LEARNING TO DRIVE FROM THE FUTURE, PLANNING TRAJECTORIES IN ONE FORWARD PASS

Anonymous authors
Paper under double-blind review

ABSTRACT

World–action models (WAMs) enrich autonomous-driving policies with future-state supervision, yet it remains unclear which visual representations best support joint future-state and action learning and how to retain the benefits of future supervision in a single-pass planner. We introduce JEWAM, a Joint-Embedding World–Action Model for joint future-state and action learning, and use this framework to evaluate different visual representations. Comparisons of features from ViT, MAE, CroCo v2, DINOv3, and V-JEPA under a shared interface show that representation choice substantially affects planning. V-JEPA's predictive video features achieve the strongest planning results among the evaluated representations, motivating their adoption in the final model. We further show that future-state supervision improves action learning while its benefits are retained with single-pass, action-only inference. Direct full-horizon prediction outperforms the tested iterative variants at lower inference cost, supporting effective world–action learning without future rollout or iterative denoising at deployment. Using only front-view camera observations, JEWAM achieves 90.98 EPDMS on NAVSIM v2 NavTest and 39.86 on NavHard.

1 INTRODUCTION

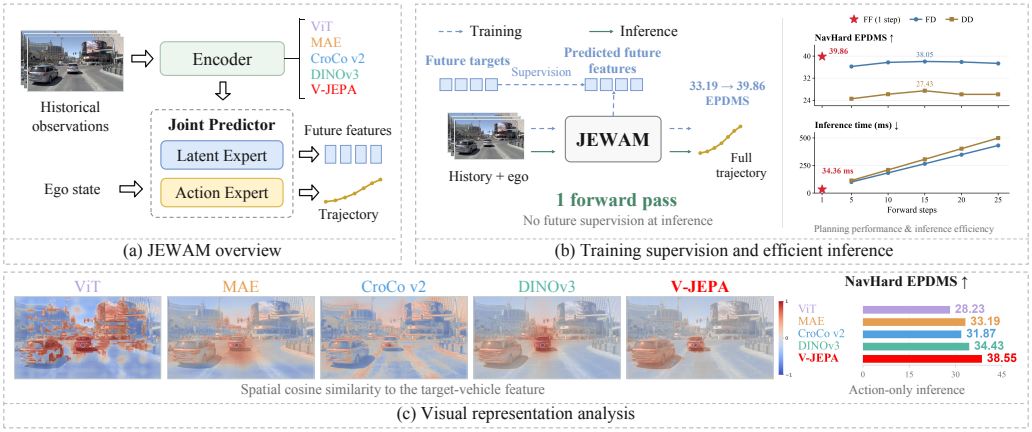

Figure 1: **Learning from future representations, planning in one pass. (a)** JEWAM compares five encoders within a shared world–action framework. **(b)** Future supervision improves planning, while deployment uses only the single-pass action path. **(c)** Feature visualizations and NavHard scores for five encoders.

Driving policies must select actions that account for how the surrounding scene evolves. World–action models (WAMs) support this goal by learning future world states alongside driving actions (Xia et al., 2026; Shi et al., 2026; Zhou et al., 2026; Wang et al., 2026a). Compared with direct observation-to-action learning (Chitta et al., 2023; Jiang et al., 2023; Sun et al., 2026), 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.

state prediction provides supervision about scene evolution beyond ego-trajectory labels. A central question is which visual representations best support joint future-state and action learning.

Video-generative WAMs provide dense supervision for motion and scene evolution by predicting future visual observations alongside actions (Zhang et al., 2025b; Xia et al., 2026; Shi et al., 2026; Liu et al., 2026b). However, a visual-synthesis objective does not directly establish which aspects of that supervision improve trajectory planning (Shi et al., 2026). When future video or actions are generated through iterative sampling at deployment, repeated model evaluations also increase inference cost (Zhang et al., 2025b; Xia et al., 2026; Shi et al., 2026; Liu et al., 2026b). Recent work in autonomous driving and robotic manipulation shows that future prediction can improve action learning even when future outputs are omitted at inference (Li et al., 2026a; Wang et al., 2026a; Yuan et al., 2026; Ye et al., 2026). Building on this separation of training and deployment, we study which representations support effective joint learning and whether the resulting gains can be retained when the planner predicts complete trajectories in one forward pass.

Latent-predictive WAMs construct future-state targets in representation space without reconstructing future images (Li et al., 2025a; Wang et al., 2026a; Hong et al., 2026). However, these methods typically evaluate future prediction within their chosen latent spaces, leaving the effect of representation choice on action learning less clearly understood. We therefore compare features from ViT, MAE, CroCo v2, DINOv3, and V-JEPA 2.1 within a shared world–action interface (Dosovitskiy et al., 2021; He et al., 2022; Weinzaepfel et al., 2023; Siméoni et al., 2025; Mur-Labadia et al., 2026). For each representation, the encoder supplies both historical context features and future-state targets, while the variants share the joint prediction setup and training budget. This comparison examines the planning utility of predictive video features alongside features learned through image-based pretraining.

We introduce JEWAM, a Joint-Embedding World–Action Model for joint future-state and action learning, and use it as a common framework to study these representation choices. During training, a unified predictor uses historical observations and the current ego state to predict future-state representations and full-horizon actions, with encoded future observations providing the representation targets. Guided by the representation comparison, the final JEWAM model adopts V-JEPA-based features. Following the asymmetric routing in Metis (Li et al., 2026a), the action path can operate without future-state tokens. At inference, JEWAM removes the teacher and future-state prediction branch and predicts the complete action horizon in one forward pass, without future rollout or iterative denoising. Figure 1 summarizes representation choice, future supervision, and action-only deployment.

The representation study reveals substantial differences in planning performance across features. With action-only inference, V-JEPA exceeds DINOv3, the strongest image-pretrained baseline in our comparison, by 4.12 NavHard EPDMS points. A separate training ablation shows that, under matched training settings and action-only inference, adding the future-state training branch improves NavHard EPDMS from 33.19 to 39.86. Feed-forward joint prediction also outperforms the evaluated iterative variants, with lower inference cost across the tested denoising budgets.

With the selected V-JEPA encoder, JEWAM achieves 90.98 EPDMS on NAVSIM v2 NavTest and 39.86 on NavHard (Cao et al., 2026; Dauner et al., 2024) using only front-view camera observations. These are the highest planning scores among the methods in our comparison tables for these two benchmarks.

Our contributions are summarized as follows:

- We introduce JEWAM, a joint-embedding world–action model that learns future-state representations and full-horizon actions with V-JEPA-based supervision and supports action-only inference in a single forward pass.
- We compare five visual representations under a shared world–action interface to evaluate how feature choice affects planning, complemented by ablations of future-state supervision and feed-forward versus iterative prediction.
- Using only front-view camera observations, JEWAM achieves the highest planning scores in our NAVSIM v2 NavTest and NavHard comparison tables, with a single-pass, action-only deployment path.

