OSWAM: AN EFFECTIVE WORLD ACTION MODEL WITH ONE-STEP ACTION GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Direct-policy World Action Models (WAM) have demonstrated the benefits of decoupling action generation from explicit future visual prediction for efficient inference in autonomous driving and embodied control. However, they still require tens of denoising steps to achieve competitive action quality, incurring substantial latency and hindering real-time deployment. Simply reducing the number of denoising steps will collapse the nonlinear multi-step denoising trajectory into coarse updates, substantially degrading action quality. To reduce the denoising steps while enabling real-time action generation, we present an effective world action model with one-step action generation, namely **OSWAM**, for autonomous driving and embodied control. We argue that it is the absence of structured action priors in random noise initialization that forces direct-policy WAMs to recover coherent action sequences through iterative denoising. Hence, we construct imagination tokens to initialize the diffusion process, providing physical priors and action-conditioned representations. We further introduce privileged information distillation with future conditioning to enhance future-aware action generation. As a result, OSWAM can efficiently and effectively generate high quality action in just one diffusion step. Our experiments demonstrate that OSWAM achieves comparable or even better results, in terms of both autonomous driving and embodied control, than direct-policy WAM and Joint-WAM that require tens of steps. It achieves approximately $3\times$ speedup over direct-policy WAM baselines and supports real-time compiled inference with latency as low as 40ms on an NVIDIA RTX 5090 GPU. Codes and models will be publicly available.

1 INTRODUCTION

Physical intelligence (Agarwal et al., 2025; 2026), including autonomous driving (Chen et al., 2024) and embodied control (Yuan et al., 2026), holds the promise of freeing human workers from physical tasks that are dangerous, laborious, or tedious. A general-purpose physical agent should not only act from visual observations, but also anticipate the physical consequences of their actions. Existing Vision-Language-Action (VLA) models, including those standard models (Black et al., 2024; Intelligence et al., 2025; Li et al., 2026e) and specialized end-to-end ones (Liao et al., 2025; Chi et al., 2025; Li et al., 2025b), typically predict actions directly without explicitly modeling the future evolution of the environment. This has motivated a shift toward World Action Models (WAMs) (Yuan et al., 2026; Zhao et al., 2026c; Li et al., 2026d; Liu et al., 2026a) for physical AI, which build on pretrained video generative models (Wan et al., 2025; Team et al., 2026) to jointly predict future visual states and their corresponding actions.

However, WAMs follow the imagine-then-execute paradigm (Bi et al., 2026; Li et al., 2026d) and incur prohibitive inference latency, rendering them impractical for latency-critical applications such as autonomous driving. Borrowing few-step techniques from image and video generation, some methods apply step distillation (Akbari et al., 2026) to reduce this latency, but often at the expense of a substantial drop in control accuracy, as illustrated in Fig. 1. One possible strategy is to switch from the imagine-then-execute paradigm to direct-policy WAMs (Yuan et al., 2026; Zhao et al., 2026c) to address this issue. These methods retain video prediction as a training-time objective that shapes physically grounded representations, without synthesizing future observations at test time. This direct-policy formulation is more than $3\times$ faster than the imagine-then-execute paradigm.

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

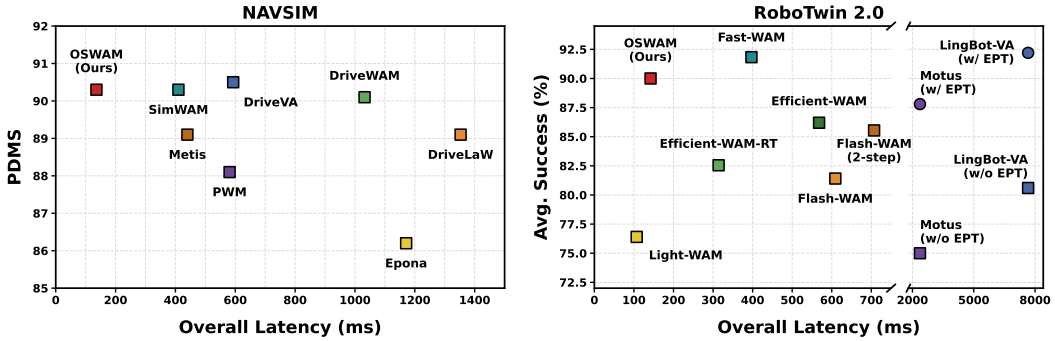

Figure 1: NAVSIM and Robotwin 2.0 inference efficiency-performance comparison. EPT denotes embodied pretraining. All latencies are measured on NVIDIA H20 GPU.

Despite this speedup, direct-policy WAMs still require tens of denoising steps to achieve competitive action quality, whose latency remains too high for real-time applications such as autonomous driving. This bottleneck cannot be resolved by reducing the number of denoising steps. Specifically, as shown in Fig. 2, the performance degrades as the step count decreases, dropping by 20% on NAVSIM (autonomous driving) and 40% on RoboTwin (embodied control) in the one-step setting. We attribute this to the lack of action structure in random noise: multiple denoising steps can gradually impose temporal and physical consistency, whereas one-step generation must recover a coherent action sequence in a single update. Existing few-step accelerations, such as step distillation (Akbari et al., 2026) and trajectory rectification (Liu et al., 2022), still start from Gaussian noise and, therefore, do not supply the missing action prior. Action generation is low-dimensional and typically uses far fewer denoising steps than image or video synthesis, so this lightweight change is more appropriate than transplanting distillation designed for high-dimensional visual generation.

In addition, existing methods such as Direct-policy WAMs (Yuan et al., 2026; Zhao et al., 2026c) use a consistent first-frame-only interface during training and inference, relying on video co-training to encode future dynamics into the current representation. However, compressing all future information into a single first-frame representation may provide insufficient temporal context for effective future-aware control. Subsequent methods therefore reintroduce future features at inference, while reducing rollout cost through caching, sparse interaction (Zhao et al., 2026a), or compact latent representations (Li et al., 2026a). Although these designs reduce the cost of future imagination, they still introduce future-dependent features at test time, increasing inference overhead.

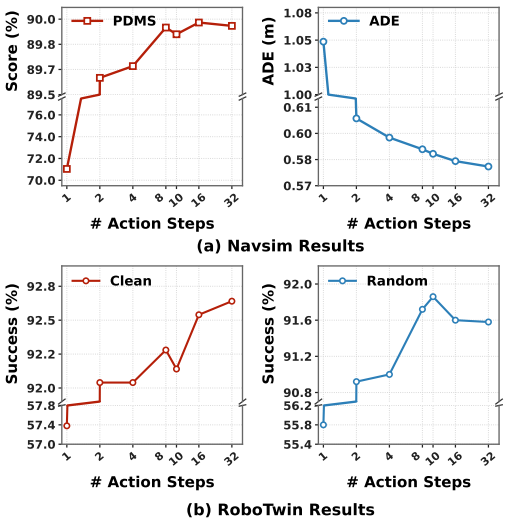

Figure 2: Test-time scaling of Fast-WAM (Yuan et al., 2026) with varying action denoising steps on (a) NAVSIM and (b) RoboTwin.

To address the aforementioned issues, we propose **OSWAM**, a world action model with one-step action generation for more effective physical control. Our key observation is that previous one-step action generation methods fail mainly because the action diffusion processing is initialized from unstructured Gaussian noise. To supply the missing action structure, we learn a compact set of *imagination tokens* from the video Diffusion Transformer (DiT) and use them as structured start points for action denoising. These tokens are far fewer than video first-frame tokens and are obtained in a single forward pass, so they add negligible computation, yet they interact with visual features to accumulate physically grounded, action-relevant context. We then noise the imagination tokens to an intermediate timestep and decode them with the action DiT in one step, replacing the long denoising trajectory required by random initialization. To enhance future-aware control, we train these imagination tokens with privileged-information distillation. A teacher attends to visual

tokens over the full future horizon, while the student imagination tokens, at test time, attend only to the first frame. By distilling the teacher's privileged representations into the imagination tokens and subsequently into the action DiT, OSWAM transfers long-horizon context without reintroducing future-dependent features at inference. This design preserves the first-frame-only interface of direct-policy WAMs, while enabling high-quality action generation in a single denoising step.

