Beyond Imitation: Learning Safe End-to-End Autonomous Driving from Hard Negatives

Junli Wang[1,2,5], Zhihua Hua[3], Xueyi Liu[1,5], Zebin Xing[1,5], Wei Zhang[4], Kun Ma[2], Guang Chen[2], Hangjun Ye[2], Long Chen[2], Qichao Zhang[1,5†]

[1] SKL MAIS, Institute of Automation, Chinese Academy of Sciences
[2] Xiaomi Embodied Intelligence Team [3] Fudan University
[4] Harbin Institute of Technology [5] School of Artificial Intelligence, UCAS

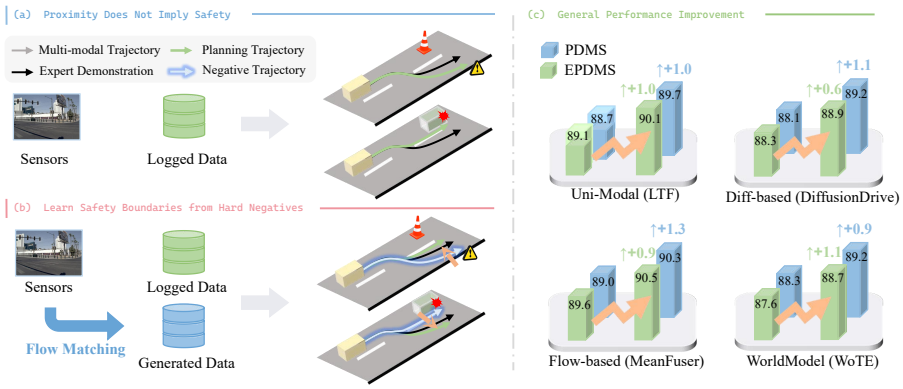

Fig. 1: Learning safe autonomous driving from hard negatives. (a) Uni-modal and multi-modal end-to-end driving models learn by imitating expert demonstrations, yet they lack an understanding of "what constitutes poor imitation." (b) We generate unsafe hard negatives close to expert via flow matching, guiding models to avoid such behaviors. (c) Our approach yields significant performance improvements on reactive (EPDMS) and non-reactive (PDMS) closed-loop benchmarks for multiple methods.

Abstract. Existing imitation learning methods for end-to-end autonomous driving predominantly learn from successful demonstrations by minimizing geometric deviations from expert trajectories. This paradigm implicitly assumes that spatial proximity implies behavioral safety, leading to a critical objective mismatch: trajectories with nearly identical imitation losses may exhibit drastically different safety outcomes, where one remains recoverable while the other results in collision. To address this limitation, we propose **BeyondDrive**, a failure-aware imitation learning framework that jointly learns from successful and failed driving behaviors. First, we introduce a flow matching-based negative trajectory

† Corresponding author. Code link: https://github.com/wjl2244/BeyondDrive.
This work was supported by Beijing Natural Science Foundation-Xiaomi Innovation Joint Fund L253007, by Beijing Nova Program (202604841268) and Beijing Natural Science Foundation under Grant 4242052.

generator that synthesizes safety-critical yet expert-proximate trajectories, enabling explicit modeling of safety asymmetry. Second, we develop a diversity-aware sampling strategy that mitigates mode collapse and improves coverage of diverse failure modes during negative trajectory generation. Third, we propose a Repulsive Distance Loss that simultaneously attracts predictions toward expert demonstrations while repelling them from hard negative trajectories, thereby establishing discriminative safety boundaries in trajectory space. Applied to the uni-modal baseline Latent TransFuser, BeyondDrive achieves 89.7 PDMS on the NAVSIMv1 closed-loop benchmark, outperforming prior state-of-the-art methods. Moreover, BeyondDrive generalizes effectively across different autonomous driving architectures, including multi-modal planners, and further demonstrates strong zero-shot transferability on the HUGSIM benchmark.