2 Related Work

2.1 Video-Generative World–Action Models

Video-generative driving WAMs couple future visual prediction with action learning through autoregressive video–action modeling (Chen et al., 2025; Bartoccioni et al., 2025), joint future-frame and action prediction (Zhao et al., 2025; Liu et al., 2026c; Shi et al., 2026; Liu et al., 2026b), or separate video and trajectory branches (Zhang et al., 2025b; Xia et al., 2026). Some also predict depth alongside future video (Zhou et al., 2026; Sheng et al., 2026). However, visual supervision need not imply visual rollout at deployment. DriveVLA-W0 uses a lightweight action expert, while Metis and UNIVERSE bypass future-video generation for action prediction (Li et al., 2026c;a; Liu et al., 2026a). Inference cost therefore depends on whether the planner requires visual rollout and whether action decoding remains iterative.

2.2 Latent-Predictive World–Action Models

Latent-predictive WAMs learn future scene representations without video reconstruction, using scene features, compact tokens, or BEV states as prediction targets (Li et al., 2025a; Zheng et al., 2025; Wang et al., 2026a; Yang et al., 2026; Li et al., 2025b; Zhang et al., 2025a). Their inference paths differ, with DriveFuture conditioning a diffusion planner on predicted future latents (Hong et al., 2026), whereas a future target can also supervise training without being consumed by the deployed planner. Beyond the inference path, the representation used for joint learning also matters. V-JEPA learns spatiotemporal video features through latent prediction, while DINOv3 learns image features (Bardes et al., 2024; Assran et al., 2025; Mur-Labadia et al., 2026; Siméoni et al., 2025). Drive-JEPA studies driving-video pretraining with V-JEPA and introduces a proposal-based planner with multimodal trajectory distillation (Wang et al., 2026b). JEWAM studies representation choice within joint future-state and action training: each compared encoder supplies both historical context and future targets under a shared world–action interface. The selected model uses V-JEPA features and predicts the complete trajectory in one forward pass. Its experiments separately assess representation choice, the benefit of future supervision, and feed-forward versus iterative prediction, while adopting asymmetric routing from Metis (Li et al., 2026a).

3 Method

3.1 Problem Formulation

At decision step n, the planner observes historical images $\mathbf{f}_{0:n}$ and the current ego state e_n, and predicts a trajectory $\mathbf{a}_{n+1:n+m}$ over m future steps. Each action $a_i = (x_i, y_i, \psi_i)$ is an ego-vehicle BEV pose in the coordinate frame at step n. During training, future observations $\mathbf{f}_{n+1:n+m}$ provide additional supervision in representation space; they are unavailable to the deployed planner.

JEWAM learns future-state representations and driving actions within a common world–action framework. Let $\mathbf{z}^{\mathrm{hist}}$ denote the encoded historical context and $\mathbf{z}^{\mathrm{fut}}$ the future representation targets. These symbols denote token sequences, whose temporal granularity depends on the encoder and need not match the action sequence. The joint predictor $\mathcal{M}_\Theta$ estimates

$$\left(\hat{\mathbf{z}}^{\mathrm{fut}}, \hat{\mathbf{a}}_{n+1:n+m}\right) = \mathcal{M}_\Theta\left(\mathbf{z}^{\mathrm{hist}}, e_n\right). \tag{1}$$

Equation 1 defines the joint prediction task. Future images supply target representations, not predictor inputs. At deployment, only the action path is retained:

$$\hat{\mathbf{a}}_{n+1:n+m} = \mathcal{M}^a_\Theta\left(\mathbf{z}^{\mathrm{hist}}, e_n\right). \tag{2}$$

The goal is to learn from future observations while predicting the complete trajectory in one forward pass, without future rollout or iterative action generation.

3.2 Joint World–Action Learning

Shared representation interface. JEWAM comprises a student encoder, a training-only teacher encoder, and a Mixture-of-Transformers (MoT) predictor with future-state and action experts. The

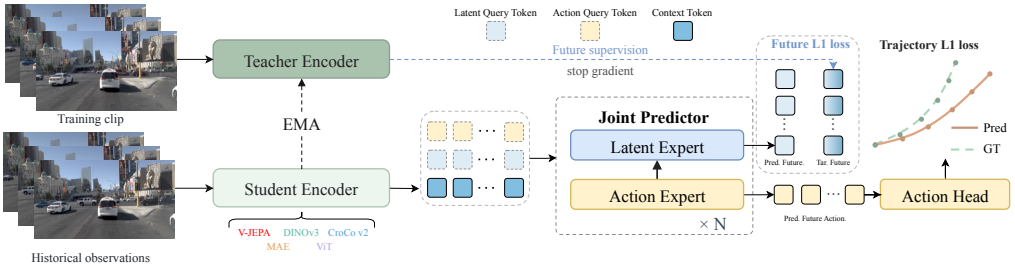

Figure 2: JEWAM's joint training framework. The student encoder supplies historical context, the EMA teacher constructs stopped-gradient future-state targets, and the MoT predictor learns future representations and full-horizon actions. At deployment, the teacher and future-state prediction branch are removed and the action path predicts the complete trajectory in one forward pass.

encoder choice determines both historical context features and future-state targets. We use this interface to compare ViT, MAE, CroCo v2, DINOv3, and V-JEPA features under a shared joint prediction setup and training budget. The comparison evaluates each representation configuration as a whole, rather than changing future targets while holding context features fixed. Guided by the results in Sec. 4.3, the final model uses V-JEPA 2.1 features (Assran et al., 2025; Mur-Labadia et al., 2026). Additional details of the representation comparison are given in Appendix A.2.

Context and supervision. The student encoder E_θ processes historical observations. The teacher $E_{\bar{\theta}}$ is initialized from the student and updated by an exponential moving average of its parameters. It encodes the complete training clip, from which we select the token positions $\mathcal{I}_{\text{fut}}$ corresponding to the future supervision interval:

$$\mathbf{z}^{\text{hist}} = E_\theta(\mathbf{f}_{0:n}), \qquad \mathbf{z}^{\text{fut}} = \text{sg}\big([E_{\bar{\theta}}(\mathbf{f}_{0:n+m})]_{\mathcal{I}_{\text{fut}}}\big). \tag{3}$$

The stop-gradient operator sg blocks gradients through the teacher. All compared encoders process temporal observations with causal attention. The student sees only historical images; the teacher's future representations are used only as training targets.

V-JEPA instantiation. The selected V-JEPA encoder uses spatiotemporal rotary positional encoding (RoPE) (Su et al., 2024) and causal attention over latent temporal positions. Each two-frame tubelet becomes one latent time step, so temporal causality is defined at this granularity. With four historical and eight future frames, the encoder represents two historical and four future tubelet positions, while the planner predicts eight future BEV poses. The MoT predictor augments projected context tokens with learnable future-state tokens and action queries. The latent stream uses 3D RoPE over time, height, and width. The action stream uses 2D RoPE over latent time and the action index within each tubelet, aligning its two poses per latent interval with the future-state tokens.

Joint predictor and supervision. Each MoT block uses expert-specific normalization, query-key-value projections, output projections, and MLP parameters, with shared attention over the merged token sequence. Following Metis (Li et al., 2026a), the attention mask permits future-state queries to read action keys and values, while blocking action queries from reading future-state keys and values. Both branches can access historical context and ego-state conditioning. This asymmetric routing is designed to transfer supervision through joint training while supporting an action path that can run without future-state tokens.