Our contributions are summarized as follows:

- We propose OSWAM, a world action model that enables high-quality one-step action generation. It learns compact imagination tokens from the video DiT as structured start points for the action DiT. These tokens are obtained in a single forward pass with negligible overhead and are decoded in one denoising step.
- We introduce privileged-information distillation to transfer long-horizon context into the imagination tokens. A teacher attends to visual tokens over the full future horizon, while the student remains first-frame-only at test time, preserving the interface of direct-policy WAMs without future-dependent features at inference.
- Experiments on autonomous driving and embodied control show that OSWAM achieves competitive results with multi-step WAMs while being substantially faster.

2 Related Work

World Action Models for Autonomous Driving. World models for autonomous driving have recently been extended into world action models (WAMs). Existing methods fall into two categories: bird's-eye view (BEV)-based and video-generation-based world models. BEV-based approaches (Li et al., 2025b; Zheng et al., 2024a; Li et al., 2025c) forecast occupancy evolution and ego motion, predict future latent scene features, or infer future BEV layouts for online trajectory evaluation. They are computationally efficient but do not exploit large-scale video generative priors. A complementary line of research leverages pretrained video generators and predicts the future in the image domain. These methods (Shi et al., 2026; Liu et al., 2026a; Xia et al., 2026) transfer such priors to driving, but jointly rolling out video features at test time remains inefficient. SimWAM (Zhao et al., 2026c) follows the direct-policy design of Fast-WAM (Yuan et al., 2026) to avoid test-time video rollout, yet the remaining multi-step denoising is still impractical for onboard driving. In contrast, our proposed OSWAM inherits large-scale video generative priors without test-time video rollout, and reduces action generation to a single denoising step.

World Action Models for Embodied Control. Unlike autonomous driving, where compact BEV representations naturally encode road topology and traffic interactions, embodied control needs to model fine-grained object interactions and appearance changes in close-range scenes. As these phenomena are naturally captured in video, embodied WAMs are predominantly built on pretrained video generators for image-domain future modeling in two ways: explicit future imagination and direct-policy inference. Explicit-imagination methods (Li et al., 2026d; Bi et al., 2026) generate future visual representations at test time to guide action prediction, incurring substantial rollout latency. Direct-policy WAMs (Yuan et al., 2026) retain video prediction as a training signal but generates actions directly from current observations, avoiding test-time visual rollout; however, its iterative action denoising remains a bottleneck for high-frequency closed-loop control. In contrast, OSWAM preserves large-scale video-generative priors without test-time visual rollout and reduces action generation to a single denoising step.

Effective World Action Models. The main deployment bottleneck for WAMs lies in the high computational cost of diffusion-based video and action generators, whose test-time latency is prohibitive for onboard driving and closed-loop manipulation. Drawing one step distillation from video generation, Flash-WAM (Akbari et al., 2026) compresses joint video-action diffusion to a single denoising step per modality, but still synthesizes future video at inference. Fast-WAM (Yuan et al., 2026) removes test-time video rollout by predicting actions from current observations only, yet retains iterative action denoising. Light-WAM (Li et al., 2026f) further replaces the generative action expert with a lightweight fusion head for single-pass action prediction, but underperforms on challenging benchmarks such as RoboTwin. SimWAM (Zhao et al., 2026c) follows Fast-WAM to drop test-time video rollout, but its multi-step action denoising keeps inference latency well above the budget required for onboard deployment. Building on this video-rollout-free direct-policy formulation, OSWAM addresses the remaining action-generation bottleneck through one-step action denoising.

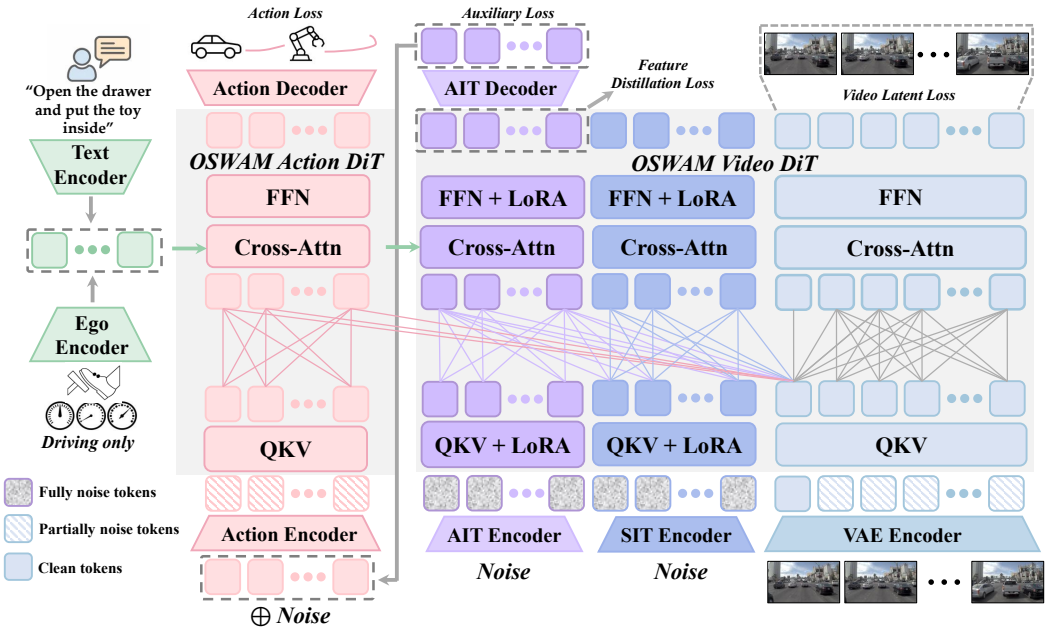

Figure 3: The overall architecture of OSWAM. A single Video DiT pass produces an observation-conditioned action prior with AIT and complementary scene context with SIT, providing a structured initialization for one-step Action DiT denoising. Privileged-information distillation injects future-informed representations during training, while inference uses only the current observation.

3 METHOD

We present OSWAM, a direct-policy world action model that enables high-quality one-step action generation for autonomous driving and embodied control. The preliminary concepts associated with our method are provided in **Appendix A.1**. In the following sections, we first present the overall architecture of OSWAM, then describe in detail the fundamental components of OSWAM, including imagination tokens, Role-aware 3D RoPE, and privileged-information distillation.

3.1 OVERALL ARCHITECTURE

An overview of OSWAM is shown in Fig. 3. As in previous works (Yuan et al., 2026; Zhao et al., 2026c), we consider predicting an action chunk $\mathbf{a}_{1:H}$ from the current frame f_0 and an optional task instruction l, with ego-state information additionally provided for driving tasks. We follow FastWAM in the world model backbone configuration and direct-policy setting. Specifically, the Video DiT uses the Wan2.2-5B backbone (Wan et al., 2025), accompanied by its pretrained T5 text encoder and video VAE. The Action DiT is initialized by interpolating and downsampling the pretrained Video DiT weights. For autonomous driving, we add an ego encoder to encode the ego state and the driving command. The task languages and ego states are both provided to all tokens through cross-attention. The full model is organized as a Mixture-of-Transformer (MoT) architecture with shared attention between the video and action blocks. We also follow the direct-policy protocol: video modeling is used only as a training signal, and future video is not rolled out at inference.

Unlike prior direct-policy methods, which use the Video DiT at inference solely to encode the first-frame visual feature for Action DiT through shared MoT attention, OSWAM additionally introduces two groups of imagination tokens into the Video DiT: *Action Imagination Tokens* (AIT) and *Scene Imagination Tokens* (SIT). As shown in Fig. 3, both AIT and SIT are initialized from noise in the Video DiT and interact with the first-frame visual features. The resulting AIT representation then replaces the unstructured Gaussian initialization of the Action DiT, providing a compact, structured start point for action denoising. SIT provides additional latent slots that aggregate scene context for AIT; they are updated indirectly through privileged distillation and the auxiliary action loss on

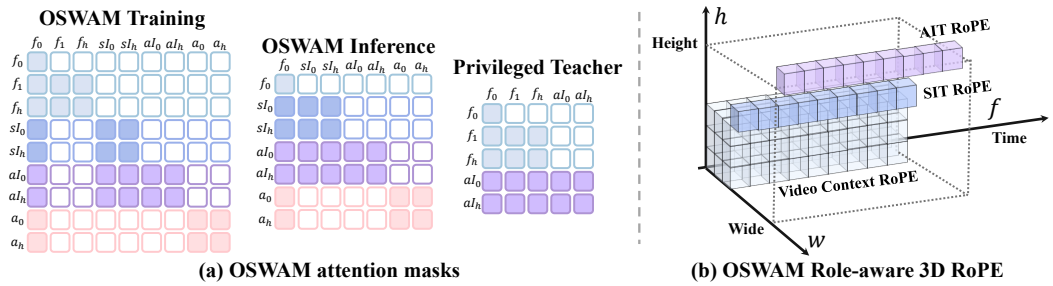

Figure 4: (a) The attention mask of OSWAM and the privileged teacher. (b) Role-aware RoPE explicitly disentangles token groups using unique positional encodings. sI is SIT and aI is AIT.

