# RETRANDRIVE: A REINFORCED TRANSITION-AWARE FRAMEWORK FOR VISION-LANGUAGE-ACTION AUTONOMOUS DRIVING

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

## ABSTRACT

Autonomous driving policies are commonly optimized by evaluating the quality of each predicted plan in isolation, despite being executed through continuous replanning. This formulation overlooks a distinct aspect of sequential decision-making: the quality of transitions between consecutive plans. As a result, a planner can produce individually high-quality trajectories while exhibiting unstable behavior over time, repeatedly altering its intended motion without a corresponding change in the scene. The missing ingredient is an explicit objective over inter-plan transitions, capturing whether successive predictions evolve coherently with the changing scene rather than merely optimizing each plan in isolation. We present ReTranDrive, which defines this objective by shifting the optimization target from single-step planning quality to sequential execution coherence: scoring a plan against the current observation alone becomes a transition-aware formulation over consecutive plan pairs. Applied to a VLM-conditioned diffusion planner as a differentiable term during imitation and a simulator-verifiable reward during reinforcement, it regularizes only those inter-plan changes not warranted by scene evolution rather than enforcing unconditional temporal consistency. To resolve the credit-assignment challenge inherent in pairwise scoring, we introduce an asymmetric scheme that assigns inconsistency strictly to the decision responsible for it. On the NAVSIM benchmark, ReTranDrive attains a state-of-the-art EPDMS of 90.94 under the extended metric and a PDMS of 94.40, reaching 99.58% of human driving performance.

## 1 INTRODUCTION

The planning objective in end-to-end driving has evolved from single-trajectory regression (Hu et al., 2023; Jiang et al., 2023), through multi-modal generation with diffusion and flow models (Liao et al., 2025; Xing et al., 2025), to semantic conditioning with vision-language models (Xiong et al., 2026). Each step broadened what a single decision can express, yet all define the objective over one decision at a time: a plan is generated and scored from the current observation alone, and its relation to the plan committed a step earlier is left unconstrained. Driving, however, unfolds as a sequence of such decisions, and behavior quality depends as much on how consecutive plans relate as on the merit of any individual one.

Bringing the world knowledge and reasoning of Vision-Language Models into the planner (Mao et al., 2023; Tian et al., 2024; Hwang et al., 2024; Sima et al., 2024) has produced VLA planners that inject semantics at different stages: some emit high-level commands or textual trajectories that steer a downstream policy (Jiang et al., 2026; Fu et al., 2025), some fuse VLM features with perception representations before a shared action head (Wang et al., 2024), and others condition a generative trajectory decoder on VLM hidden states (Xiong et al., 2026). What none of them alters is the objective, which still rewards each plan on its own, so richer per-frame semantics sharpen individual decisions without making consecutive plans coherent. In contrast, ReTranDrive leaves per-frame semantics to these planners and instead shifts their optimization objective from single-step planning quality to sequential execution coherence.



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.

Sequential execution coherence, however, cannot be imposed by a fixed rule. The remedies that sequential control has long relied on, whether they regularize consecutive plans, average overlapping predictions, or hard-code hysteresis, are inference-time heuristics that stabilize behavior only by anchoring it to the policy's own past output. Each therefore trades away the decisive maneuvers that safety sometimes demands, leaving the policy overly conservative and slow to act. What is missing is a learned criterion for when a prior decision should be honored and when it should be overturned. Such a criterion is unattainable by imitation, since expert logs record only the trajectory a driver executed and never the alternatives weighed and discarded, so the supervision that would separate a warranted change from an unwarranted one is absent by construction. This points to reinforcement learning, in which a planner is rewarded for the coherence of the transitions it actually generates rather than for matching a fixed expert target (Jiang et al., 2025; Gao et al., 2026; Huang et al., 2025).

Motivated by this analysis, we present **ReTranDrive**, a reinforced transition-aware framework for VLA-based planning that introduces a reference-based transition objective in place of the single-step scoring of each plan. The objective treats the plan from the preceding observation as a reference and quantifies the transition from it to the current prediction, recasting single-frame trajectory scoring as an assessment of sequential execution coherence. We embed this objective into a VLM-conditioned diffusion planner in two complementary forms, a differentiable term during imitation and a simulator-verifiable reward during reinforcement, so that the planner jointly optimizes single-step planning quality and sequential execution coherence, selectively regularizing unnecessary plan transitions rather than indiscriminately enforcing temporal consistency across consecutive predictions.

We evaluate ReTranDrive on the NAVSIM navtest benchmark (Dauner et al., 2024). It attains a state-of-the-art EPDMS of 90.94 under the NAVSIM-v2 extended metric and a PDMS of 94.40 under the v1 metric, reaching 99.58% of human driving performance, along with improved continuity across consecutive plans. Our contributions are threefold:

- We propose ReTranDrive, a reinforced transition-aware framework for VLA-based planning that shifts the optimization objective from single-step planning quality to sequential execution coherence, addressing a mismatch between single-step planning and the sequential nature of driving, obscured both by per-frame evaluation and by conditioning that excludes the previous decision.
- We propose a reference-based transition objective, realized during imitation as a differentiable term and during diffusion-policy reinforcement as a direct, simulator-verifiable reward, with a credit-assignment design that charges an inconsistency to the decision responsible for it rather than penalizing change unconditionally.
- We conduct extensive experiments on NAVSIM, where ReTranDrive achieves a state-of-the-art EPDMS of 90.94 and a competitive PDMS of 94.40, and provide analyses isolating the effect of transition-aware optimization.