The future-state head predicts teacher representations, and the action head predicts normalized BEV poses. We jointly optimize the L1 future-state and action losses:

$$\mathcal{L}_l = \left\|\hat{\mathbf{z}}^{\text{fut}} - \mathbf{z}^{\text{fut}}\right\|_1, \qquad \mathcal{L}_a = \left\|\hat{\mathbf{a}}_{n+1:n+m} - \mathbf{a}_{n+1:n+m}\right\|_1, \qquad \mathcal{L}_{\text{JEWAM}} = \mathcal{L}_l + \mathcal{L}_a. \tag{4}$$

3.3 Single-Pass Action Inference

At inference, JEWAM encodes historical observations and predicts all future BEV poses together using the action queries and action head. The EMA teacher, future-state queries, and future-state

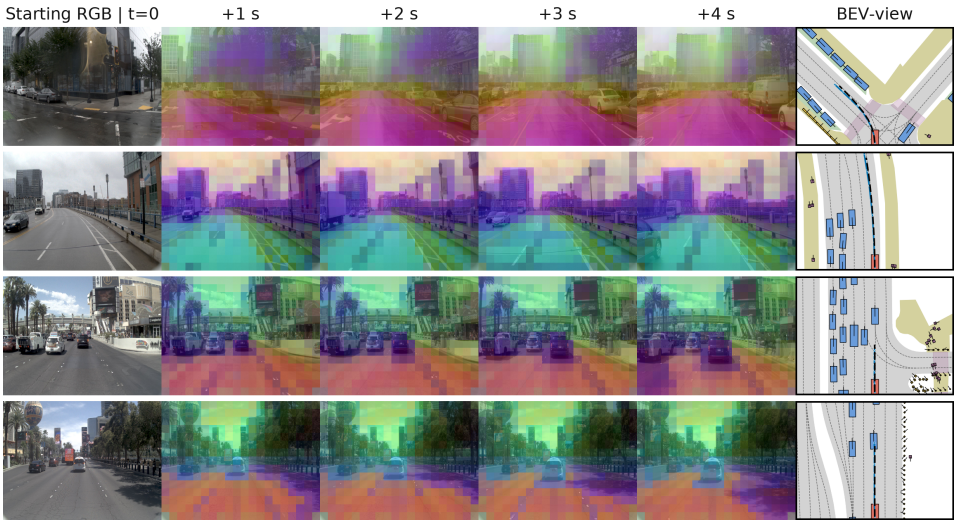

Figure 3: PCA analysis of predicted future-state representations and BEV visualization of ego trajectories.

head are omitted. One forward pass refers to predicting the complete trajectory from the observed context without autoregressive rollout or iterative denoising. The training-branch ablation measures the benefit of future supervision, and the same-checkpoint inference ablation tests whether planning performance is retained when future-state queries are removed (Sec. 4.3).

4 Experiments

4.1 Experimental Setup

Benchmark and Metrics. We evaluate JEWAM on NAVSIM (Dauner et al., 2024; Cao et al., 2026), which is built on OpenScene, a 2 Hz redistribution of the nuPlan dataset (Karnchanachari et al., 2024). We train on the official `navtrain` split and report EPDMS on NAVSIM v2 `navtest` and `navhard`. For NAVSIM v2, we use the corrected official evaluator with human-behavior penalty filtering, while `navhard` follows the two-stage pseudo-closed-loop evaluation protocol. All scores are reported as percentages, with higher values indicating better performance. Scores and their differences are rounded independently to two decimal places.

Implementation Details. Each training clip contains 12 front-view frames at 2 Hz with a spatial resolution of 240×320: the first four frames serve as planner input, and the remaining eight provide future supervision. The visual encoder is initialized from the V-JEPA 2.1 ViT-G checkpoint, and its exponential-moving-average teacher is updated with a momentum of 0.99925. We train JEWAM and its ablation variants for 20,000 optimization steps on 8 NVIDIA A800 GPUs with a global batch size of 128. We use AdamW with a weight decay of 0.02, set the learning rates of the encoder and predictor to 5×10^{-5} and 1×10^{-4}, respectively, and apply a cosine schedule with 2,000 warm-up steps. Unless otherwise specified, all variants use the same data, optimization settings, prediction horizon, and training budget. All reported inference times were measured on a single Alibaba Zhenwu PPU-ZW810E.

We compare JEWAM with traditional end-to-end, vision-language-action (VLA), and world-model-based planners. Results on NAVSIM v2 `navtest` and `navhard` evaluate overall planning quality and performance under the challenging two-stage protocol, respectively.

4.2 Main Results

On NAVSIM v2 `navtest`, JEWAM achieves 90.98 EPDMS, the highest score among the methods in Table 1, outperforming both video-generative and latent-predictive WAM baselines. It also

Table 1: Comparison on NAVSIM v2 `navtest`. * denotes reinforcement-learning training.

Method	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
Traditional E2E Planners										
TransFuser (Chitta et al., 2023)	96.90	89.90	97.80	99.70	87.10	95.40	92.70	98.30	87.20	76.70
DiffusionDrive (Liao et al., 2025)	98.20	95.90	99.40	99.80	87.50	97.30	96.80	98.30	87.70	84.50
VLA-based Planners										
ReCogDrive* (Li et al., 2026d)	98.30	95.20	99.50	99.80	87.10	97.50	96.60	98.30	86.50	83.60
SGDrive (Li et al., 2026b)	98.60	94.30	99.50	99.90	86.00	97.90	96.10	98.30	85.90	86.20
DriveFine* (Dang et al., 2026)	98.70	97.30	99.50	99.80	88.70	97.80	97.70	98.40	83.80	89.70
World-model / WAM-based Planners										
DriveVLA-W0 (Li et al., 2026c)	98.50	99.10	98.00	99.70	86.40	98.10	93.20	97.90	58.90	86.10
DriveLaW (Xia et al., 2026)	98.70	96.90	99.60	99.80	87.50	98.30	97.60	98.40	77.40	88.80
Latent-WAM (Wang et al., 2026a)	98.10	97.30	99.60	99.80	87.70	97.30	97.60	98.10	87.30	89.30
DriveWAM (Shi et al., 2026)	98.30	98.10	99.70	99.80	88.00	97.60	98.10	98.40	88.90	90.50
JEWAM (Ours)	99.20	98.10	99.70	99.90	87.10	98.80	97.90	98.30	83.50	**90.98**

Table 2: Comparison on NAVSIM v2 `navhard`.