AIT. Together, AIT and SIT comprise far fewer tokens than the first-frame visual input and therefore introduce only a small computational overhead during inference. To strengthen future-aware control under this first-frame-only interface, we introduce privileged-information distillation. The privileged teacher comprises a Video DiT with AIT that attends to visual tokens over the full clip, while the AIT in OSWAM attends only first-frame visual context. By aligning OSWAM's intermediate imagination token features with those of the teacher, we transfer this privileged information to improve action generation, while OSWAM remains first-frame-only at inference.

3.2 Imagination Tokens

Direct-policy WAMs eliminate future video generation at inference but still require multi-step denoising. To enable one-step action generation, we introduce two groups of imagination tokens.

Action and Scene Imagination Tokens. As mentioned in the last subsection, we augment the Video DiT input with two groups of noise-initialized latent tokens: AIT and SIT. We adapt the Video DiT using LoRA (Hu et al., 2021) to support these additional tokens. Their combined token count is much smaller than that of the first-frame visual input tokens, introducing only a small computational overhead during inference.

During training, video tokens follow the standard diffusion process with randomly sampled noise levels, whereas AIT are constructed by encoding an initial action noise sample $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$ and assigned the fixed timestep $t_{\max}$. Within the Video DiT, AIT attends to first-frame visual tokens, SIT, and other AIT, while receiving task conditioning through cross-attention. These interactions integrate visual evidence and scene context into AIT. An auxiliary decoder maps the resulting AIT representations to an action velocity $\hat{\mathbf{v}}^A$, yielding a structured action prior $\bar{\mathbf{a}} = \boldsymbol{\epsilon} - \hat{\mathbf{v}}^A$. We then detach this prior and renoise it with the same initial noise that was encoded into AIT:

$$\mathbf{a}_{\tau_r}^{\text{init}} = (1 - \sigma_r)\,\text{sg}(\bar{\mathbf{a}}) + \sigma_r \boldsymbol{\epsilon}, \qquad \sigma_r = \tau_r / t_{\max}, \tag{1}$$

where σ_r controls the noise level and sg prevents gradients from propagating through the structured action prior. Conditioned on first-frame visual features and task inputs, the Action DiT predicts a refinement velocity $\hat{\mathbf{v}}^{\text{act}}$ and performs the single denoising step:

$$\hat{\mathbf{a}} = \mathbf{a}_{\tau_r}^{\text{init}} - \sigma_r \hat{\mathbf{v}}^{\text{act}}. \tag{2}$$

Its training target is $(\mathbf{a}_{\tau_r}^{\text{init}} - \mathbf{a}_{\text{gt}})/\sigma_r$, enabling the Action DiT to refine the structured initialization into the target action in a single denoising step.

To complement the action prior encoded by AIT, SIT acts as latent context slots that aggregate scene context for AIT to attend to. They are initialized from Gaussian noise independent of the AIT, projected into the Video DiT input space through a separate encoder, and assigned the timestep $t_{\max}$. As shown in Fig. 4 (a), under the structured attention mask, SIT attends to first-frame visual tokens and other SIT, while AIT additionally attends to SIT to access the aggregated scene context. SIT is learned indirectly through the auxiliary loss and feature distillation on AIT, with gradients flowing through the attention connections from AIT to SIT.

Role-aware 3D RoPE. As shown in Fig. 4 (b), we introduce Role-aware 3D RoPE to reflect the distinct functions of video tokens, AIT, and SIT within a shared positional coordinate system. Video

tokens represent local visual content and retain their original spatiotemporal coordinates. AIT encodes action structure and is assigned spatial coordinates outside the video grid to distinguish them from visual tokens. SIT serves as scene context aggregators and is anchored at the grid center, providing a common spatial reference for gathering information across the scene:

$$\mathbf{p}^V_{t,h,w} = (t, h, w), \quad \mathbf{p}^A_n = (t_n, H_v, W_v), \quad \mathbf{p}^S_n = \Big(t_n, \tfrac{H_v-1}{2}, \tfrac{W_v-1}{2}\Big). \tag{3}$$

where H_v and W_v are the video token grid dimensions. For each action position n, the corresponding AIT and SIT share the temporal coordinate $t_n = ns_a/(s_v d_t)$. Here, s_a and s_v are the action and video sampling strides, and d_t is the VAE temporal downsampling factor. This converts action positions to the video latent timeline. Thus, AIT and SIT share temporal coordinates but use distinct spatial anchors, preserving temporal alignment while providing role-specific positional cues for action initialization and scene context aggregation, respectively.

3.3 Privileged-Information Distillation

Although imagination tokens provide a structured action prior, long-horizon context is only implicitly encoded in the first-frame representation. We therefore introduce privileged-information distillation to transfer representations learned from privileged future observations into OSWAM's AIT, strengthening their predictive capability without requiring future observations at inference.

We separately train a Video DiT teacher with auxiliary AIT, following the same training settings as OSWAM except for the token groups and attention masks. As shown in Fig. 4 (a), the teacher contains video tokens and AIT, with AIT attending to the full video clip. OSWAM additionally includes SIT, and its AIT attends to first-frame visual tokens, SIT, and other AIT, while SIT attends only to first-frame visual tokens and other SIT.

During distillation, we freeze the teacher and provide it with ground-truth video latents over the full future horizon. The first-frame latents remain clean, while all future-frame latents are corrupted with noise at a fixed diffusion timestep τ_v^T before the teacher forward pass. The teacher's AIT remains initialized from pure noise at $t_{\max}$ and acquires privileged information by attending to these future video tokens. OSWAM retains its standard training procedure, using a randomly sampled video timestep and the fixed AIT timestep $t_{\max}$. The teacher and OSWAM share the same video noise and action noise samples, respectively. We then align OSWAM's AIT features with the corresponding teacher features at a set of Video DiT blocks $\mathcal{J}$, providing multi-depth distillation supervision. SIT is supervised indirectly through their attention connections to AIT. At inference, the teacher is discarded, preserving OSWAM's first-frame-only interface without additional computation.

3.4 Training Objective

OSWAM jointly learns visual dynamics, structured action DiT start points, and one-step action denoising. The Video DiT is supervised by video flow matching, while its AIT representations receive complementary action and privileged feature supervision to learn the structured initialization consumed by the Action DiT:

$$\mathcal{L}_{\text{vid}} = \mathbb{E}\big[\|\hat{\mathbf{v}}^V - (\boldsymbol{\epsilon}^V - \mathbf{z}^+)\|_2^2\big], \tag{4}$$

where $\mathbf{z}^+$ denotes clean future video latents and $\boldsymbol{\epsilon}^V$ is the video noise. To learn an observation-conditioned action prior, the AIT decoder is supervised with the action velocity target at $t_{\max}$:

$$\mathcal{L}_{\text{AIT}} = \mathbb{E}\big[\|\hat{\mathbf{v}}^A - (\boldsymbol{\epsilon} - \mathbf{a}_{\text{gt}})\|_2^2\big]. \tag{5}$$

Privileged feature distillation further guides the prior by aligning the OSWAM's AIT representations with teacher across the selected blocks $\mathcal{J}$:

$$\mathcal{L}_{\text{dist}} = |\mathcal{J}|^{-1}\mathbb{E}\big[\textstyle\sum_{j\in\mathcal{J}} \|\mathbf{h}_j^{\text{stu}} - \text{sg}(\mathbf{h}_j^{\text{tea}})\|_2^2\big], \tag{6}$$

where $\mathbf{h}_j^{\text{stu}}$ and $\mathbf{h}_j^{\text{tea}}$ denote corresponding AIT features after block j. Together, $\mathcal{L}_{\text{AIT}}$ and $\mathcal{L}_{\text{dist}}$ support learning an observation-conditioned structured initialization for the Action DiT. Given $\mathbf{a}^{\text{init}}_{\tau_r}$ constructed from the AIT prediction, the Action DiT learns to predict the demonstrated action chunks in one denoising step:

$$\mathcal{L}_{\text{act}} = \mathbb{E}\big[\|\hat{\mathbf{v}}^{\text{act}} - \sigma_r^{-1}(\mathbf{a}^{\text{init}}_{\tau_r} - \mathbf{a}_{\text{gt}})\|_2^2\big]. \tag{7}$$

Table 1: Comparison with state-of-the-art planners on the NAVSIM `navtest`. N^v and N^a denote the number of video and action **denoising steps**. †SimWAM is reported without reinforcement learning. ‡DriveVA is reported without CARLA mix training. *Latency reproduced by us.