## 2 RELATED WORK

### 2.1 END-TO-END AND VISION-LANGUAGE-ACTION PLANNING

End-to-end driving unifies perception and planning into a single differentiable pipeline that regresses future trajectories from raw sensors. Discriminative planners such as UniAD (Hu et al., 2023) and VAD (Jiang et al., 2023) couple the two stages efficiently, but their deterministic regression tends toward a single averaged mode and struggles to capture behavioral multi-modality; probabilistic and generative formulations (Chen et al., 2024; Chitta et al., 2022) partially relax this rigidity. To cope with long-tail events that geometry alone cannot resolve, a growing body of work injects the world knowledge of Vision-Language Models into the stack. These vision-language-action (VLA) planners differ mainly in where semantics enter: some emit high-level commands or textual trajectories that steer a downstream policy (Mao et al., 2023; Tian et al., 2024; Hwang et al., 2024; Jiang et al., 2026; Fu et al., 2025), some fuse VLM features with perception representations before a shared action head (Wang et al., 2024), and others condition a generative trajectory decoder on VLM hidden states (Xiong et al., 2026). Instead of adding another semantic injection point, ReTranDrive optimizes directly for sequential execution coherence.

### 2.2 Diffusion-based Planning and Policy Reinforcement

Diffusion models, originally developed for generative modeling (Ho et al., 2020) and visuomotor control (Chi et al., 2025), have become a strong paradigm for multi-modal trajectory generation. DiffusionDrive (Liao et al., 2025), GoalFlow (Xing et al., 2025), and Diffusion Planner (Zheng et al., 2025) recast planning as a denoising process, producing smooth and diverse trajectories that outperform one-shot regression. The sub-optimality that behavior cloning inherits from expert demonstrations has since motivated a turn to reinforcement learning: AlphaDrive (Jiang et al., 2025) applies GRPO over discrete meta-actions, RAD (Gao et al., 2026) trains a policy in a photorealistic 3DGS environment, and Gen-Drive (Huang et al., 2025) couples diffusion generation with a learned reward model. ReCogDrive (Xiong et al., 2026) likewise drives a VLM-conditioned diffusion planner with a per-frame simulator reward. In every case, however, the reward is computed on a single trajectory in isolation, so the optimizer is never informed about, and thus cannot improve, sequential execution coherence.

### 2.3 Sequential Coherence and Decision Commitment

Sequential control has long kept consecutive plans in agreement through fixed, inference-time corrections: receding-horizon optimizers are warm-started from the previous solution, action-chunking policies with temporal ensembling (Zhao et al., 2023; Black et al., 2024) average overlapping predictions, and production stacks impose hand-tuned hysteresis on mode switches. Each is designed by hand and applied outside the learned policy; ReTranDrive instead learns the criterion directly via a reference-based transition reward and diffusion-policy reinforcement.

## 3 PRELIMINARIES

**Problem Definition and Multi-Frame Notation.** We cast trajectory planning as a recurring decision process. At each planning step $k$, the ego vehicle receives an observation $O_k = (\mathcal{I}_k, S_k, \ell_k)$ comprising multi-view camera images $\mathcal{I}_k$, ego status $S_k$ (e.g., velocity and acceleration), and a navigation command $\ell_k$, and predicts a future trajectory

$$Y_k = \{y_k^0, y_k^1, \ldots, y_k^{T-1}\}, \qquad y_k^t \in \mathbb{R}^3, \tag{1}$$

of $T$ waypoints spaced by a fixed interval $\Delta t_{\text{sample}}$, where each $y_k^t$ stacks the planar position and heading at the $t$-th future step. A planner is a policy $Y_k \sim \pi_\theta(\cdot \mid O_k)$.

Because the vehicle replans continually, consecutive steps are not independent: the plan $Y_{k-1}$ committed at the previous step and the current plan $Y_k$ span overlapping horizons. With an inter-step interval $\Delta t_{\text{obs}}$ and waypoint spacing $\Delta t_{\text{sample}}$, the two plans are offset by

$$s = \frac{\Delta t_{\text{obs}}}{\Delta t_{\text{sample}}} \tag{2}$$

future steps, so the tail of $Y_{k-1}$ and the head of $Y_k$ describe the same absolute time interval. We isolate this shared segment as

$$Y_k^{\text{ov}} = \{y_k^t\}_{t=0}^{T-s-1}, \qquad Y_{k-1}^{\text{ov}} = \{y_{k-1}^t\}_{t=s}^{T-1}, \tag{3}$$

so that $y_k^t$ and $y_{k-1}^{t+s}$ predict the ego state at the same physical time. Comparing $Y_{k-1}^{\text{ov}}$ with $Y_k^{\text{ov}}$ therefore measures how the current decision revises the one committed a step earlier over their shared horizon, rather than contrasting two unrelated trajectories. ReTranDrive is defined on exactly this segment.

**Diffusion Action Expert.** We realize the policy $\pi_\theta$ as a conditional diffusion model over trajectories, following recent diffusion planners (Ho et al., 2020; Liao et al., 2025; Xiong et al., 2026). Writing $x_0 := Y_k$ and dropping $k$ where unambiguous, a forward Markov process gradually corrupts $x_0$ into $x_1, \ldots, x_N$ with Gaussian noise, and the expert learns to reverse it. A trajectory is generated from $x_N \sim \mathcal{N}(0, I)$ through $N$ Gaussian denoising steps

$$\pi_\theta(x_{i-1} \mid x_i, c) = \mathcal{N}\big(x_{i-1};\ \mu_\theta(x_i, i, c),\ \sigma_i^2 I\big), \tag{4}$$