Method	Stage	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	S. ↑	EPDMS ↑
Traditional E2E Planners												
TransFuser (Chitta et al., 2023)	S1	96.20	79.50	99.10	99.50	84.10	95.10	94.20	97.50	79.10	–	23.10
	S2	77.70	70.20	84.20	98.00	85.10	75.60	45.40	95.70	75.90	–	
DiffusionDrive (Liao et al., 2025)	S1	96.80	86.00	98.80	99.30	84.00	95.80	96.70	97.60	79.60	–	27.50
	S2	80.10	72.80	84.40	98.40	85.90	76.60	46.40	96.30	72.80	–	
VLA-based Planners												
SGDrive (Li et al., 2026b)	S1	95.80	87.60	97.80	99.80	84.40	94.70	92.90	97.80	28.90	–	25.50
	S2	79.40	65.40	79.10	98.90	88.90	75.30	42.70	96.40	29.60	–	
ReCogDrive (Li et al., 2026d)	S1	96.40	78.90	98.70	99.80	82.60	95.60	94.40	97.60	74.20	–	25.70
	S2	80.20	65.00	82.40	98.70	85.20	76.90	43.80	96.60	71.80	–	
DriveFine (Dang et al., 2026)	S1	97.60	90.00	99.10	99.30	84.90	96.70	97.30	97.60	72.00	–	30.50
	S2	82.10	71.30	84.80	98.40	88.10	74.30	47.20	96.80	72.80	–	
World-model / WAM-based Planners												
DriveVLA-W0 (Li et al., 2026c)	S1	96.80	83.30	99.00	99.60	84.60	95.30	96.40	97.60	78.20	–	24.40
	S2	76.80	64.30	79.90	98.30	89.20	75.00	46.80	95.80	53.10	–	
DriveLaW (Xia et al., 2026)	S1	97.30	89.10	99.20	99.60	84.30	97.10	96.20	97.80	67.60	–	30.60
	S2	82.50	67.60	83.50	98.10	84.80	78.50	45.80	96.40	57.30	–	
DriveWAM (Shi et al., 2026)	S1	97.20	92.40	99.70	99.30	84.60	95.80	98.20	97.80	81.30	81.40	36.10
	S2	82.20	76.40	85.90	98.30	87.00	77.50	50.40	95.20	72.80	45.20	
JEWAM (Ours)	S1	97.60	94.90	99.60	99.30	83.80	97.10	98.40	97.80	77.80	83.30	**39.86**
	S2	85.50	77.20	87.70	98.30	83.60	82.80	48.00	96.30	60.50	46.90	

surpasses the strongest listed VLA planner, DriveFine, which achieves 89.70 EPDMS (Dang et al., 2026). JEWAM obtains these results using action-only inference.

NAVSIM v2 `navhard` scores both the initial scene and precomputed follow-up scenes near the first-stage endpoint, testing planning under shifted observations through a two-stage pseudo-closed-loop protocol (Cao et al., 2026). JEWAM attains the highest final EPDMS among the methods in Table 2, reaching **39.86** versus 36.10 for our full-split DriveWAM evaluation (Shi et al., 2026). The corresponding stage scores are 83.30 versus 81.40 in Stage 1 and 46.90 versus 45.20 in Stage 2. JEWAM also exceeds the strongest listed traditional and VLA baselines, DiffusionDrive (27.50) and DriveFine (30.50), on final EPDMS (Liao et al., 2025; Dang et al., 2026).

The second-stage breakdown makes the comparison with DriveWAM more informative: JEWAM scores higher on no-at-fault collisions (85.50 versus 82.20), time to collision (82.80 versus 77.50), and driving-direction compliance (87.70 versus 85.90). Its ego progress (83.60 versus 87.00), lane keeping (48.00 versus 50.40), and extended comfort (60.50 versus 72.80) are lower, so the advantage does not extend to every component. Together, the final and stage-wise results show stronger aggregate planning performance for the complete JEWAM system on the evaluated `navhard` protocol, with particularly clear gains in these second-stage safety-related measures.

Beyond the NAVSIM v2 evaluations, we also evaluate JEWAM on NAVSIM v1 `navtest`, where it achieves 92.08 PDMS, the highest score in our comparison table. Detailed results and comparisons are provided in Appendix A.1.

Figure 3 provides a qualitative view of predicted future-state features and ego trajectories in four selected scenes. The feature maps show coherent road regions and distinct regions around nearby vehicles, while the predicted trajectories remain close to the recorded paths. The future-state branch is enabled for this visualization; JEWAM uses action-only inference for planning evaluation unless otherwise specified.

4.3 Ablation Studies

We first compare visual representations within the shared world–action framework. We then test whether future-state supervision improves action learning and whether its benefits are retained with action-only inference. Finally, we compare single-pass and iterative predictor formulations. We report NAVSIM v2 `navhard` throughout and include `navtest` results where available.

Representation Choice. To examine how the representation used for both historical context and future supervision affects planning, we compare ViT (Dosovitskiy et al., 2021), MAE (He et al., 2022), CroCo v2 (Weinzaepfel et al., 2023), DINOv3 (Siméoni et al., 2025), and V-JEPA 2.1 (Mur-Labadia et al., 2026), using the 300M-parameter ViT-L variant of V-JEPA 2.1 to keep encoder parameter counts comparable. All encoders use causal temporal attention, and all predictors use future-state supervision and asymmetric routing. The shared interface preserves each encoder's temporal tokenization: V-JEPA uses two-frame tubelets, whereas the other encoders use one-frame tokens (Appendix A.2). Future On/Off evaluates the same checkpoint with or without future-state queries at inference.

Table 3: Latent representation and future-state inference results

	Future On		Future Off	
Encoder	NavTest	NavHard	NavTest	NavHard
ViT	85.49	28.08	85.44	28.23
MAE ViT-L	87.69	33.15	87.68	33.19
CroCo v2 ViT-L	86.58	31.67	86.59	31.87
DINOv3 ViT-L	88.48	34.04	88.51	34.43
V-JEPA-300M ViT-L	**89.44**	**38.51**	**89.50**	**38.55**

V-JEPA-300M achieves the strongest planning results among the compared representations (Table 3). With action-only inference, it reaches 89.50 on `navtest` and 38.55 on `navhard`, exceeding DINOv3, the strongest image-pretrained baseline, by 0.99 and 4.12 points, respectively. Removing future-state queries changes scores by at most 0.06 on `navtest` and 0.39 on `navhard`, preserving the representation ranking. These results motivate selecting V-JEPA for the final model.

Figures 4 and 1(c) provide complementary qualitative views of the encoder features. In the selected scenes, V-JEPA forms coherent road regions and distinct regions near vehicles. These visualizations illustrate feature organization; the planning results provide the basis for representation selection. Appendix A.3 additionally compares predicted future-state features over the same scene and prediction horizon.

Future Supervision and Action-Only Inference. We test whether future-state supervision improves action learning by comparing joint future-state and action training with action-only training under the same data, backbone, optimization, and training budget. Both variants use action-only inference. Joint training improves `navhard` EPDMS from 33.19 to 39.86, a gain of 6.67 points (Table 4). Thus, the future-state training branch provides useful supervision beyond trajectory labels. This ablation removes future tokens, teacher targets, and the future-state loss together, so the gain belongs to the complete branch rather than the loss alone.

Table 4: Future-state supervision.

Training Variant	EPDMS ↑
w/o Future	33.19
w/ Future	**39.86**

Table 5: Effect of future-state prediction at inference.