Method	N^v	N^a	NC↑	DAC↑	EP↑	TTC↑	C↑	PDMS↑	Latency↓
Human Agent	–	–	100.0	100.0	87.5	100.0	99.9	94.8	–
Traditional E2E and VLA-based planners									
UniAD (Hu et al., 2023)	–	–	97.8	91.9	78.8	92.9	100.0	83.4	–
TransFuser (Chitta et al., 2022)	–	–	97.7	92.8	79.2	92.8	100.0	84.0	–
DiffusionDrive (Liao et al., 2025)	–	–	98.2	96.2	82.2	94.7	100.0	88.1	–
WoTE (Li et al., 2025c)	–	–	98.5	96.8	81.9	94.9	99.9	88.3	–
UniWorldVLA (Liu et al., 2026b)	–	–	98.7	96.7	83.2	96.1	100.0	89.4	–
DriveDreamer-Policy (Zhou et al., 2026a)	–	–	98.4	97.1	83.5	95.1	100.0	89.2	–
AutoVLA (Zhou et al., 2026b)	–	–	98.4	95.6	81.9	98.0	99.9	89.1	–
DriveVLA-W0 (Li et al., 2026e)	–	–	98.7	99.1	83.3	95.3	99.3	90.2	–
Imagine-then-execute-based WAM methods									
Epona (Zhang et al., 2025)	–	100	97.9	95.1	80.4	93.8	99.9	86.2	1171
PWM (Zhao et al., 2026b)	–	–	98.6	95.9	81.8	95.4	100.0	88.1	581
DriveLaW (Xia et al., 2026)	1	5	99.0	97.1	81.3	96.7	100.0	89.1	1353
DriveVA‡ (Liu et al., 2026a)	2	2	99.0	97.3	83.2	98.4	100.0	90.5	593*
DriveWAM (Shi et al., 2026)	3	10	98.3	98.1	84.3	95.2	100.0	90.1	1032
Direct policy-based WAM planners									
SimWAM† (Zhao et al., 2026c)	–	10	98.7	98.0	83.9	95.9	100.0	90.3	410
Metis (Li et al., 2026b)	–	10	98.3	97.1	83.4	94.7	100.0	89.1	440*
OSWAM (Ours)	–	1	98.6	98.0	84.3	95.6	100.0	90.3	136

Scaling the action residual by σ_r^{-1} ensures that one Euler update with step size σ_r directly moves $\mathbf{a}_{\tau_r}^{\text{init}}$ to the ground-truth action $\mathbf{a}_{\text{gt}}$. The overall objective is:

$$\mathcal{L} = \mathcal{L}_{\text{vid}} + \mathcal{L}_{\text{AIT}} + \lambda_{\text{dist}}\mathcal{L}_{\text{dist}} + \mathcal{L}_{\text{act}}. \tag{8}$$

4 Experiments

4.1 Datasets and Evaluation Metrics

Autonomous Driving. For autonomous driving, we evaluate OSWAM on the NAVSIM (Dauner et al., 2024), which provides real-world driving data sampled at 2Hz, with eight cameras covering a 360-degree field of view and LiDAR observations. The dataset comprises 1192 scenarios for training and 136 scenarios for testing. NAVSIM introduces practical and reliable metrics that effectively bridge open-loop and closed-loop evaluation. Specifically, NAVSIM evaluates model performance using the Predictive Driver Model Score (PDMS), which is calculated based on five factors: No At-Fault Collision (NC), Drivable Area Compliance (DAC), Time-to-Collision (TTC), Comfort (Comf.), and Ego Progress (EP). We report PDMS, which is calculated as $\text{PDMS} = \text{NC} \times \text{DAC} \times (5 \times \text{EP} + 5 \times \text{TTC} + 2 \times \text{C})/12$.

Embodied Control. For embodied control, we evaluate OSWAM on RoboTwin 2.0 (Chen et al., 2025), which is a challenging bimanual manipulation benchmark featuring more than 50 tasks that require coordinated dual-arm control. We adopt the multi-task setting of Fast-WAM, using data from over 50 tasks. The training data consist of 2,500 demonstrations from clean scenes and 25,000 demonstrations collected with heavy scene randomization. Performance is evaluated by the average success rate over 100 trials per task under both clean and randomized conditions.

4.2 Implementation Details

Following Fast-WAM, we initialize OSWAM from Wan2.2-5B and adopt the Action DiT with a reduced hidden dimension of 1024. The imagination-token LoRA adapters use rank 8 and $\alpha = 4$. For NAVSIM, we follow (Zhao et al., 2026c) to use a single front camera at 384×672 resolution and predict eight waypoints over four seconds at 2Hz. We use the current frame and eight future frames in training. For RoboTwin, we concatenate the head and two wrist views into a 384×320 image and use an action horizon of 32, with nine video frames sampled at a temporal stride of 4. For temporal RoPE, we set $s_a = 1$ and $d_t = 4$ on both benchmarks, with $s_v = 1$ for NAVSIM and $s_v = 4$ for

Table 2: Success rates on RoboTwin 2.0 simulation (Clean and Randomized splits, 50 tasks). EPT denotes embodied pretraining.

Method	N^v	N^a	EPT	Clean	Rand.	Average	Latency (ms)
VLA-based methods							
π_0 (Black et al., 2024)	–	–	w/	65.92	58.40	62.20	–
$\pi_{0.5}$ (Intelligence et al., 2025)	–	–	w/	82.74	76.76	79.80	–
X-VLA (Zheng et al.)	–	–	w/	72.90	72.80	72.80	–
Imagine-then-execute-based WAM methods							
Motus (Bi et al., 2026)	–	–	w/	88.66	87.02	87.80	2367
Motus (Bi et al., 2026)	–	–	w/o	77.56	77.00	77.30	2367
LingBot-VA (Li et al., 2026d)	25	50	w/	92.90	91.50	92.20	7659
LingBot-VA (Li et al., 2026d)	25	50	w/o	80.60	-	80.60	7659
Flash-WAM (Akbari et al., 2026)	1	2	w/o	88.42	82.66	85.54	707
Flash-WAM (Akbari et al., 2026)	1	1	w/o	82.56	80.26	81.41	609
Efficient-WAM (Li et al., 2026a)	10	10	w/o	86.70	85.70	86.20	568
Efficient-WAM-RT (Li et al., 2026a)	2	10	w/o	83.10	82.00	82.55	314
Direct policy-based WAM methods							
Fast-WAM (Yuan et al., 2026)	–	10	w/o	91.88	91.78	91.83	397
Faster-WAM (Ma et al., 2026)	–	10	w/o	89.70	88.64	89.17	–
Light-WAM (Li et al., 2026f)	–	–	w/o	76.40	76.30	76.40	107
OSWAM (Ours)	–	1	w/o	90.04	89.96	90.00	142

RoboTwin. AIT and SIT therefore use $t_n = n/4$, $(n = 1, \ldots, 8)$ on NAVSIM and $t_n = n/16$, $(n = 1, \ldots, 32)$ on RoboTwin; video latents use temporal coordinates $\{0, 1, 2\}$ in training.