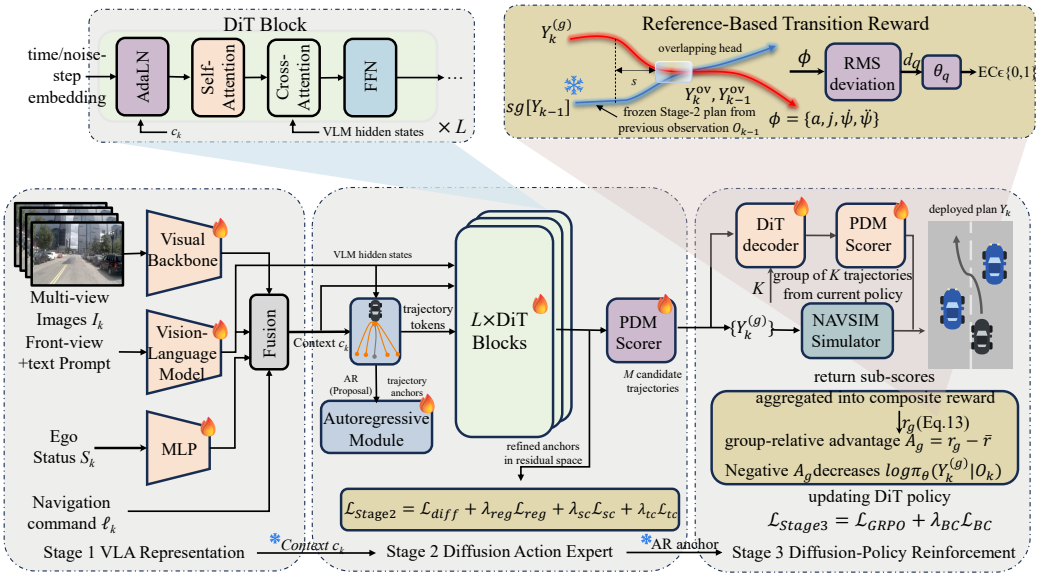


Figure 1: Overview of ReTranDrive. A VLA-conditioned diffusion planner trained in three stages: VLA representation, a diffusion action expert learned by imitation, and diffusion-policy reinforcement driven by a reference-based transition reward that scores the current plan against the frozen previous plan for sequential execution coherence.

conditioned on a context $c$ obtained from $O_k$. The probability the policy assigns to a trajectory then factorizes over its denoising chain,

$$\log \pi_\theta(x_{0:N} \mid c) = \log p(x_N) + \sum_{i=1}^{N} \log \pi_\theta(x_{i-1} \mid x_i, c), \tag{5}$$

a decomposition we exploit during reinforcement fine-tuning.

**Group-Relative Policy Optimization.** To improve the expert beyond imitation, we fine-tune it with a group-relative policy gradient (Jiang et al., 2025; Xiong et al., 2026). For an observation $O_k$, a group of $K$ trajectories $\{Y_k^{(g)}\}_{g=1}^{K}$ is sampled from the current policy, and each draw yields a deployed trajectory with a scalar simulator reward $r_g$. A value-free advantage is obtained by centering the reward within the group,

$$A_g = r_g - \bar{r}, \qquad \bar{r} = \frac{1}{K} \sum_{h=1}^{K} r_h, \tag{6}$$

and the policy is updated to raise the likelihood of above-average draws; combined with the chain factorization above, this signal reaches every denoising step.

## 4 METHODOLOGY

**Overview.** In this section, we present the overall framework and training paradigm of ReTranDrive (Figure 1), which pairs a VLA-conditioned diffusion planner with a three-stage training scheme. First, we adapt a vision-language model into a VLA representation that encodes each multi-view observation, together with the ego status, into the semantic and geometric context a plan requires. Second, we train a diffusion action expert on this representation by imitation, augmenting the standard objective with a differentiable coherence term that regularizes adjacent plans and provides a coherence-aware initialization. Finally, we fine-tune the expert with reinforcement against a simulator-verifiable reward defined over a plan and its predecessor, optimizing for the sequential execution coherence that imitation alone cannot supply.

### 4.1 VLA-CONDITIONED DIFFUSION ACTION EXPERT

**VLA representation (Stage 1).** Each observation $O_k$ is encoded into a conditioning context $c_k$ for the planner. A visual backbone extracts geometric and appearance features from the multi-view images, while a driving-adapted vision-language model, fine-tuned once in Stage 1 and frozen afterwards, contributes scene-level semantic priors through its hidden states. Together with an embedding of the ego status $S_k$, these features are organized into the scene and ego tokens that constitute $c_k$ and condition the diffusion action expert.

**Diffusion action expert (Stage 2).** Conditioned on the context $c_k$, an autoregressive module first emits a set of trajectory anchors, which a Diffusion Transformer (DiT) then refines. Rather than synthesizing a trajectory from scratch, the DiT models the residual between each anchor and the expert trajectory (Liao et al., 2025). It is optimized by the $\epsilon$-prediction objective

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{i,\epsilon}\big\|\epsilon_\theta(x_i, i, c_k) - \epsilon\big\|_2^2, \tag{7}$$

alongside a winner-takes-all regression $\mathcal{L}_{\text{reg}}$ that draws the closest anchor toward the expert trajectory and a scoring loss $\mathcal{L}_{\text{sc}}$ that trains the candidate ranker, forming the imitation objective

$$\mathcal{L}_{\text{IL}} = \mathcal{L}_{\text{diff}} + \lambda_{\text{reg}}\,\mathcal{L}_{\text{reg}} + \lambda_{\text{sc}}\,\mathcal{L}_{\text{sc}}. \tag{8}$$

At inference the module produces $M$ candidates and deploys the one the scorer ranks highest. Imitation alone leaves the coherence between successive plans unconstrained, the gap that the coherence term introduced next begins to close.