Inference Path	NavTest ↑	NavHard ↑
Action + future state	90.89	39.78
Action only	**90.98**	**39.86**

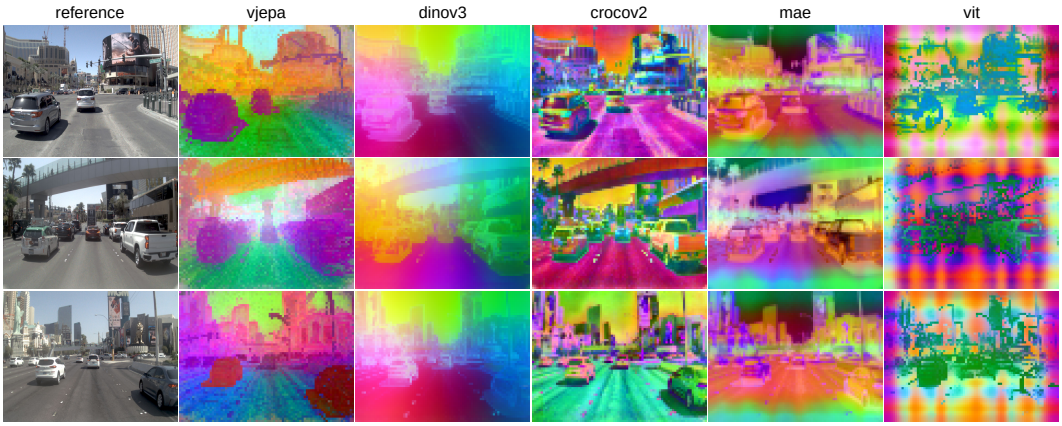

Figure 4: PCA analysis results of features extracted by different encoders across three driving scenes.

We also check whether the jointly trained action predictor needs future-state queries at inference. Removing them from the same checkpoint changes `navtest` and `navhard` EPDMS by only $+0.09$ and $+0.09$, respectively (Table 5). This comparison supports retention of planning performance with action-only inference, without implying numerically identical trajectories.

Single-Pass Prediction and Efficiency. We next examine the predictor formulation to assess whether iterative predictors provide planning gains that justify their inference cost. JEWAM performs joint future-state and action prediction using a feed-forward predictor, avoiding the iterative denoising commonly required by generative formulations. Under the same world–action interface and asymmetric attention mask described in Sec. 3.2, we compare the feed-forward FF model with two DiT-based variants. FD replaces only the action predictor, while DD replaces both the future-state and action predictors. Detailed architectures, objectives, and training and inference procedures are provided in Appendix A.4.

Table 6: Feed-forward and generative predictor variants. DiT-based variants use 15 denoising steps.

Variant	Future-State Predictor	Action Predictor	NavTest	NavHard
FF	FF	FF	**90.98**	**39.86**
FD	FF	DiT	90.42	38.05
DD	DiT	DiT	84.56	27.43

Replacing only the action predictor with DiT reduces `navtest`/`navhard` from 90.98/39.86 to 90.42/38.05; replacing both predictors further reduces the scores to 84.56/27.43 (Table 6). These results support feed-forward prediction under the tested implementations and training budget. Because the variants also change the training objective, they do not isolate iterative sampling as the cause. JEWAM achieves the highest planning scores and shortest inference time among these variants.

Table 7: Denoising steps on NavHard.

Denoising Steps	FD	DD
5	36.24	24.53
10	37.71	26.18
15	**38.05**	**27.43**
20	37.87	26.13
25	37.35	26.13
FF (one pass)	**39.86**	

The remaining question is whether the DiT variants underperform simply because their sampling budget is insufficient. Increasing the budget from 5 to 15 steps improves FD from 36.24 to 38.05 and DD from 24.53 to 27.43, whereas both variants decline at 20 and 25 steps (Table 7). Even at their best 15-step setting, FD and DD remain 1.81 and 12.43 points below the one-pass FF score of 39.86, respectively.

We further evaluate how planning performance and update frequency vary with the number of denoising steps (Figure 5). FD and DD achieve their best NavHard EPDMS at 15 steps, while further increasing the step count reduces update frequency without improving planning performance. The

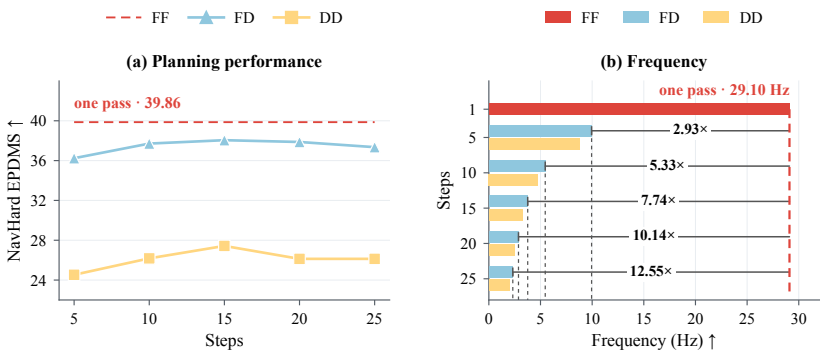

Figure 5: NavHard planning quality versus update frequency.

one-pass FF model achieves the highest NavHard EPDMS among the evaluated variants while operating at 29.10 Hz, yielding a 7.74× speedup over the faster baseline at 15 steps.

5 Conclusion

We presented JEWAM, a joint-embedding world–action framework for learning from future representations and planning trajectories in one forward pass. Comparing five visual representations within this framework shows that feature choice substantially affects planning, with V-JEPA achieving the strongest results among the evaluated configurations and motivating its adoption in the final model. Controlled ablations show that future-state supervision improves action learning and that the resulting planning performance is retained with action-only inference. Direct full-horizon prediction also outperforms the tested iterative variants under the evaluated training settings. Together, these findings support learning from future observations during training while deploying a single-pass trajectory planner. Our evaluation uses NAVSIM's benchmark protocols; future work will examine how these findings transfer to interactive closed-loop driving.

AI Use Statement

We used an AI assistant primarily to polish the manuscript and, for selected sections, to draft English prose from section and subsection outlines written by the authors. It also assisted with literature lookup, occasional feedback on experimental analysis, and a small number of auxiliary testing and experimental scripts. The authors reviewed and revised the AI-assisted text, conducted the experiments, and checked the code and results before including them in the paper. The authors take full responsibility for the final content.

Reproducibility Statement

The Experimental Setup subsection describes the NAVSIM data splits and evaluation protocols, the organization of historical and future frames, and the model initialization and training settings. The method and appendix provide further details of the model and experimental variants. Organizing the data, training the models, and evaluating them according to these specifications allows the reported experiments and results to be reproduced.

References

Mahmoud Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Mojtaba Komeili, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, Sergio Arnaud, Abha Gejji, Ada Martin, Francois Robert Hogan, Daniel Dugas, Piotr Bojanowski, Vasil Khalidov, Patrick Labatut, Francisco Massa, Marc Szafraniec, Kapil Krishnakumar, Yong Li, Xiaodong Ma, Sarath Chandar, Franziska Meier, Yann LeCun, Michael Rabbat, and Nicolas

Ballas. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025. doi: 10.48550/arXiv.2506.09985. URL https://arxiv.org/abs/2506.09985.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mahmoud Assran, and Nicolas Ballas. Revisiting feature prediction for learning visual representations from video, 2024. URL https://arxiv.org/abs/2404.08471.

Florent Bartoccioni, Elias Ramzi, Victor Besnier, Shashanka Venkataramanan, Tuan-Hung Vu, Yihong Xu, Loick Chambon, Spyros Gidaris, Serkan Odabas, David Hurych, Renaud Marlet, Alexandre Boulch, Mickael Chen, Éloi Zablocki, Andrei Bursuc, Eduardo Valle, and Matthieu Cord. Vavim and vavam: Autonomous driving through video generative modeling, 2025. URL https://arxiv.org/abs/2502.15672.