We train with AdamW using a learning rate of 10^{-4}, weight decay of 0.01, and cosine decay with 5% linear warmup. NAVSIM training runs for 20,000 steps with a global batch size of 256, while RoboTwin training runs for 29,335 steps with a global batch size of 512. For distillation, the frozen teacher uses a video timestep of 200, and feature alignment is applied after Video DiT blocks $[8, 16, 24, 30]$. We set $\lambda_{\text{dist}} = 10$. Distillation starts after 500 steps, with its weight increasing linearly over the next 500 steps. Across both benchmarks, we set the AIT diffusion timestep to $t_{\max} = 1000$ and the re-noising timestep for Action DiT initialization to $\tau_r = 500$ ($\sigma_r = 0.5$). During inference, we use a single Action DiT denoising step. Unless otherwise specified, all latency measurements are conducted on a single NVIDIA H20 GPU and Intel Xeon Max 9468 CPU.

4.3 Main Results

NAVSIM. Table 1 compares OSWAM with state-of-the-art planners on NAVSIM. OSWAM achieves a competitive PDMS of 90.3 with a latency of 136 ms. Compared with SimWAM, OSWAM matches its PDMS while reducing action denoising from ten steps to one, yielding a $3.01\times$ speedup. OSWAM also outperforms Metis by 1.2 PDMS with a $3.24\times$ speedup. Compared with DriveVA, OSWAM achieves a $4.36\times$ speedup with only a 0.2 lower PDMS. These results demonstrate the effectiveness of structured initialization in enabling efficient one-step action generation while maintaining competitive planning performance.

RoboTwin 2.0. Table 2 reports success rates across 50 tasks under Clean and Randomized settings. OSWAM achieves an average success rate of 90.00% with a latency of 142ms. Compared with Fast-WAM, OSWAM reduces action denoising from ten steps to one and achieves a $2.80\times$ speedup, with only 1.83% decrease in average success rate. Compared with the one-step Flash-WAM variant, OSWAM improves average success by 8.59% and achieves a $4.29\times$ speedup. While Light-WAM achieves lower latency of 107ms, OSWAM improves largely the average success at the cost of only 35ms additional latency. Overall, OSWAM is competitive with multi-step direct-policy WAMs while substantially accelerating inference on challenging embodied control tasks.

Inference Efficiency. Table 3 compares inference efficiency with cached language context under the same environment. On RoboTwin 2.0, OSWAM reduces action GPU time from 67.61 to 6.78ms compared with Fast-WAM, while imagination tokens increase visual GPU time by 5.48ms. This yields a $3.39\times$ overall speedup, reducing latency from 323.67 to 95.35ms with nearly unchanged peak memory. On NAVSIM, OSWAM similarly achieves a $3.32\times$ speedup over SimWAM, reducing overall latency to 99.70ms. With compiled inference using `torch.compile` and CUDA Graph

Table 3: **Inference efficiency on RoboTwin 2.0 and Navsim.** Latency is measured with cached language context on a single NVIDIA RTX 5090 GPU and Intel Xeon Platinum 8470Q.

Dataset	Model	VAE GPU Time	Visual GPU Time	Action GPU Time	Peak Mem.	Overall Latency	Compiled Latency
RoboTwin 2.0	LingBot-VA	32.65	1165.12	1205.38	19.12	3365.35	–
	Motus	11.14	567.37	Shared	17.95	1472.85	–
	Fast-WAM	5.27	18.45	67.61	12.70	323.67	73.64
	OSWAM	5.27	23.93	6.78	12.75	**95.35**	**40.81**
Navsim	SimWAM	9.83	22.80	72.78	12.87	330.8	90.53
	OSWAM	9.82	30.87	7.32	12.90	**99.70**	**52.64**

Table 4: Component ablations on NAVSIM and RoboTwin.

Method	NAVSIM			RoboTwin		
	Latency↓	PDMS↑	ADE↓	Latency↓	Clean↑	Rand.↑
Baseline	108.35	71.04	1.049	112.84	57.38	55.80
+ AIT	135.92	89.81	0.594	139.82	87.42	88.16
+ Distillation	135.92	89.96	0.589	139.82	89.42	89.62
+ SIT	136.09	**90.28**	**0.587**	142.67	**90.04**	**89.96**

Table 5: Ablation on λ_{dist}.

λ_{dist}	PDMS↑	ADE↓
Baseline	89.81	0.594
0.01	89.71	0.595
0.1	89.92	0.596
1	90.01	0.591
10	**90.28**	**0.587**

optimization, OSWAM achieves latencies of 40.81ms on RoboTwin 2.0 and 52.64ms on NAVSIM, supporting inference rates above 10Hz and facilitating deployment in latency-critical applications such as autonomous driving. These results show that the reduction in action denoising cost outweighs the additional computation introduced by imagination tokens.

4.4 Ablation Studies

Effectiveness of Each Component. To more clearly illustrate the effectiveness of the different components in OSWAM, we conduct experiments by adding each of them to a baseline, which is set to one-step Fast-WAM. As shown in Table 4, AIT substantially improves performance across both benchmarks, raising NAVSIM PDMS from 71.04 to 89.81, and increasing RoboTwin success rates from 57.38% to 87.42% under Clean settings and from 55.80% to 88.16% under Randomized settings. The gains demonstrate the importance of structured start points for one-step action generation. Besides, privileged-information distillation further improves PDMS to 89.96 and RoboTwin success rates to 89.42% and 89.62%, without increasing inference latency. Finally, incorporating SIT yields the best performance on both benchmarks, demonstrating the complementary benefit of aggregating scene context for learning effective action priors.

Distillation Weight. Table 5 studies the effect of the distillation weight λ_{dist} on NAVSIM. While a small weight of 0.01 slightly underperforms the baseline, increasing the distillation strength generally improves performance. The best result is obtained with $\lambda_{\text{dist}} = 10$, reaching a 90.28 PDMS and a 0.587 ADE. This suggests that sufficiently strong distillation supervision is important for transferring future context into AIT. We therefore use $\lambda_{\text{dist}} = 10$ as the default setting.

More ablation studies and qualitative visualizations are provided in the **Appendix A.3** and **A.4**.

5 Conclusion

In this paper, we presented OSWAM, an efficient world-action model for one-step action generation. To address the reliance of direct-policy WAMs on iterative action denoising, we introduced Action Imagination Tokens to learn a structured initialization, supported by Scene Imagination Tokens that aggregate complementary scene context. We further developed privileged-information distillation to transfer full future context into the Action Imagination Tokens. Together, these designs enable action generation with a single Video DiT pass and one Action DiT denoising step. Experiments on NAVSIM, and RoboTwin 2.0 demonstrated the competitive planning and manipulation performance of OSWAM and its substantial latency reductions over direct-policy baselines. OSWAM offers a practical approach to efficient world-action modeling for latency-critical control.

Limitations. Our study is focused on learning action DiT structured initialization. Although SIT and privileged-information distillation support this objective, their potential to directly enhance the Action DiT remains unexplored. Extending scene-context conditioning and privileged supervision to the action expert is a promising direction for future work.

REFERENCES

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Niket Agarwal, Arslan Ali, Jon Allen, Martin Antolini, Adeline Aubame, Alisson Azzolini, Junjie Bai, Maciej Bala, Yogesh Balaji, Josh Bapst, et al. Cosmos 3: Omnimodal world models for physical ai. *arXiv preprint arXiv:2606.02800*, 2026.

Arman Akbari, Ci Zhang, Arash Akbari, Lin Zhao, Yixiao Chen, Weiwei Chen, Xuan Zhang, Geng Yuan, and Yanzhi Wang. Flash-wam: Modality-aware distillation for world action models. *arXiv preprint arXiv:2606.05254*, 2026.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35101–35113, 2026.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10164–10183, 2024.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

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

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

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 Robotics and Automation Letters*, 11(1):226–233, 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.

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

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 *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17853–17862. IEEE, 2023.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

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 *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8306–8316. IEEE, 2023.

Jiajun Li, Tiecheng Guo, Yifan Ye, Rongyu Zhang, Xiaowei Chi, Qianpu Sun, Ying Li, Yunfan Lou, Yan Huang, Zhihe Lu, et al. Efficient-wam: A 1b-parameter world-action model with low-cost future imagination. *arXiv preprint arXiv:2606.10040*, 2026a.

