2 | 81.9 | 95.1 | **100** | 88.1 |
| WoTE (Li et al., 2025e) | 3xC+L | 98.5 | 96.8 | 81.9 | 94.9 | 99.9 | 88.3 |
| SeerDrive (Zhang et al., 2025a) | 3xC+L | 98.4 | 97.0 | 83.2 | 94.9 | 99.9 | 88.9 |
| Drive-JEPA (Wang et al., 2026b) | 1xC | 98.7 | 96.2 | 82.9 | **100.0** | 95.5 | 89.0 |
| DriveSuprim (Yao et al., 2026) | 3xC | 97.8 | 97.3 | 86.7 | 93.6 | **100.0** | 89.9 |
| *VLA-based Planner* | | | | | | | |
| Vega (Zuo et al., 2026) | 1xC | **98.9** | 95.3 | 81.6 | 96.1 | **100.0** | 87.9 |
| AutoVLA* (Zhou et al., 2025b) | 3xC | 98.4 | 95.6 | 81.9 | 98.0 | 99.9 | 80.5 |
| AdaThinkDrive* (Luo et al., 2025) | 1xC | 98.4 | 97.8 | 84.4 | 95.2 | **100.0** | 90.3 |
| HyWorldVLA* (Yu et al., 2026) | 1xC | 98.6 | 98.9 | 84.6 | 95.1 | 99.3 | 90.6 |
| ReCogDrive* (Li et al., 2025f) | 1xC | 97.9 | 97.3 | **87.3** | 94.9 | **100.0** | 90.8 |
| EFL-VLA (Luo et al., 2026) | 1xC | **98.9** | 98.1 | 85.3 | 96.0 | **100.0** | 91.0 |
| SGDrive* (Li et al., 2026b) | 1xC | 98.6 | 97.8 | 85.8 | 96.2 | **100.0** | 91.1 |
| *WAM-based Planner* | | | | | | | |
| Epona (Zhang et al., 2025c) | 1xC | 97.9 | 95.1 | 80.4 | 93.8 | 99.9 | 86.2 |
| Resim (Yang et al., 2025a) | 1xC | - | - | - | - | - | 86.6 |
| ImagiDrive (Li et al., 2025a) | 1xC | 98.6 | 96.2 | 80.5 | 94.5 | **100.0** | 87.4 |
| PWM (Zhao et al., 2025) | 1xC | 98.6 | 95.9 | 81.8 | 95.4 | **100.0** | 88.1 |
| DriveVLA-W0 (Li et al., 2025d) | 1xC | 98.7 | 96.2 | 82.2 | 95.5 | **100.0** | 88.4 |
| UniWorldVLA (Liu et al., 2026b) | 1xC | 98.7 | 96.7 | 83.2 | 96.1 | **100.0** | 89.4 |
| DriveLAW (Xia et al., 2026) | 1xC | 99.0 | 97.1 | 81.3 | 96.7 | **100.0** | 89.1 |
| **Metis (Ours)** | 1xC | 98.4 | 97.8 | 84.1 | 95.1 | **100.0** | 89.9 |
| **Metis † (Ours)** | 1xC | 98.6 | **98.1** | 84.4 | 95.4 | **100.0** | 90.3 |

corresponding ground-truth targets. These results provide qualitative evidence that the proposed spatial geometry module captures meaningful future spatial structure in addition to visual dynamics, providing spatially grounded representations for downstream action prediction and planning.

## H Extended related work

**Video generation and geometry-aware world modeling.** Recent video foundation models and predictive driving models provide increasingly strong spatiotemporal priors for modeling future scene evolution (Kim et al., 2026; Wan et al., 2025; Yang et al., 2024a). In parallel, geometry-aware driving approaches incorporate explicit depth supervision, future point-cloud prediction, and spatiotemporal 3D representations to learn spatially grounded scene representations (Yang et al., 2024b; Min et al., 2024; 2023). Occupancy-based methods further model fine-grained 3D scene structure and future evolution through dense occupancy prediction and forecasting (Zheng et al., 2024; Yang et al., 2025b; Wei et al., 2023). These studies highlight the complementary roles of visual dynamics and explicit geometric supervision in learning physically grounded representations for perception and planning.



Figure 7: Qualitative results of Metis in NAVSIM.



Figure 8: Qualitative failure results of Metis in NAVSIM.