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, 2026. URL https://arxiv.org/abs/2506.04218.

Yuntao Chen, Yuqi Wang, and Zhaoxiang Zhang. Drivinggpt: Unifying driving world modeling and planning with multi-modal autoregressive transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26890–26900, 2025. URL https://arxiv.org/abs/2412.18607.

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, 2023. URL https://arxiv.org/abs/2205.15997.

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. URL https://arxiv.org/abs/2602.14577.

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*, 2024. URL https://arxiv.org/abs/2406.15349.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.

Yufeng Hong, Xiaotian Zhou, Yingyan Li, Xiangpo Zhou, Lin Liu, Yadan Luo, Shaoqing Xu, Lei Yang, and Ziying Song. Drivefuture: Future-aware latent world models for autonomous driving. *arXiv preprint arXiv:2605.09701*, 2026. URL https://arxiv.org/abs/2605.09701.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17853–17862, 2023. URL https://arxiv.org/abs/2212.10156.

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 *IEEE/CVF International Conference on Computer Vision*, pp. 8340–8350, 2023. URL https://arxiv.org/abs/2303.12077.

Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, and Holger Caesar. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. In *IEEE International Conference on Robotics and Automation*, 2024. URL https://arxiv.org/abs/2403.04133.

Jingyu Li, Zhe Liu, Dongnan Hu, Junjie Wu, Zipei Ma, Wenxiao Wu, Chao Han, Zhihui Hao, Zhikang Liu, Kun Zhan, Jiankang Deng, Xiatian Zhu, and Li Zhang. Metis: A generalizable and efficient world-action model for autonomous driving and urban navigation. *arXiv preprint arXiv:2606.15869*, 2026a. URL https://arxiv.org/abs/2606.15869.

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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4032–4042, 2026b. URL https://openaccess.thecvf.com/content/CVPR2026/html/Li_SGDrive_Scene-to-Goal_Hierarchical_World_Cognition_for_Autonomous_Driving_CVPR_2026_paper.html.

Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. *arXiv preprint arXiv:2406.08481*, 2025a. doi: 10.48550/arXiv.2406.08481. URL https://arxiv.org/abs/2406.08481.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025b. URL https://openaccess.thecvf.com/content/ICCV2025/html/Li_End-to-End_Driving_with_Online_Trajectory_Evaluation_via_BEV_World_Model_ICCV_2025_paper.html.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, Lu Hou, Lue Fan, and Zhaoxiang Zhang. Drivevla-w0: World models amplify data scaling law in autonomous driving. In *International Conference on Learning Representations*, 2026c. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0d70423f59c5fdd24f0dd3fa52e34623-Abstract-Conference.html.

Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. In *International Conference on Learning Representations*, 2026d. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/ff7bf6014f7826da531aa50f4538ee19-Abstract-Conference.html.

Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, and Xinggang Wang. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *arXiv preprint arXiv:2411.15139*, 2025. doi: 10.48550/arXiv.2411.15139. URL https://arxiv.org/abs/2411.15139.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.02747.

Mengmeng Liu, Diankun Zhang, Jiuming Liu, Jianfeng Cui, Hongwei Xie, Guang Chen, Hangjun Ye, Francesco Nex, Hao Cheng, and Michael Ying Yang. Universe: Unified video action models for autonomous driving with flexible mask-modulated modality generation, 2026a. URL https://arxiv.org/abs/2607.05133.

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, 2026b. URL https://arxiv.org/abs/2604.04198.

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*, 2026c. doi: 10.48550/arXiv.2603.27287. URL https://arxiv.org/abs/2603.27287.

Lorenzo Mur-Labadia, Matthew Muckley, Amir Bar, Mido Assran, Koustuv Sinha, Mike Rabbat, Yann LeCun, Nicolas Ballas, and Adrien Bardes. V-jepa 2.1: Unlocking dense features in video self-supervised learning. *arXiv preprint arXiv:2603.14482*, 2026. doi: 10.48550/arXiv.2603.14482. URL https://arxiv.org/abs/2603.14482.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2212.09748.

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

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

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025. URL https://arxiv.org/abs/2508.10104.

Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024. doi: 10.1016/j.neucom.2023.127063. URL https://arxiv.org/abs/2104.09864.

Wenchao Sun, Xuewu Lin, Keyu Chen, Zixiang Pei, Xiang Li, Yining Shi, and Sifa Zheng. Sparsedrivev2: Scoring is all you need for end-to-end autonomous driving. *arXiv preprint arXiv:2603.29163*, 2026. URL https://arxiv.org/abs/2603.29163.

Linbo Wang, Yupeng Zheng, Qiang Chen, Shiwei Li, Yichen Zhang, Zebin Xing, Qichao Zhang, Xiang Li, Deheng Qian, Pengxuan Yang, Yihang Dong, Ce Hao, Xiaoqing Ye, Junyu han, Yifeng Pan, and Dongbin Zhao. Latent-wam: Latent world action modeling for end-to-end autonomous driving. *arXiv preprint arXiv:2603.24581*, 2026a. doi: 10.48550/arXiv.2603.24581. URL https://arxiv.org/abs/2603.24581.

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:2601.22032v1*, 2026b. doi: 10.48550/arXiv.2601.22032. URL https://arxiv.org/abs/2601.22032v1.

Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jerome Revaud. CroCo v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17969–17980, 2023.

Tianze Xia, Yongkang Li, Lijun Zhou, Jingfeng Yao, Kaixin Xiong, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. Drivelaw: Unifying planning and video generation in a latent driving world. *arXiv preprint arXiv:2512.23421*, 2026. URL https://arxiv.org/abs/2512.23421.

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 *Proceedings of the AAAI Conference on Artificial Intelligence*, 2026. URL https://ojs.aaai.org/index.php/AAAI/article/view/38149.

Angen Ye, Boyuan Wang, Chaojun Ni, Guan Huang, Guosheng Zhao, Hao Li, Hengtao Li, Jie Li, Jindi Lv, Jingyu Liu, Min Cao, Peng Li, Qiuping Deng, Wenjun Mei, Xiaofeng Wang, Xinze Chen, Xinyu Zhou, Yang Wang, Yifan Chang, Yifan Li, Yukun Zhou, Yun Ye, Zhichao Liu, and Zheng Zhu. GigaWorld-Policy: An efficient action-centered world–action model, 2026. URL https://arxiv.org/abs/2603.17240.

Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination?, 2026. URL https://arxiv.org/abs/2603.16666.

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 *Advances in Neural Information Processing Systems*, 2025a. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/0ee633a6ade45eab4276352b3ee79c7a-Abstract-Conference.html.

Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, Xun Cao, and Wei Yin. Epona: Autoregressive diffusion world model for autonomous driving. *arXiv preprint arXiv:2506.24113*, 2025b. doi: 10.48550/arXiv.2506.24113. URL https://arxiv.org/abs/2506.24113.

Zhida Zhao, Talas Fu, Yifan Wang, Lijun Wang, and Huchuan Lu. From forecasting to planning: Policy world model for collaborative state-action prediction. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2510.19654.

Yupeng Zheng, Pengxuan Yang, Zebin Xing, Qichao Zhang, Yuhang Zheng, Yinfeng Gao, Pengfei Li, Teng Zhang, Zhongpu Xia, Peng Jia, and Dongbin Zhao. World4drive: End-to-end autonomous driving via intention-aware physical latent world model, 2025. URL https://arxiv.org/abs/2507.00603.