Jingyu Li, Zhe Liu, Dongnan Hu, Junjie Wu, Zipei Ma, Wenxiao Wu, Chao Han, Zhihui Hao, Zhikang Liu, Kun Zhan, et al. Metis: A generalizable and efficient world-action model for autonomous driving and urban navigation. *arXiv preprint arXiv:2606.15869*, 2026b.

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*, 2026c.

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.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026d.

Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. In *International Conference on Learning Representations*, volume 2025, pp. 42942–42959, 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. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27137–27146. IEEE, 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. In *International Conference on Learning Representations*, volume 2026, pp. 7890–7911, 2026e.

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*, 2025d.

Ziang Li, Dongzhou Cheng, Yibin Wang, Shiyue Wang, Xiaoyang Xu, Lingxuan Weng, Juan Wang, and Jiaqi Wang. Light-wam: Efficient world action models with state-fusion action decoding. *arXiv preprint arXiv:2606.08242*, 2026f.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12037–12047. IEEE, 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*, 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.

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

Liheng Ma, Rui Heng Yang, Zhanguang Zhang, Mateo Clemente, Ziwen Hu, Tongtong Cao, and Yingxue Zhang. Faster-wam: Do world action models need deep action modules? *arXiv preprint arXiv:2608.02365*, 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.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, Yihang Chen, Jie Liu, Yansong Cheng, Yao Yao, Jiayi Zhu, Yihao Meng, Kecheng Zheng, Qingyan Bai, Jingye Chen, Zehong Shen, Yue Yu, Xing Zhu, Yujun Shen, and Hao Ouyang. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026.

Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8372–8381. IEEE, 2023.

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.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22442–22452. IEEE, 2025.

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.

Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, Kun Ma, et al. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. In *International Conference on Learning Representations*, volume 2026, pp. 157518–157556, 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. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11910–11918, 2026.

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

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27220–27230. IEEE, 2025.

Weiheng Zhao, Haoyi Jiang, Xin Shi, Liu Liu, Fan Huang, Zhizhong Su, Wei Sui, and Xinggang Wang. Faster-wam: Efficient inference-time future conditioning for robust world action models. *arXiv preprint arXiv:2608.04404*, 2026a.

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, 2026b.

Zongchuang Zhao, Xin Zhou, Tianyang Xu, Zhengyang Sun, Kaixuan Zhou, Honglin Li, Dingkang Liang, and Xiang Bai. Simwam: A simple world action model for end-to-end autonomous driving. *arXiv preprint arXiv:2608.07468*, 2026c.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-vla: Soft-prompted transformer as scalable cross-embodiment vision-language-action model, 2025. *URL https://arxiv. org/abs/2510.10274*.

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

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

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*, 2024c.

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*, 2026a.

Zewei Zhou, Tianhui Cai, Seth 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. *Advances in Neural Information Processing Systems*, 38: 27920–27956, 2026b.

A Appendix

We provide the following materials in this appendix:

- Section A.1: Preliminary for OSWAM.
- Section A.2: More experiment results.
- Section A.3: More ablation studies.
- Section A.4: Qualitative visualizations.
- Section A.5: Declaration of LLM assistance.

A.1 Preliminary

A world action model predicts an action chunk by introducing future visual observations as an intermediate variable. Let o be the current observation, l an optional command, $a_{1:H}$ an action chunk, and $v_{1:T}$ the future visual states. Inverse-dynamics WAMs (Li et al., 2026d) first model the distribution of future visual states conditioned on the current observation and command, and then infer actions from the generated future:

$$p(a_{1:H} \mid o, l) = \int p(v_{1:T} \mid o, l)\, p(a_{1:H} \mid o, l, v_{1:T})\, dv_{1:T}. \tag{9}$$

Joint-WAMs (Bi et al., 2026), in contrast, model future video and actions within a shared generative process, $p(a_{1:H}, v_{1:T} \mid o, l)$. Both paradigms tightly couple control prediction with iterative future-video generation, which introduces substantial computational overhead.

Table 6: Comparison with state-of-the-art planners on the NAVSIM-v2 `navtest`. N^v and N^a denote the number of video and action denoising steps, respectively.

Method	N^v	N^a	NC↑	DAC↑	DDC↑	TLC↑	EP↑	TTC↑	LK↑	HC↑	EC↑	EPDMS↑
Human Agent	–	–	100.0	100.0	99.8	100.0	87.4	100.0	100.0	98.1	90.1	90.3
Traditional E2E and VLA-based planners												
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
GTRS-Dense (Li et al., 2025d)	–	–	97.6	97.5	99.0	99.9	87.9	97.0	95.9	97.5	55.9	82.3
DriveSuprim (Yao et al., 2026)	–	–	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
ReCogDrive (Xiong et al., 2026)	–	–	98.3	95.2	99.5	99.8	87.1	97.5	96.6	98.3	86.5	83.6
SGDrive (Li et al., 2026c)	–	–	98.6	94.3	99.5	99.9	86.0	97.9	96.1	98.3	85.9	86.2
DriveVLA-W0 (Li et al., 2026e)	–	–	98.5	99.1	98.0	99.7	86.4	98.1	93.2	97.9	58.9	86.1
WAM-based planners												
Epona (Zhang et al., 2025)	–	100	97.1	95.7	99.3	99.7	88.6	96.3	97.0	98.0	67.8	85.1
DriveLaW (Xia et al., 2026)	1	5	98.7	96.9	99.6	99.8	87.5	98.3	97.6	98.4	77.4	88.6
Metis (Li et al., 2026b)	–	10	98.4	97.2	99.6	99.8	87.8	97.7	97.8	98.4	88.0	89.5
SimWAM (Zhao et al., 2026c)	–	10	98.6	98.0	99.7	99.9	87.5	98.4	97.9	98.3	84.4	90.2
OSWAM (Ours)	–	1	98.5	98.0	99.7	99.8	87.9	98.4	98.0	98.4	84.3	**90.3**

Direct-policy WAMs (Yuan et al., 2026; Zhao et al., 2026c) remove explicit test-time video generation by retaining video prediction only as a training objective. At inference, the video DiT produces a latent representation $z(o)$ in a single forward pass, and actions are predicted directly as:

$$p(a_{1:H} \mid o, l) = p(a_{1:H} \mid z(o)). \tag{10}$$

However, action generation is iterative due to flow matching. Given $a_t = (1-t)a + t\epsilon$ with velocity target $\epsilon - a$, inference integrates the learned velocity field from noise ($t_K = 1$) to data ($t_0 = 0$) as:

$$a_{t_{k-1}} = a_{t_k} + (t_{k-1} - t_k)\, v_\theta(a_{t_k}, t_k). \qquad k = K, \ldots, 1, \tag{11}$$

High-quality actions typically require multiple velocity evaluations, making K a remaining inference bottleneck.

Our proposed OSWAM aims to preserve this direct-policy interface while reducing action generation to a single velocity evaluation, $K = 1$. A one-step update from an arbitrary time $t \in (0, 1]$ is:

$$\hat{a} = a_t - t\, v_\theta(a_t, t). \tag{12}$$

When the start is pure noise, $t = 1$ and $\hat{a} = a_1 - v_\theta(a_1, 1)$. OSWAM instead starts from a structured intermediate state with $t < 1$, which makes this single update sufficient for applications that require latency-critical control, such as autonomous driving.

A.2 More experiment results

Navsim v2. Compared with Navsim v1, Navsim v2 introduces the Extended Predictive Driver Model Score (EPDMS). EPDMS incorporates Driving Direction Compliance (DDC), Traffic Light Compliance (TLC), Lane Keeping (LK), and Extended Comfort (EC), while extending the original comfort metric to History Comfort (HC). As shown in Table 6, OSWAM achieves the highest EPDMS of 90.3 among the compared planners with only one action denoising step. It slightly surpasses SimWAM 90.2 and outperforms Metis 89.5, both of which require ten action denoising steps. These results further demonstrate the effectiveness of OSWAM under more comprehensive evaluation criteria encompassing safety, traffic-rule compliance, and driving comfort.

nuScene. We also evaluate OSWAM on the nuScenes (Caesar et al., 2020) dataset for autonomous driving. nuScenes comprises 1,000 real-world driving scenes, with keyframes annotated at 2 Hz. The scene images are captured by 6 cameras covering 360 degree FOV horizontally. The dataset is split into 700 scenes for training, 150 for validation, and 150 for testing. Following previous works (Zhao et al., 2026b; Wang et al., 2025), L_2 Displacement Error (DE) and Collision Rate (CR) are adopted to comprehensively evaluate the planning performance.