### 4.2 REFERENCE-BASED TRANSITION COHERENCE

**Coherence measure.** To quantify how a plan revises the one before it, the current plan $Y_k$ is compared against a reference $Y_{k-1}$, fixed to the frozen Stage 2 trajectory deployed one step earlier rather than a plan from the model being updated; holding the reference fixed while the policy changes is what makes the credit assignment developed below well posed. Both plans are rolled out by the simulator into absolute-coordinate ego states, which removes the mismatch between their ego-centric frames, and are aligned by the offset $s$ so that the comparison is restricted to their shared segment $Y_{k-1}^{\text{ov}}, Y_k^{\text{ov}}$. On this segment, four dynamic features

$$\phi = \{a,\ j,\ \dot{\psi},\ \ddot{\psi}\}, \tag{9}$$

the acceleration magnitude, jerk, yaw rate, and yaw acceleration, are compared through the root-mean-square deviation of each across the overlap $\mathcal{T}_{\text{ov}}$,

$$d_q(Y_{k-1}, Y_k) = \sqrt{\frac{1}{|\mathcal{T}_{\text{ov}}|}\sum_{t\in\mathcal{T}_{\text{ov}}}\big(q_t(Y_k) - q_t(Y_{k-1})\big)^2}, \qquad q \in \phi. \tag{10}$$

Two plans are deemed consistent when all four deviations stay within tolerance, giving the binary indicator

$$\text{EC}(Y_{k-1}, Y_k) = \prod_{q\in\phi} \mathbb{1}[d_q(Y_{k-1}, Y_k) \le \theta_q]\,, \tag{11}$$

with per-feature tolerances $\theta_q$. This indicator is exactly the extended-comfort quantity the simulator scores, making the signal simulator-verifiable rather than a hand-designed proxy.

**Coherence as a training signal.** Because the indicator is non-differentiable, it enters training in two complementary forms. During imitation (Stage 2) it is relaxed into a smooth hinge that penalizes only violations,

$$\mathcal{L}_{\text{tc}} = \frac{1}{|\phi|}\sum_{q\in\phi}\text{SmoothL1}\Big(\max\Big(0,\ \tfrac{d_q}{\theta_q} - 1\Big)\Big)\,, \tag{12}$$

which augments the imitation objective, $\mathcal{L}_{\text{Stage2}} = \mathcal{L}_{\text{IL}} + \lambda_{\text{tc}}\,\mathcal{L}_{\text{tc}}$, yielding a coherence-aware initialization without invoking the simulator. During reinforcement (Stage 3), where a differentiable surrogate is no longer needed, the exact indicator is optimized directly as the transition reward $r_{\text{trans}} = \text{EC} \in \{0, 1\}$, developed in the reinforcement objective below.

### 4.3 Diffusion-Policy Reinforcement with Consecutive-Plan Credit Assignment

**Composite reward.** For each observation, a group of $K$ trajectories is drawn from the current policy. For draw $g$, the scorer selects the deployed trajectory $Y_k^{(g)}$, whose simulator sub-scores are aggregated into a scalar reward that couples a multiplicative safety gate with a weighted average of progress, time-to-collision, single-frame comfort, and the transition term,

$$r_g = m_g \cdot \frac{5\,\mathrm{EP}_g + 5\,\mathrm{TTC}_g + 2\,\mathrm{C}_g + 2\,\mathrm{EC}_g}{5+5+2+2}, \qquad m_g = \mathrm{NC}_g \cdot \mathrm{DAC}_g, \tag{13}$$

where $\mathrm{EC}_g = \mathrm{EC}(Y_{k-1}, Y_k^{(g)})$ is scored against the frozen reference. When no predecessor is available, the transition term is dropped and its weight removed from the denominator, rather than scored as zero.

**Advantage and credit assignment.** A group-relative advantage $A_g = r_g - \bar{r}$, at a fixed scale, drives the update. Since the transition reward couples the current plan to its predecessor, an inconsistency implicates two plans at once, and penalizing both would drive the policy toward a trivially self-consistent, sluggish mode. ReTranDrive resolves this asymmetrically. As the reference $Y_{k-1}$ is the frozen Stage 2 trajectory, it is treated as a stop-gradient quantity $\mathrm{sg}[Y_{k-1}]$ that enters only through the reward and receives no gradient, so the advantage acts solely on the current draw,

$$A_g < 0 \implies \text{decrease } \log \pi_\theta\big(Y_k^{(g)} \mid O_k\big). \tag{14}$$

An inconsistency is thereby charged entirely to the decision that produced it, never to the reference. EC acts as a reward rather than a hard constraint, so a decisive change that improves the remaining sub-scores can still be favored; necessary maneuvers are therefore preserved.

**Optimization objective.** Through the chain factorization of the diffusion likelihood, the group-relative signal is distributed over the denoising steps of each draw,

$$\mathcal{L}_{\mathrm{GRPO}} = -\frac{1}{KN} \sum_{g=1}^{K} \sum_{i=1}^{N} w_i\, A_g\, \log \pi_\theta\big(x_{i-1}^{(g)} \mid x_i^{(g)}, c_k\big), \tag{15}$$

where $w_i$ follows a geometric schedule with ratio $\gamma$ that down-weights the early, high-variance denoising steps. To keep exploration near the imitation policy, trajectories $\tilde{x}$ sampled from the frozen Stage 2 decoder $\pi_{\bar{\theta}}$ are anchored by a behavior-cloning term

$$\mathcal{L}_{\mathrm{BC}} = -\mathbb{E}_{\tilde{x}\sim\pi_{\bar{\theta}}}\big[\log \pi_\theta(\tilde{x} \mid c_k)\big], \tag{16}$$

giving the Stage 3 objective $\mathcal{L}_{\mathrm{Stage3}} = \mathcal{L}_{\mathrm{GRPO}} + \lambda_{\mathrm{BC}}\,\mathcal{L}_{\mathrm{BC}}$. Since each group is resampled from the current policy at every update, the behavior policy coincides with the target, so no importance-ratio correction or clipping is required.