Keywords: Autonomous driving · End-to-end planning · Negative sample learning

1 Introduction

End-to-end autonomous driving is commonly formulated as trajectory imitation, where models are trained to minimize geometric deviations from expert demonstrations. Recent years have witnessed remarkable progress in this paradigm, with methods such as TransFuser [33], UniAD [17], and VAD [19] achieving strong planning performance through direct trajectory regression. However, these approaches rely on a fundamental assumption: trajectories that are geometrically close to expert demonstrations are also behaviorally safe.

In practice, this assumption does not always hold. Driving exhibits a critical property that we term *safety asymmetry*: trajectories with nearly identical imitation errors may lead to drastically different safety outcomes. As illustrated in Fig. 1, two trajectories that deviate left or right from an expert demonstration can incur similar geometric losses, while one remains recoverable and the other results in collision. Since conventional imitation learning only optimizes toward successful demonstrations without explicitly modeling unsafe behaviors, the learned policy lacks clear safety boundaries between good and bad imitations.

To alleviate this issue, recent methods have shifted toward multi-modal trajectory generation and trajectory scoring. Methods such as VADv2 [6], DiffusionDrive [24], MeanFuser [41], World4Drive [51], and WoTE [21] generate multiple candidate trajectories using anchor-based designs or generative models [26, 37], followed by rule-based or learned scoring mechanisms for trajectory selection. Although these methods can identify unsafe or low-quality trajectories during inference, the identified failures are ultimately discarded after scoring and never explicitly participate in policy optimization. As a result, existing methods can recognize poor trajectories, but cannot effectively learn from them.

This naturally raises an important question: *Can end-to-end driving policies explicitly learn from expert-proximate failures to establish discriminative safety boundaries?*

Contrastive learning [16] provides a natural perspective for addressing this problem, as its core principle is to simultaneously attract positive samples while repelling negative ones. However, constructing effective negative samples for autonomous driving is highly non-trivial. Useful negative trajectories must satisfy two conflicting properties: they should remain spatially close to expert demonstrations to form meaningful contrasts, while simultaneously containing explicit safety-critical behaviors that provide informative supervision. We refer to such trajectories as *hard negatives*. Generating high-quality hard negatives therefore becomes the key challenge.

Based on these insights, we propose **BeyondDrive**, a failure-aware contrastive learning framework for end-to-end autonomous driving. BeyondDrive enables driving policies to explicitly learn safety boundaries by jointly leveraging successful demonstrations and synthesized hard negatives. Specifically, we first introduce a flow matching-based failure generator that actively synthesizes safety-critical trajectories near the expert manifold. To improve failure diversity and avoid mode collapse, we further incorporate diversity-aware sampling strategies during trajectory generation. Subsequently, we design a two-stage selection mechanism to identify informative negative trajectories that are both unsafe and geometrically close to expert demonstrations. Finally, we propose a Repulsive Distance Loss (RD Loss) that simultaneously pulls predictions toward expert trajectories while pushing them away from synthesized failures, enabling explicit contrastive safety optimization in trajectory space.

We apply BeyondDrive to Latent TransFuser (LTF) [9], a uni-modal baseline on the NAVSIM benchmark [9]. BeyondDrive improves the baseline to 89.7 PDMS on NAVSIM v1, outperforming numerous recent autonomous driving methods. Furthermore, BeyondDrive further demonstrates strong generalizability across diverse multi-modal planning paradigms, including diffusion-based methods such as DiffusionDrive [24], flow-based methods such as MeanFuser [41], and world-model-based approaches such as WoTE [21], consistently yielding significant performance improvements. Additional zero-shot experiments on HUGSIM further demonstrate the robustness and generalizability of our framework.

The contributions of this work are summarized as follows:

- We propose **BeyondDrive**, a failure-aware contrastive learning framework that explicitly learns safety boundaries using synthesized hard negatives.
- We introduce a flow matching-based failure generation strategy together with diversity-aware sampling and contrastive safety optimization, enabling effective construction and utilization of informative negative trajectories.
- Extensive experiments on NAVSIM and HUGSIM demonstrate that BeyondDrive consistently improves both uni-modal and multi-modal autonomous driving architectures, achieving strong closed-loop planning performance and robust generalization.