Yang Zhou, Xiaofeng Wang, Hao Shao, Letian Wang, Guosheng Zhao, Jiangnan Shao, Jiagang Zhu, Tingdong Yu, Zheng Zhu, Guan Huang, and Steven L. Waslander. Drivedreamer-policy: A geometry-grounded world-action model for unified generation and planning, 2026. URL https://arxiv.org/abs/2604.01765.

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. In *Advances in Neural Information Processing Systems*, 2025. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/2843fccca5bedd369a4764848b9bd546-Abstract-Conference.html.

A Additional Results and Implementation Details

A.1 Additional NAVSIM v1 Results

We report PDMS on NAVSIM v1 `navtest` in Table 8.

Table 8: Comparison on NAVSIM v1 `navtest`. Baseline scores are published results under their respective sensor configurations; JEWAM uses a single front-view camera.

Method	NC ↑	DAC ↑	EP ↑	TTC ↑	C ↑	PDMS ↑
Traditional E2E Planners						
UniAD (Hu et al., 2023)	97.80	91.90	78.80	92.90	100.00	83.40
TransFuser (Chitta et al., 2023)	97.70	92.80	79.20	92.80	100.00	84.00
WorldRFT (Yang et al., 2026)	97.80	96.80	81.70	94.00	100.00	87.80
DiffusionDrive (Liao et al., 2025)	98.20	96.20	82.20	94.70	100.00	88.10
WoTE (Li et al., 2025b)	98.50	96.80	81.90	94.90	99.90	88.30
SeerDrive (Zhang et al., 2025a)	98.40	97.00	83.20	94.90	99.90	88.90
SparseDriveV2 (Sun et al., 2026)	98.50	98.40	88.60	95.00	99.90	92.00
VLA-based Planners						
UniWorldVLA (Liu et al., 2026c)	98.70	96.70	83.20	96.10	100.00	89.40
DriveDreamer-Policy (Zhou et al., 2026)	98.40	97.10	83.50	95.10	100.00	89.20
AutoVLA (Zhou et al., 2025)	98.40	95.60	81.90	98.00	99.90	89.10
ReCogDrive (Li et al., 2026d)	97.90	97.30	87.30	94.90	100.00	90.80
ExploreVLA (Sheng et al., 2026)	98.80	98.40	83.50	96.50	99.90	90.40
SGDrive (Li et al., 2026b)	98.60	97.80	85.80	96.20	100.00	91.10
World-model / WAM-based Planners						
Epona (Zhang et al., 2025b)	97.90	95.10	80.40	93.80	99.90	86.20
PWM (Zhao et al., 2025)	98.60	95.90	81.80	95.40	100.00	88.10
DriveLaW (Xia et al., 2026)	99.00	97.10	81.30	96.70	100.00	89.10
DriveWAM (Shi et al., 2026)	98.30	98.10	84.30	95.20	100.00	90.10
DriveVLA-W0 (Li et al., 2026c)	98.70	99.10	83.30	95.30	99.30	90.20
JEWAM (Ours)	99.20	98.10	87.10	98.80	99.80	**92.08**

On NAVSIM v1 `navtest`, JEWAM achieves **92.08 PDMS**, exceeding SparseDriveV2 by 0.08 points, the strongest compared VLA planner SGDrive by 0.98 points, and the strongest compared WAM planner DriveVLA-W0 by 1.88 points (Table 8) (Sun et al., 2026; Li et al., 2026b;c). JEWAM also reaches 87.10 EP, the highest among the compared WAM-based planners, while maintaining 99.20 NC, 98.10 DAC, and 98.80 TTC. These are cross-method comparisons under the baselines' published sensor settings, rather than a controlled same-input evaluation. NAVSIM v1 PDMS and NAVSIM v2 EPDMS use different metric suites, so their absolute values should not be compared directly; JEWAM nevertheless ranks first among the compared methods in all three reported benchmark settings. These results evaluate the complete JEWAM system; the controlled training ablation in the main text provides the evidence for the benefit of future-state supervision.

For the same-checkpoint inference ablation in Table 5, enabling future-state prediction yields 92.10 PDMS on NAVSIM v1 `navtest`, while action-only inference yields 92.08 PDMS, a difference of -0.02 points when the future-state path is disabled.

A.2 Representation Comparison Interface

The representation study changes the encoder supplying both historical context and future-state targets. It compares ViT, MAE ViT-L, CroCo v2 ViT-L, DINOv3 ViT-L, and the 300M-parameter V-JEPA 2.1 ViT-L. The main JEWAM results instead use the larger V-JEPA 2.1 ViT-G initialization. The representation study and the main model results therefore serve different purposes: the former compares feature choices at comparable encoder scales, while the latter evaluates the selected full model.

All variants use four historical frames and eight future supervision frames, with causal temporal attention in the encoder and asymmetric routing in the joint predictor. Historical features are projected into the predictor, and the future-state head predicts targets in the selected representation space. V-JEPA uses two-frame tubelets; the other encoders use one-frame tokens. Thus, the common interface

matches the observation and prediction intervals without requiring identical latent sequence lengths. In the V-JEPA instance, the four historical and eight future frames correspond to two historical and four future latent time positions; the one-frame variants retain four and eight positions, respectively. All variants predict the same eight-pose action horizon.

The comparison shares the data, joint prediction setup, optimization settings, and training budget described in the main text. It retains differences in pretrained weights and tokenization, and does not isolate the effects of pretraining data, pretraining objective, or future targets independently. Future On/Off compares inference paths from the same jointly trained checkpoint; it is distinct from removing future supervision during training.

A.3 Structure of Predicted Future-State Features

To examine whether the spatial organization of encoder features carries into predicted future states, Figure 6 compares the predicted future-state features of five models on the same NAVSIM scene. All predictions use the same four observed frames and are displayed at one-second intervals over a four-second horizon. For each model, we fit one PCA basis to its current and four predicted feature maps and apply a shared 1st–99th percentile color scale across those time points. PCA bases are independent across models, so colors are not directly comparable across models. V-JEPA-300M uses two-frame latent tubelets, while the other encoders use one-frame tokens sampled at the same displayed times.

In this scene, V-JEPA's predicted maps preserve a distinct roadway region and localized regions around the nearby vehicles as the traffic configuration changes. The other representations show smoother gradients or less clearly separated regions in their predicted maps. This comparison illustrates that V-JEPA features can support structured future-state prediction, complementing the encoder-feature analysis in Figure 4 and the planning results in Table 3.

A.4 Controlled Prediction Variants

To compare alternative expert formulations within the MoT predictor, we construct three variants denoted FF, FD, and DD, where the first letter specifies the future-state expert and the second specifies the action expert. F denotes the feed-forward expert used by JEWAM, which maps learnable tokens to its outputs in a single forward pass. D denotes a Diffusion Transformer (DiT) expert that processes noisy tokens in its output space together with a flow timestep (Peebles & Xie, 2023). The timestep embedding is injected into each D expert through adaptive layer normalization (AdaLN), and the expert is trained using conditional flow matching (Lipman et al., 2023). At inference, each D expert starts from Gaussian noise and iteratively integrates its predicted velocity field. All variants retain the same student and teacher encoders, target future-state representations, ego-state conditioning, prediction horizons, expert dimensions, world–action interaction interface, and asymmetric attention mask described in Sec. 3.2.