## 5 Experiments

### 5.1 Experimental Setup

**Datasets.** We evaluate ReTranDrive on NAVSIM (Dauner et al., 2024), a data-driven benchmark for vision-based autonomous driving that pairs real-world driving logs with a non-reactive simulation protocol for scalable evaluation. It provides navtrain for training and navtest for evaluation. We report the v1 score (PDMS) and the extended score (EPDMS), which augments the v1 metrics with driving-direction and traffic-light compliance, lane keeping, history comfort, and extended comfort.

**Implementation Details.** We train ReTranDrive on 16 NVIDIA A800 GPUs through a three-stage pipeline. Stage 1 fine-tunes a released driving-oriented VLM (Xiong et al., 2026) on our driving data for 5 epochs; the resulting VLM is then frozen, and only its cached hidden states are used downstream. Stage 2 trains the diffusion action expert and the scorer for 40 epochs, adapting the visual backbone with LoRA while keeping its base and the VLM frozen. Stage 3 fine-tunes the diffusion decoder and the scorer by diffusion-policy reinforcement for 4 epochs, holding the VLM and a frozen copy of the Stage-2 decoder fixed. All stages use AdamW with a per-GPU batch size of 8 and a base learning rate of $2{\times}10^{-5}$ in Stage 1, $2{\times}10^{-4}$ in Stage 2, and $2{\times}10^{-5}$ in Stage 3, with cosine decay after $10\%$ linear warmup. The imitation objective uses $\lambda_{\mathrm{reg}}{=}1$ and $\lambda_{\mathrm{sc}}{=}1$, with coherence weight $\lambda_{\mathrm{tc}}{=}0.01$; Stage 3 samples $K{=}8$ trajectories per observation, discounts denoising steps by $\gamma{=}0.6$, and sets $\lambda_{\mathrm{BC}}{=}0.1$. Inference uses a 12-step DDIM process with $M{=}64$ candidates.

Table 1: Performance comparison on the NAVSIM navtest benchmark. Best results are highlighted in bold.

| Method | Venue | PDMS↑ | NC↑ | DAC↑ | EP↑ | TTC↑ | Comf.↑ |
|---|---|---|---|---|---|---|---|
| Human | – | 94.8 | 100.0 | 100.0 | 87.5 | 100.0 | 99.9 |
| **End-to-End Methods** | | | | | | | |
| UniAD Hu et al. (2023) | CVPR'23 | 83.4 | 97.8 | 91.9 | 78.8 | 92.9 | **100** |
| iPad Guo et al. (2026) | RA-L'26 | 91.7 | 98.6 | 98.3 | 88.0 | 94.9 | **100** |
| DiffusionDrive Liao et al. (2025) | CVPR'25 | 88.1 | 98.2 | 96.2 | 82.2 | 94.7 | **100** |
| Centaur Sima et al. (2025) | arXiv'25 | 92.6 | **99.5** | 98.9 | 85.9 | **98.0** | **100** |
| DriveSuprim Yao et al. (2026) | AAAI'26 | 93.5 | 98.6 | 98.6 | 91.3 | 95.5 | **100** |
| RAP-DINO Feng et al. (2026) | ICLR'26 | 93.8 | 99.1 | 98.9 | 90.3 | 96.7 | **100** |
| DrivoR Kirby et al. (2026) | CVPR'26 | 93.7 | 99.0 | 98.9 | 90.0 | 96.7 | **100** |
| **World Model Methods** | | | | | | | |
| LAW Li et al. (2024) | ICLR'25 | 84.6 | 96.4 | 95.4 | 81.7 | 88.7 | 99.9 |
| FSDrive Zeng et al. (2025) | NeurIPS'25 | 85.1 | 98.2 | 93.8 | 80.1 | 93.3 | 99.9 |
| Epona Zhang et al. (2025) | ICCV'25 | 86.2 | 97.9 | 95.1 | 80.4 | 93.8 | 99.9 |
| PWM Zhao et al. (2025) | NeurIPS'25 | 88.1 | 98.6 | 95.9 | 81.8 | 95.4 | **100** |
| WoTE Li et al. (2025) | ICCV'25 | 88.3 | 98.5 | 96.8 | 81.9 | 94.9 | 99.9 |
| ResWorld Zhang et al. (2026) | arXiv'26 | 89.0 | 98.9 | 96.5 | 83.1 | 95.6 | **100** |
| DriveLaW Xia et al. (2026) | CVPR'26 | 89.1 | 99.0 | 97.1 | 81.3 | 96.7 | **100** |
| **Vision-Language Model Methods** | | | | | | | |
| UniVLA Wang et al. (2026b) | ICLR'26 | 81.7 | 96.9 | 91.1 | 76.8 | 91.7 | 96.7 |
| DriveVLA-W0 Li et al. (2026b) | ICLR'26 | 90.2 | 98.7 | **99.1** | 83.3 | 95.3 | 99.9 |
| ReCogDrive Xiong et al. (2026) | ICLR'26 | 90.8 | 97.9 | 97.3 | 87.3 | 94.9 | **100.0** |
| SGDrive Li et al. (2026a) | arXiv'26 | 91.1 | 98.6 | 97.8 | 85.8 | 96.2 | **100.0** |
| SpanVLA Zhou et al. (2026) | arXiv'26 | 90.3 | 99.1 | 97.1 | 86.3 | 95.2 | **100.0** |
| LatentVLA Xie et al. (2026) | arXiv'26 | 92.4 | 98.9 | 98.2 | 88.2 | 96.0 | **100.0** |
| **ReTranDrive (ours)** | – | **94.4** | 99.1 | 99.0 | **91.9** | 96.4 | **100.0** |