For nuScenes, we use a single front camera at 384×672 resolution. Each training sample contains the current frame and eight future frames sampled at 2Hz. We use eight action steps to match the four-second video horizon, with supervision and evaluation limited to the first three seconds. Each action is represented as (x, y, θ) in the current LiDAR coordinate frame, with padded and

Table 7: **End-to-end motion planning performance on the nuScenes Caesar et al. (2020) dataset.**

Method	Auxiliary Supervision	L2 (m) ↓				Collision Rate (%) ↓			
		1s	2s	3s	Avg.	1s	2s	3s	Avg.
ST-P3 Hu et al. (2022)	Map & Box & Depth	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71
UniAD Hu et al. (2023)	Map & Box & Motion & Tracklets & Occ	0.48	0.96	1.65	1.03	0.05	0.17	0.71	0.31
OccNet Tong et al. (2023)	3D-Occ & Map & Box	1.29	2.13	2.99	2.14	0.21	0.59	1.37	0.72
OccWorld Zheng et al. (2024a)	3D-Occ	0.52	1.27	2.41	1.40	0.12	0.40	2.08	0.87
VAD-Tiny Jiang et al. (2023)	Map & Box & Motion	0.60	1.23	2.06	1.30	0.31	0.53	1.33	0.72
VAD-Base Jiang et al. (2023)	Map & Box & Motion	0.54	1.15	1.98	1.22	0.04	0.39	1.17	0.53
GenAD Zheng et al. (2024b)	Map & Box & Motion	0.36	0.83	1.55	0.91	0.06	0.23	1.00	0.43
Doe-1 Zheng et al. (2024c)	QA	0.50	1.18	2.11	1.26	0.04	0.37	1.19	0.53
Epona (Zhang et al., 2025)	None	0.61	1.17	1.98	1.25	0.01	0.22	0.85	0.36
DriveLaW (Xia et al., 2026)	None	0.44	1.10	1.91	1.15	0.15	0.10	0.48	0.24
PWM (Zhao et al., 2026b)	Text	0.20	0.38	**0.65**	**0.41**	0.01	0.02	0.09	0.04
OSWAM (Ours)	None	**0.19**	**0.36**	0.66	**0.41**	**0.00**	**0.01**	**0.07**	**0.03**

Table 8: Action initialization ablation.

τ_r	PDMS↑	ADE↓
Rectified-flow refinement	89.97	0.609
400	90.17	0.587
500	90.28	0.587
600	90.08	0.598
Uniform (400-600)	90.02	0.591

Table 9: Teacher video timestep ablation.

Teacher video timestep	PDMS↑	ADE↓
0	90.20	0.585
200	90.28	0.587
400	90.19	0.588
600	90.16	0.590
800	90.16	0.591
1000	90.20	0.591

unsupervised positions excluded from the action loss. For Role-aware 3D RoPE, we set $s_a = s_v = 1$ and $d_t = 4$, giving AIT and SIT temporal coordinates $t_n = n/4$ for $(n = 1, \ldots, 8)$. Video latents use temporal coordinates $\{0, 1, 2\}$ during training.

We train for 10,000 steps with a global batch size of 128 using AdamW, a learning rate of 10^{-4}, weight decay of 0.01, and cosine decay with 5% linear warmup. For distillation, the frozen teacher uses a video timestep of 200, and feature alignment is applied after Video DiT blocks $[8, 16, 24, 30]$. We set $\lambda_{\text{dist}} = 10$. Distillation starts after 500 steps, with its weight increased linearly over the next 500 steps. We set the AIT diffusion timestep to $t_{\max} = 1000$ and the re-noising timestep for Action DiT initialization to $\tau_r = 500$ ($\sigma_r = 0.5$).

Results. Table 7 reports open-loop planning performance on nuScenes. Without auxiliary supervision, OSWAM achieves an average 0.41 m L2 error, matching PWM, while reducing the average collision rate from 0.04% to 0.03%. It also achieves the lowest L2 errors at 1 and 2 seconds and the lowest collision rates across all evaluated horizons. These results demonstrate that OSWAM not only accelerates action generation but also delivers strong open-loop planning performance, matching the best average L2 error while achieving the lowest average collision rate.

One-Step Action Generation. Tables 10 and 11 compare OSWAM with direct policy baselines with a one-step denoising setting. On NAVSIM, OSWAM achieves a PDMS of 90.3, outperforming the reported one-step results of SimWAM, Metis, and DriveVA by 21.4, 3.3, and 77.1 points, respectively. On RoboTwin 2.0, OSWAM improves the average success rate from 56.59% to 90.00% over our one-step Fast-WAM baseline, with consistent gains under both Clean and Randomized settings. These results demonstrate that OSWAM enables effective one-step action generation across both autonomous driving and embodied control.

A.3 Ablation Studies

Action Initialization. To evaluate different strategies for utilizing the structured action prior, we compare rectified-flow refinement (Liu et al., 2022) with re-noising at different noise levels. As shown in Table 8, all re-noising variants outperform rectified-flow refinement, achieving higher PDMS and lower ADE. Increasing τ_r from 400 to 500 improves PDMS without affecting ADE, whereas a further increase to 600 slightly degrades both metrics. Uniformly sampling τ_r from 400 to 600 also underperforms the fixed setting of 500. These results suggest that a fixed intermediate noise level provides a favorable trade-off between preserving the structured prior and allowing sufficient action refinement; we therefore set $\tau_r = 500$ by default.

Table 10: One-step planning performance on NAVSIM.

Method	PDMS↑	NC↑	DAC↑
SimWAM (1-step)	68.9	97.4	91.3
Metis (1-step)	87.0	–	–
DriveVA (1-step)	13.2	61.8	36.9
OSWAM (Ours)	**90.3**	**98.6**	**98.0**

Table 11: One-step success rates on RoboTwin 2.0.

Method	Clean↑	Rand.↑	Average↑
Fast-WAM (1-step)	57.38	55.80	56.59
OSWAM (Ours)	**90.04**	**89.96**	**90.00**

Teacher Video Timestep. To investigate how the teacher noise level affects AIT distillation, we vary the teacher video timestep τ_v^T from 0 to 1000. As shown in Table 9, performance changes only slightly across the evaluated settings, with PDMS ranging from 90.16 to 90.28. Even at $\tau_v^T = 1000$, where future video inputs are pure noise, OSWAM achieves a PDMS of 90.20, matching the clean-future setting ($\tau_v^T = 0$). This suggests that the teacher's learned future-modeling representations can provide useful supervision for AIT even without access to clean ground-truth future frames. We use $\tau_v^T = 200$ by default, as it achieves the highest PDMS.

A.4 Qualitative visualizations

Figure 5 compares SimWAM and OSWAM under the one-step action generation setting. SimWAM struggles to produce smooth trajectories, particularly when turning or navigating complex interactions with surrounding agents, leading to drivable-area violations and at-fault collisions. In contrast, OSWAM handles these challenging examples with a single action denoising step, producing smoother trajectories while satisfying both no-at-fault collision and drivable-area compliance criteria. Together with the real-time inference speed reported in Table 3, these results highlight the potential of OSWAM to enable practical deployment of WAMs in autonomous driving.

A.5 Declaration of LLM assistance

During the preparation of this manuscript, we used GPT-6 solely to improve the linguistic quality of the text. After completing the initial draft, selected passages were provided to the model for suggestions on grammar, clarity, and conciseness. All AI-assisted revisions were subsequently reviewed and, where necessary, further edited by the authors. The authors take full responsibility for the accuracy, integrity, and scholarly appropriateness of the final manuscript.

Table 12: Per-task success rates (%) on RoboTwin 2.0 under clean and randomized settings.