FF versus FD. FF is the main JEWAM architecture described in Sec. 3.2, where the future-state and action experts both operate on learnable tokens and are supervised by the L1 objectives in Eq. 4. FD retains the feed-forward future-state expert and its L1 supervision but replaces the learnable action queries with noisy action tokens processed by a D action expert. During training, we sample action noise $\boldsymbol{\epsilon}_a \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and construct the noisy actions $\tilde{\mathbf{a}}_\tau = (1-\tau)\boldsymbol{\epsilon}_a + \tau \mathbf{a}_{n+1:n+m}$. The D action expert predicts the corresponding action velocity conditioned on historical latent states and the current ego state, while the future-state expert can attend to the action tokens under the shared asymmetric mask. During inference, FD initializes the action tokens from Gaussian noise and iteratively updates them to generate the action sequence. The FD objective is

$$\mathcal{L}_{\mathrm{FD}} = \mathcal{L}_l + \mathbb{E}_{\boldsymbol{\epsilon}_a, \tau}\left[\left\| v_\Phi^a\left(\tilde{\mathbf{a}}_\tau, \tau; \mathbf{z}^{\mathrm{hist}}, e_n\right) - \left(\mathbf{a}_{n+1:n+m} - \boldsymbol{\epsilon}_a\right)\right\|_2^2\right]. \tag{5}$$

FD versus DD. DD retains the D action expert from FD and further replaces the feed-forward future-state expert with a D future-state expert. The target future-state representations produced by the teacher encoder remain unchanged but now define the clean endpoint of a second flow path. We independently sample future-state noise $\boldsymbol{\epsilon}_l$ and action noise $\boldsymbol{\epsilon}_a$, and construct $\tilde{\mathbf{z}}_\tau^{\mathrm{fut}} = (1-\tau)\boldsymbol{\epsilon}_l + \tau \mathbf{z}^{\mathrm{fut}}$ and $\tilde{\mathbf{a}}_\tau = (1-\tau)\boldsymbol{\epsilon}_a + \tau \mathbf{a}_{n+1:n+m}$ using the same flow timestep. Under the same asymmetric mask, the future-state expert attends to both noisy future-state and action tokens, whereas the action expert

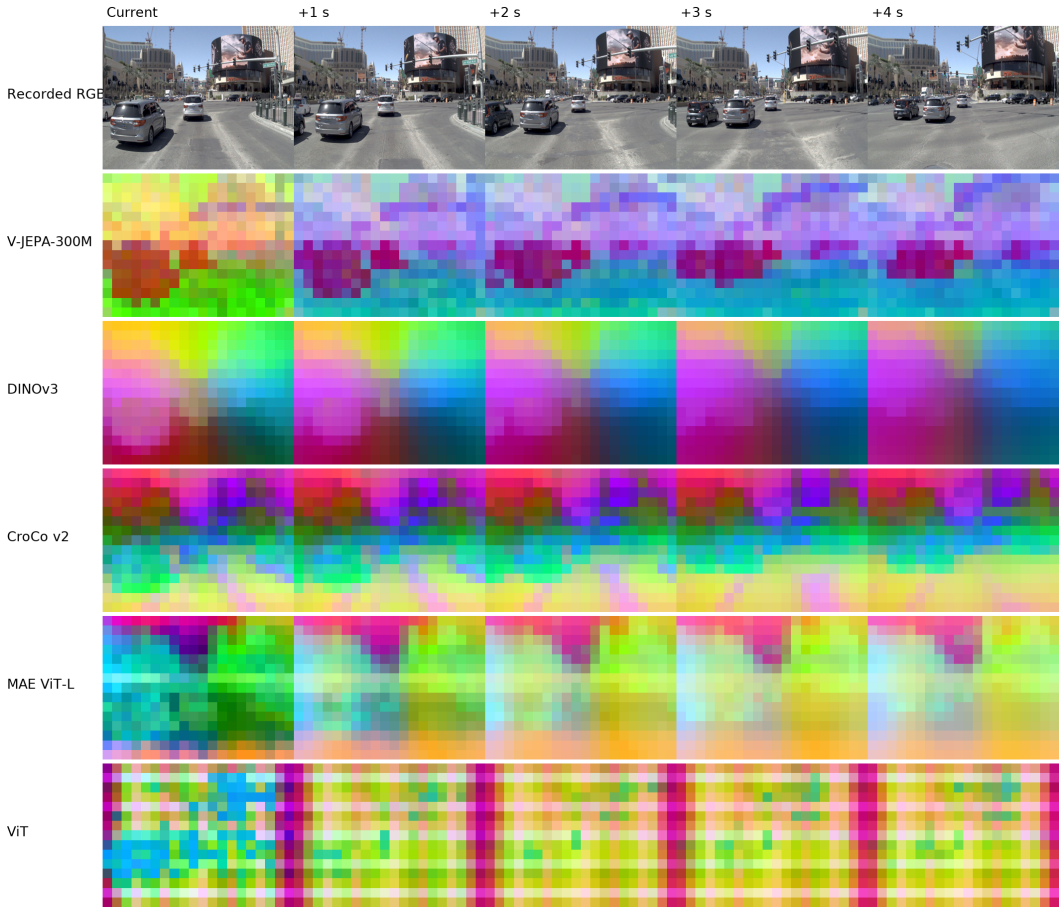

Figure 6: PCA visualization of observed and predicted future-state features in one matched NAVSIM scene. The top row shows recorded RGB frames; the feature rows show the current representation and predictions at $+1$ to $+4$ s for V-JEPA-300M, DINOv3, CroCo v2, MAE, and ViT. Each model uses one PCA basis and color scale shared across its five time points; colors are not aligned across models.

attends to noisy action tokens but not noisy future-state tokens. Both experts also attend to historical latent states and ego-state conditioning. We denote their predictions by $\hat{\mathbf{v}}_{l,\tau}$ and $\hat{\mathbf{v}}_{a,\tau}$, and the corresponding target velocities by $\mathbf{u}_l = \mathbf{z}^{\text{fut}} - \boldsymbol{\epsilon}_l$ and $\mathbf{u}_a = \mathbf{a}_{n+1:n+m} - \boldsymbol{\epsilon}_a$. At inference, both outputs are initialized from Gaussian noise and jointly integrated over the same sampling schedule. The DD objective is

$$\mathcal{L}^l_{\text{DD}} = \mathbb{E}_{\boldsymbol{\epsilon}_l,\boldsymbol{\epsilon}_a,\tau}\left[\|\hat{\mathbf{v}}_{l,\tau} - \mathbf{u}_l\|_2^2\right], \qquad \mathcal{L}^a_{\text{DD}} = \mathbb{E}_{\boldsymbol{\epsilon}_l,\boldsymbol{\epsilon}_a,\tau}\left[\|\hat{\mathbf{v}}_{a,\tau} - \mathbf{u}_a\|_2^2\right], \\ \mathcal{L}_{\text{DD}} = \mathcal{L}^l_{\text{DD}} + \mathcal{L}^a_{\text{DD}}. \tag{6}$$