Table 2: Performance comparison on the NAVSIM navtest benchmark under the extended metrics.

| Method | Venue | NC↑ | DAC↑ | DDC↑ | TL↑ | EP↑ | TTC↑ | LK↑ | HC↑ | EC↑ | EPDMS↑ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **End-to-End Methods** | | | | | | | | | | | |
| DiffusionDrive Liao et al. (2025) | CVPR'25 | 98.2 | 95.9 | 99.4 | 99.8 | 87.5 | 97.3 | 96.8 | 98.3 | 87.7 | 84.5 |
| DiffusionDriveV2 Zou et al. (2025) | arXiv'25 | 97.7 | 96.6 | 99.2 | 99.8 | 88.9 | 97.2 | 96.0 | 97.8 | 91.0 | 87.5 |
| **World Model Methods** | | | | | | | | | | | |
| WoTE Li et al. (2025) | ICCV'25 | 98.5 | 96.8 | 98.8 | 99.8 | 86.1 | 97.9 | 95.5 | 98.3 | 82.9 | 87.7 |
| DreamerAD Yang et al. (2026) | arXiv'26 | 98.0 | 97.2 | 99.5 | 99.8 | 87.8 | 97.4 | 97.5 | 98.3 | 72.4 | 87.7 |
| PWM Zhao et al. (2025) | NeurIPS'25 | 98.8 | 95.9 | 99.4 | 99.9 | 86.4 | 98.4 | 97.6 | 98.3 | 85.3 | 88.2 |
| DriveLaW Xia et al. (2026) | CVPR'26 | 98.7 | 96.9 | **99.6** | 99.8 | 87.5 | 98.3 | 97.6 | **98.4** | 77.4 | 88.6 |
| Latent-WAM Wang et al. (2026a) | arXiv'26 | 98.1 | 97.3 | **99.6** | 99.8 | 87.7 | 97.3 | 97.6 | 98.1 | **87.3** | 89.3 |
| **Vision-Language Model Methods** | | | | | | | | | | | |
| DriveVLA-W0 Li et al. (2026b) | ICLR'26 | 98.5 | **99.1** | 98.0 | 99.7 | 86.4 | 98.1 | 93.2 | 97.9 | 58.9 | 86.1 |
| SGDrive Li et al. (2026a) | arXiv'26 | 98.6 | 94.3 | 99.5 | 99.9 | 86.0 | 97.9 | 96.1 | 98.3 | 85.9 | 86.2 |
| DriveWorld-VLA Liu et al. (2026) | arXiv'26 | 98.6 | **99.1** | **99.6** | 99.8 | 87.4 | 97.9 | 97.0 | 97.8 | 78.6 | 86.8 |
| CoWorld-VLA Huang et al. (2026) | arXiv'26 | **99.1** | 97.0 | **99.6** | 99.9 | 87.8 | 98.5 | **97.7** | 98.2 | 86.2 | 90.0 |
| **ReTranDrive (ours)** | – | 99.0 | 98.9 | 99.0 | **100.0** | **93.4** | **98.7** | 95.8 | 97.1 | 69.8 | **90.9** |

### 5.2 Main Results

As shown in Table 1, ReTranDrive attains a PDMS of 94.4, exceeding the best prior end-to-end (93.8) and VLA-based (92.4) planners and reaching 99.58% of human driving performance (94.8). Table 2 reports extended metrics, where ReTranDrive achieves a state-of-the-art EPDMS of 90.9. Compared with the strongest prior VLA method, CoWorld-VLA, and the strongest world-model method, Latent-WAM, it improves EPDMS by 0.9 and 1.6, respectively, while attaining the highest ego progress (93.4) and perfect traffic-light compliance (100.0).

Figure 2 compares ReTranDrive with ReCogDrive and SGDrive on two interactive scenarios. In the intersection case (top row), the ReCogDrive and SGDrive predictions extend into the path of the vehicle crossing the junction, whereas ReTranDrive yields and stops short of it. In the left-turn case (bottom row), both baselines drift left and approach the rear of the oncoming vehicle, while ReTranDrive follows the expert path and stays within its own lane.

### 5.3 Ablation

As shown in Table 3, each stage contributes to the final performance. The base model (ID 0) reaches a PDMS of 84.42 and an EPDMS of 84.71. Adding the VLA representation (ID 1) raises PDMS to 90.92 (+6.50) and EPDMS to 87.89 (+3.18). Adding the diffusion action expert with the imitation-




Figure 2: Qualitative comparison with prior methods on representative NAVSIM scenarios. Ground-truth trajectories are in green; ReCogDrive, SGDrive, and ReTranDrive predictions are in light blue, orange, and purple, respectively.

Table 3: Ablation study on the proposed components of ReTranDrive. Each row is cumulative, adding one stage to the previous one and starting from the base model (ID 0) that uses none of the three components; we evaluate the effect of the VLA representation, the diffusion action expert, and the diffusion-policy reinforcement on NAVSIM.

| ID | VLA representation | Diffusion action expert | Diffusion-policy reinforcement | Metrics | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | NC↑ | DAC↑ | EP↑ | TTC↑ | Comf.↑ | PDMS | EPDMS |
| 0 (Base) | × | × | × | 98.11 | 93.22 | 78.48 | 93.78 | 99.99 | 84.42 | 84.71 |
| 1 (Stage 1) | ✓ | × | × | 98.63 | 97.65 | 85.59 | 96.27 | 100.00 | 90.92 | 87.89 |
| 2 (Stage 2) | ✓ | ✓ | × | 99.32 | 99.28 | 89.37 | 97.65 | 100.00 | 94.06 | 90.10 |
| 3 (Stage 3) | ✓ | ✓ | ✓ | 99.05 | 98.95 | 91.93 | 96.44 | 100.00 | **94.40** | **90.94** |

stage coherence term (ID 2) further improves PDMS to 94.06 (+3.14) and EPDMS to 90.10 (+2.21), providing a coherence-aware initialization. Adding diffusion-policy reinforcement with the transition reward (ID 3) further improves EPDMS to 90.94 (+0.84) and PDMS to 94.40 (+0.34), with ego progress rising from 89.37 to 91.93, showing that the transition objective improves sequential behavior without suppressing progress.