Task	**OSWAM**		Fast-WAM		Light-WAM		LingBot-VA		Motus		$\pi_{0.5}$	
	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Adjust Bottle	100	99	100	100	100	100	90	94	89	93	100	99
Beat Block Hammer	99	92	99	97	83	80	96	98	95	88	96	93
Blocks Ranking RGB	99	97	100	100	96	91	99	98	99	97	92	85
Blocks Ranking Size	84	81	94	98	57	54	94	96	75	63	49	26
Click Alarmclock	100	99	100	100	100	100	99	100	100	100	98	89
Click Bell	97	99	100	100	100	100	100	100	100	100	99	66
Dump Bin Bigbin	92	95	97	96	81	75	89	96	95	91	92	97
Grab Roller	100	100	100	100	100	98	100	100	100	100	100	100
Handover Block	84	79	95	81	71	59	99	78	86	73	66	57
Handover Mic	96	100	99	100	90	94	94	96	78	63	98	97
Hanging Mug	47	68	58	62	25	17	40	28	38	38	18	17
Lift Pot	100	98	100	100	93	93	100	99	96	99	96	85
Move Can Pot	74	94	90	88	57	74	94	97	34	74	51	55
Move Pillbottle Pad	97	96	100	99	69	74	99	99	93	96	84	61
Move Playingcard Away	99	99	100	100	93	92	100	99	100	96	96	84
Move Stapler Pad	69	72	77	64	26	34	91	79	83	85	56	42
Open Laptop	99	98	98	100	91	97	92	94	95	91	90	96
Open Microwave	82	76	62	45	76	59	82	86	95	91	34	77
Pick Diverse Bottles	80	80	80	85	61	57	89	82	90	91	81	71

Continued on the next page

Table 12 (continued)

Task	**OSWAM**		Fast-WAM		Light-WAM		LingBot-VA		Motus		$\pi_{0.5}$	
	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Pick Dual Bottles	97	92	100	96	90	63	100	99	96	90	93	63
Place A2B Left	89	95	95	93	84	83	97	93	88	79	87	82
Place A2B Right	96	94	93	99	89	85	97	95	91	87	87	84
Place Bread Basket	96	96	91	93	81	80	97	95	91	94	77	64
Place Bread Skillet	92	85	90	93	92	82	95	90	86	83	85	66
Place Burger Fries	97	99	96	99	95	98	97	95	98	98	94	87
Place Can Basket	79	65	71	69	57	57	81	84	81	76	62	62
Place Cans Plasticbox	97	94	99	96	37	68	100	99	98	94	94	84
Place Container Plate	99	99	96	100	99	94	99	97	98	99	99	95
Place Dual Shoes	83	87	94	88	53	51	94	89	93	87	75	75
Place Empty Cup	98	100	100	100	91	95	100	100	99	98	100	99
Place Fan	95	89	96	96	77	74	99	93	91	87	87	85
Place Mouse Pad	83	83	83	89	58	62	93	96	66	68	60	39
Place Object Basket	86	78	89	88	81	72	91	88	81	87	80	76
Place Object Scale	88	90	90	97	72	74	96	95	88	85	86	80
Place Object Stand	91	94	90	94	78	85	99	96	98	97	91	85
Place Phone Stand	99	97	97	99	85	88	97	97	87	86	81	81
Place Shoe	93	96	96	99	84	87	98	98	99	97	92	93
Press Stapler	80	86	90	97	65	76	85	82	93	98	87	83
Put Bottles Dustbin	90	89	95	90	65	65	87	91	81	79	84	79
Put Object Cabinet	84	85	94	89	80	68	85	87	88	71	80	79
Rotate QRcode	83	84	93	89	72	84	96	91	89	73	89	87
Scan Object	85	81	89	92	60	52	96	91	67	66	72	65
Shake Bottle Horizontally	100	100	100	100	100	98	100	99	100	98	99	99
Shake Bottle	100	100	100	100	100	99	100	97	100	97	99	97
Stack Blocks Three	94	95	95	97	65	67	99	98	91	95	91	76
Stack Blocks Two	99	100	100	100	94	91	100	98	100	98	97	100
Stack Bowls Three	84	75	80	81	65	72	86	83	79	87	77	71
Stack Bowls Two	95	96	92	98	91	97	94	98	98	98	95	96
Stamp Seal	92	87	90	94	60	63	96	97	93	92	79	55
Turn Switch	60	65	61	59	33	39	44	45	84	78	62	54
Average	90.0	90.0	91.9	91.8	76.4	76.3	92.9	91.5	88.7	87.0	82.7	76.8

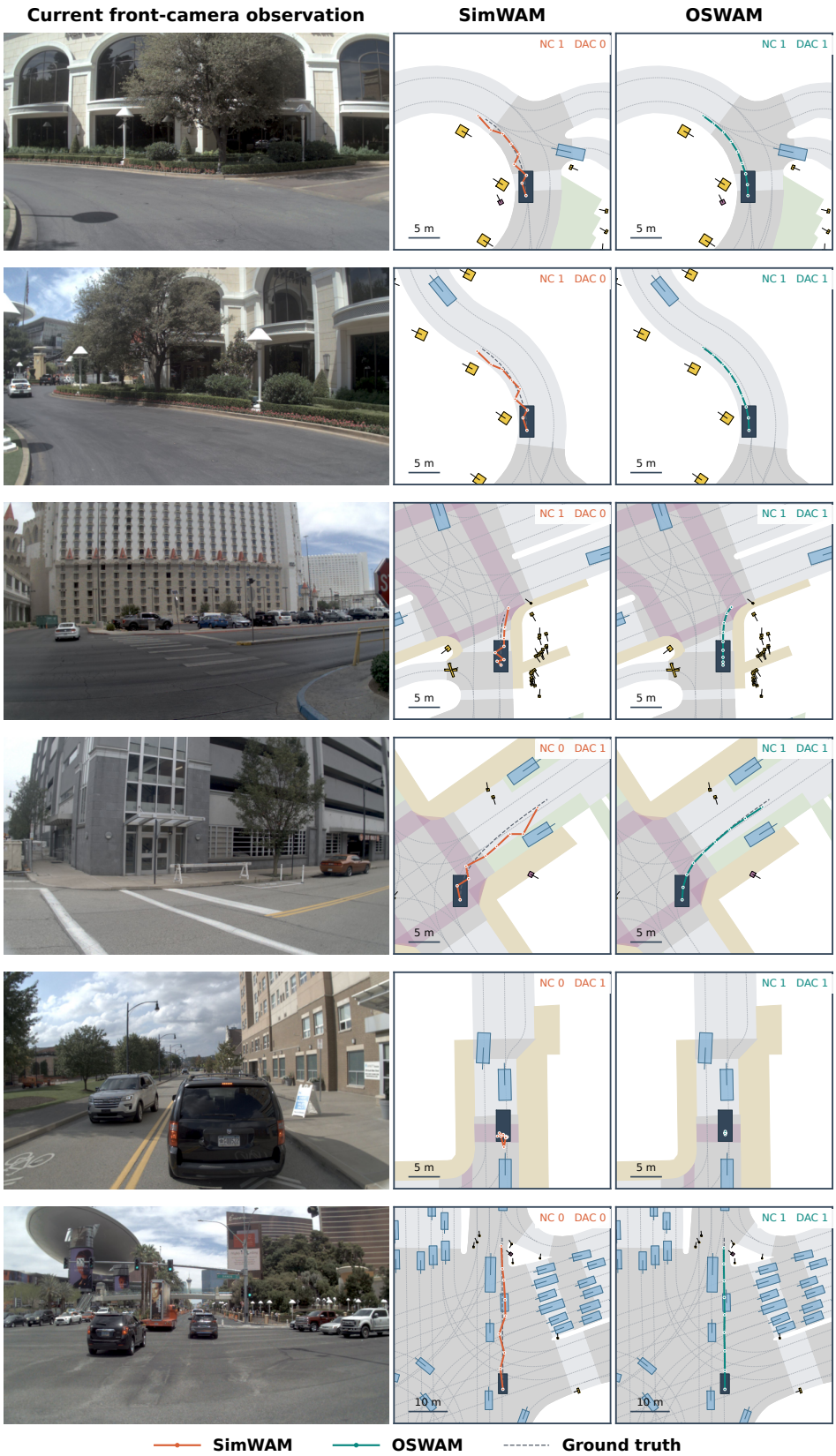

Figure 5: Qualitative comparison of **one-step** SimWAM and OSWAM on Navsim v1.