Figure 9: Qualitative results on CityWalker. We present the zero-shot results of Epona and our method (Metis), as well as the fine-tuned results of our method.

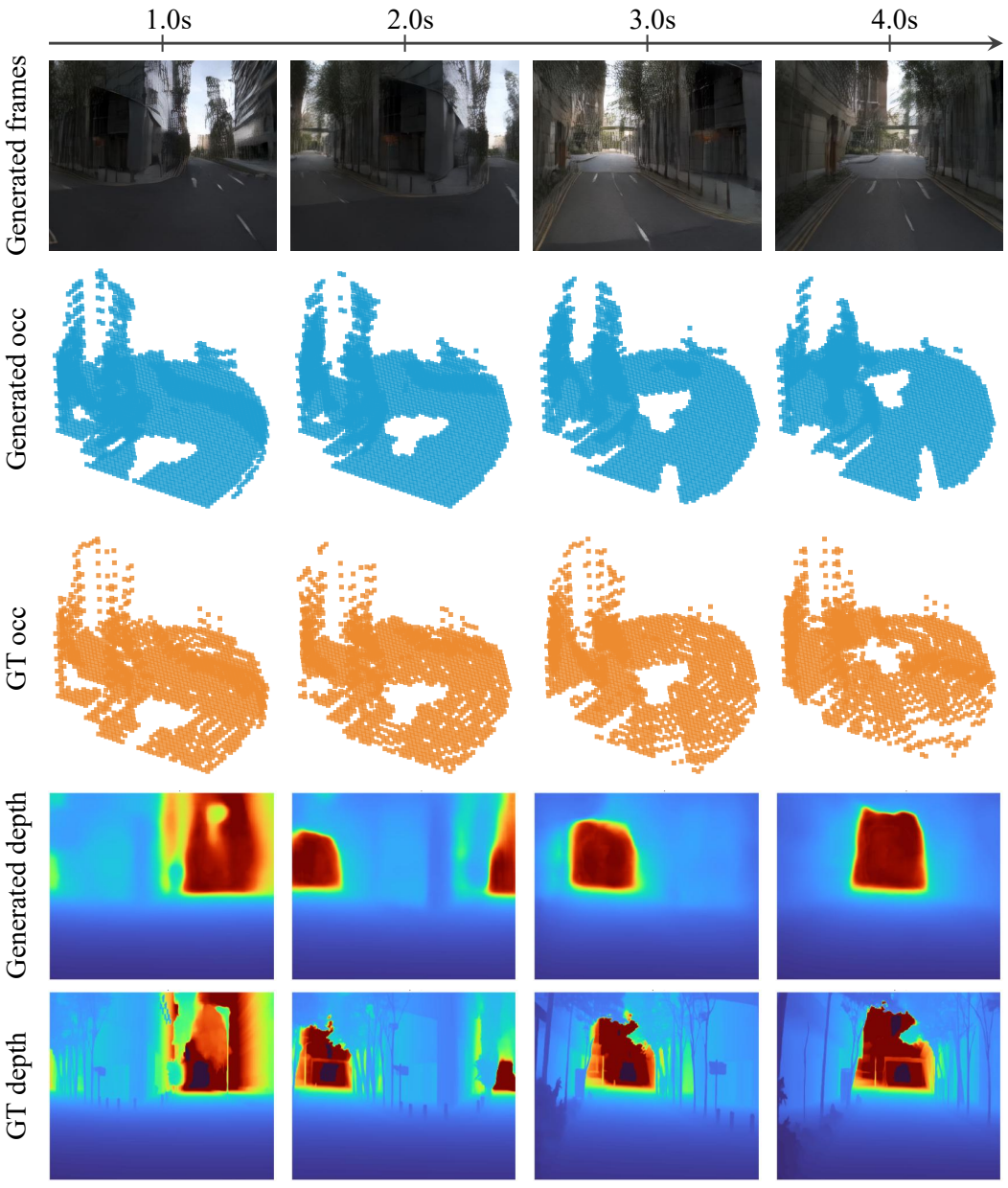



Figure 10: Qualitative visualization of future world and geometry prediction in a turning scenario. Columns correspond to future horizons from 1 to 4 s. Rows show generated frames, predicted and ground-truth occupancy, and predicted and ground-truth depth, respectively.