Figure 3 illustrates the effect of the three training stages on two representative scenarios. In the intersection case (top row), the Stage-1 prediction drifts beyond the lane boundary and departs from the expert path, whereas Stage 3 tracks the expert throughout the maneuver, with Stage 2 in between. In the multi-lane case (bottom row), the Stage-1 and Stage-2 predictions reach into the lead vehicle, a rear-end conflict that imitation alone does not resolve; Stage 3 instead holds behind the lead vehicle in this case, resolving the conflict while staying closest to the expert.




Figure 3: Qualitative comparison across the three training stages on representative NAVSIM scenarios. Ground-truth trajectories are in green; Stage 1, Stage 2, and Stage 3 predictions are in blue, orange, and purple, respectively.

Table 4: Comparison of transition reward forms.

| Transition reward | EC | EP | EPDMS |
|---|---|---|---|
| None (per-frame) | 65.13 | 93.39 | 90.45 |
| Continuous | 66.73 | 93.57 | 90.63 |
| Binary EC (ours) | 69.80 | 93.40 | 90.94 |

Table 5: Ablation on credit assignment.

| ID | Frozen reference | Stop-gradient | EC | EP | EPDMS |
|---|---|---|---|---|---|
| 1 | ✓ | ✓ | 69.80 | 93.40 | 90.94 |
| 2 | × | ✓ | 68.98 | 93.51 | 90.78 |
| 3 | × | × | 72.12 | 92.27 | 90.71 |



Front Concat View ID 1 ID 2 ID 3

Figure 4: Qualitative comparison of the credit-assignment variants of Table 5 on representative NAVSIM scenarios. Ground-truth trajectories are in green; ID 1 (ours), ID 2, and ID 3 predictions are in blue, orange, and purple, respectively.

Table 4 compares the transition reward with two alternatives. A per-frame reward yields an EC of 65.13 and an EPDMS of 90.45, while replacing the binary indicator with a continuous SmoothL1 penalty on $d_q$ raises both to 66.73 and 90.63 but still trails the binary form, which attains the best EC (69.80) and EPDMS (90.94), with ego progress essentially unchanged across the three variants.

Table 5 isolates the credit-assignment design. Using the fixed Stage-2 plan as the reference together with a stop-gradient attains an EC of 69.80 and the best EPDMS of 90.94, with ego progress at 93.40. Replacing the reference with the model's own previous plan while keeping the stop-gradient lowers EC to 68.98 and EPDMS to 90.78. Removing the stop-gradient charges inconsistency to both plans, raising EC to 72.12 but dropping progress (92.27) and EPDMS (90.71) due to a sluggish, overly self-consistent mode.

Figure 4 visualizes the credit-assignment variants of Table 5 on two left-turn scenarios. In the intersection case (top row), ID 1 and ID 2 both follow the expert's left turn, with ID 2 extending slightly further and bending differently, while ID 3 barely turns and drives straight across the intersection, separating entirely from the expert path. The curved-road case (bottom row) exhibits the same ordering, with the endpoint overrun growing from ID 1 to ID 3 and ID 3 drifting toward the oncoming lane.

## 6 CONCLUSION

We introduced ReTranDrive, a reinforced transition-aware framework that shifts the planning objective from single-step planning quality to sequential execution coherence. Rather than scoring a trajectory from the current observation alone, ReTranDrive defines rewards over consecutive plan pairs, realized in a VLM-conditioned diffusion planner both as a differentiable coherence term during imitation and as a simulator-verifiable reward during reinforcement, with an inconsistency charged to the decision that produced it. A central finding is that coherence across replanning cycles is best optimized over transitions rather than enforced as unconditional similarity between plans, which would only suppress the maneuvers that safety sometimes demands. Extensive experiments on NAVSIM show that ReTranDrive achieves state-of-the-art performance and reaches human-level driving quality. We hope this work encourages planning objectives that span consecutive decisions rather than per-decision objectives.

## AI USE STATEMENT

Generative AI tools were used solely as writing assistants to improve the clarity, grammar, and readability of this manuscript. They were not used for research ideation, methodology design, experimental implementation, data analysis, or the interpretation of results. All AI-assisted text was reviewed and verified by the authors, who take responsibility for the final content.

## REFERENCES

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. $\pi_0$: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. VADv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024.

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.

Lan Feng, Yang Gao, Eloi Zablocki, Quanyi Li, Wuyang Li, Sichao Liu, Matthieu Cord, and Alexandre Alahi. Rap: 3d rasterization augmented end-to-end planning. In *International Conference on Learning Representations*, volume 2026, pp. 37722–37737, 2026.

Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24823–24834. IEEE, 2025.

Hao Gao, Shaoyu Chen, Bo Jiang, Bencheng Liao, Yiang Shi, Xiaoyang Guo, Yuechuan Pu, Xiangyu Li, Wenyu Liu, Qian Zhang, et al. RAD: Training an end-to-end driving policy via large-scale 3DGS-based reinforcement learning. *Advances in Neural Information Processing Systems*, 38:32551–32576, 2026.

Ke Guo, Haochen Liu, Xiaojun Wu, Jia Pan, and Chen Lv. ipad: Iterative proposal-centric end-to-end autonomous driving. *IEEE Robotics and Automation Letters*, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

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.

Minqing Huang, Yujiao Xiang, Zihan Liang, Jiajie Huang, Jingqi Wang, Zhi Xu, Feiyang Tan, Hangning Zhou, Mu Yang, and Gong Che. Coworld-vla: Thinking in a multi-expert world model for autonomous driving. *arXiv preprint arXiv:2605.10426*, 2026.

Zhiyu Huang, Xinshuo Weng, Maximilian Igl, Yuxiao Chen, Yulong Cao, Boris Ivanovic, Marco Pavone, and Chen Lv. Gen-drive: Enhancing diffusion generative driving policies with reward modeling and reinforcement learning fine-tuning. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3445–3451. IEEE, 2025.

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

Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. VAD: Vectorized scene representation for efficient autonomous driving. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8306–8316. IEEE, 2023.

Bo Jiang, Shaoyu Chen, Qian Zhang, Wenyu Liu, and Xinggang Wang. AlphaDrive: Unleashing the power of VLMs in autonomous driving via reinforcement learning and reasoning. *arXiv preprint arXiv:2503.07608*, 2025.

Bo Jiang, Shaoyu Chen, Bencheng Liao, Xingyu Zhang, Wei Yin, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. SENNA: Bridging large vision-language models and end-to-end autonomous driving. *International Journal of Computer Vision*, 134(9):407, 2026.

Ellington Kirby, Alexandre Boulch, Yihong Xu, Yuan Yin, Gilles Puy, Éloi Zablocki, Andrei Bursuc, et al. Driving on registers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 32058–32069, 2026.

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*, 2026a.

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*, 2024.

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. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27137–27146, 2025.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, et al. Drivevla-w0: World models amplify data scaling law in autonomous driving. In *International Conference on Learning Representations*, volume 2026, pp. 7890–7911, 2026b.

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.

Lin Liu, Ziying Song, Caiyan Jia, Hangjun Ye, Xiaoshuai Hao, and Long Chen. Driveworld-vla: Unified latent-space world modeling with vision-language-action for autonomous driving. *arXiv preprint arXiv:2602.06521*, 2026.

Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. GPT-driver: Learning to drive with GPT. *arXiv preprint arXiv:2310.01415*, 2023.

Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. DriveLM: Driving with graph visual question answering. In *European Conference on Computer Vision*, pp. 256–274. Springer, 2024.

Chonghao Sima, Kashyap Chitta, Zhiding Yu, Shiyi Lan, Ping Luo, Andreas Geiger, Hongyang Li, and Jose M Alvarez. Centaur: Robust end-to-end autonomous driving with test-time training. *arXiv preprint arXiv:2503.11650*, 2025.

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

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

Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. OmniDrive: A holistic LLM-agent framework for autonomous driving with 3D perception, reasoning and planning. *arXiv preprint arXiv:2405.01533*, 1(2):3, 2024.

Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified vision-language-action model. In *International Conference on Learning Representations*, volume 2026, pp. 80929–80944, 2026b.

Tianze Xia, Yongkang Li, Lijun Zhou, Jingfeng Yao, Kaixin Xiong, Haiyang Sun, Bing Wang, 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.

Chengen Xie, Bin Sun, Tianyu Li, Junjie Wu, Zhihui Hao, XianPeng Lang, and Hongyang Li. Latentvla: Efficient vision-language models for autonomous driving via latent action prediction. *arXiv preprint arXiv:2601.05611*, 2026.

Zebin Xing, Xingyu Zhang, Yang Hu, Bo Jiang, Tong He, Qian Zhang, Xiaoxiao Long, and Wei Yin. Goalflow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1602–1611. IEEE, 2025.

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. *International Conference on Learning Representations*, 2026:157518–157556, 2026.

Pengxuan Yang, Yupeng Zheng, Deheng Qian, Zebin Xing, Qichao Zhang, Linbo Wang, Yichen Zhang, et al. Dreamerad: Efficient reinforcement learning via latent world model for autonomous driving. *arXiv preprint arXiv:2603.24587*, 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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40:11910–11918, 2026.

Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, Xing Wei, and Ning Guo. Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving. In *Advances in Neural Information Processing Systems*, volume 38, pp. 67299–67318, 2025.

Jinqing Zhang, Zehua Fu, Zelin Xu, Wenying Dai, Qingjie Liu, and Yunhong Wang. Resworld: Temporal residual world model for end-to-end autonomous driving. *arXiv preprint arXiv:2602.10884*, 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. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27220–27230, 2025.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

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*, volume 38, pp. 134585–134611. Curran Associates, Inc., 2025. doi: 10.52202/085713-4491.

Yinan Zheng, Ruiming Liang, Kexin Zheng, Jinliang Zheng, Liyuan Mao, Jianxiong Li, Weihao Gu, Rui Ai, Shengbo Li, Xianyuan Zhan, et al. Diffusion-based planning for autonomous driving with flexible guidance. In *International Conference on Learning Representations*, volume 2025, pp. 37207–37227, 2025.

Zewei Zhou, Ruining Yang, Yiluan Guo, Sherry X Chen, Tao Feng, Kateryna Pistunova, Yishan Shen, Lili Su, Jiaqi Ma, et al. Spanvla: Efficient action bridging and learning from negative-recovery samples for vision-language-action model. *arXiv preprint arXiv:2604.19710*, 2026.

Jialv Zou, Shaoyu Chen, Bencheng Liao, Zhiyu Zheng, Yuehao Song, Lefei Zhang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Diffusiondrivev2: Reinforcement learning-constrained truncated diffusion modeling in end-to-end autonomous driving. *arXiv preprint arXiv:2512.07745*, 2025.