2 Related Works

In this section, we review end-to-end autonomous driving, the application of generative models in end-to-end autonomous driving, and the use of negative sample learning in end-to-end autonomous driving.

2.1 End-to-End Autonomous Driving

End-to-end autonomous driving [12, 13, 30, 31, 42] has evolved rapidly in recent years. Early methods such as TransFuser [33] introduced multi-modal sensor fusion with direct trajectory regression for interpretable planning, while subsequent frameworks including UniAD [17] and VAD [19] further unified perception, prediction, and planning into differentiable end-to-end architectures. Despite their architectural differences, these methods predominantly follow an imitation-based learning paradigm, where policies are optimized by minimizing geometric deviations from expert demonstrations. As a result, supervision is restricted to successful driving behaviors, and unsafe trajectories are never explicitly modeled during training. To alleviate the limitations of direct regression, recent works have increasingly adopted candidate generation and trajectory scoring paradigms. DiffusionDrive [24] explores diffusion-based trajectory generation, MeanFuser [41] adopts flow-based generative planning, while VADv2 [6] and Hydra-MDP [22] generate trajectory candidates from predefined vocabularies and select optimal trajectories using learned scoring functions. WoTE [21] further improves planning robustness through world-model-based trajectory evaluation and ensemble scoring strategies. Although these methods can identify low-quality or unsafe trajectories during inference, the corresponding failure signals are only used for trajectory ranking and selection. Unsafe trajectories identified by the scoring module are ultimately discarded and do not explicitly participate in policy optimization. Consequently, existing methods can recognize undesirable behaviors, but remain unable to directly learn safety boundaries from them.

2.2 Negative Sample Learning in Autonomous Driving

Negative sample learning has been widely adopted for representation learning and safety-critical understanding. For trajectory prediction, AMD [34] and TrACT [48] enhance rare scenario recognition via decoupled contrastive learning and prototypical clustering, while ECAM [35] and Drive-CLIP [11] incorporate negative signals for collision avoidance and risk assessment. SimScale [39] further studies simulation-based negative data generation but is computationally expensive. DriveDPO [36] and TakeAD [27] extend this idea to preference-based and takeover-based learning, but they operate on pre-collected or predefined trajectory sets, where supervision is limited to ranking or selection over fixed candidates. As a result, unsafe yet expert-proximate trajectories are never explicitly incorporated into policy optimization, preventing the model from learning fine-grained safety boundaries in continuous trajectory space. BeyondDrive addresses this limitation by introducing such expert-proximate negative trajectories into

training and enabling contrastive optimization over the trajectory generation process.

3 Preliminary

In this section, we first introduce flow matching and classifier-free guidance.

3.1 Flow Matching.

Flow Matching (FM) is a simulation-free technique for training Continuous Normalizing Flows (CNFs) by directly regressing vector fields [26]. The core idea is to define a probability path connecting a simple prior distribution p_0 (e.g., standard Gaussian) and the target data distribution p_1, and learn a time-dependent velocity field $v_t : [0, 1] \times \mathbb{R}^d \to \mathbb{R}^d$ that generates this path [1, 26]. Given a coupling of noise samples $\mathbf{x}_0 \sim p_0$ and data samples $\mathbf{x}_1 \sim p_1$, a common choice is the conditional linear interpolation path: $\mathbf{x}_t = (1 - t)\mathbf{x}_0 + t\mathbf{x}_1$, which yields a target velocity $\mathbf{u}_t = \mathbf{x}_1 - \mathbf{x}_0$ [1, 29]. The FM objective minimizes the expected squared error between the predicted velocity $v_t(\mathbf{x}_t)$ and this target:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t\sim\mathcal{U}(0,1),\mathbf{x}_0\sim p_0,\mathbf{x}_1\sim p_1} \left\| v_t(\mathbf{x}_t, t) - (\mathbf{x}_1 - \mathbf{x}_0) \right\|_2 . \tag{1}$$

where $\mathcal{U}(0, 1)$ denotes the uniform distribution in the interval $[0, 1]$. Once trained, samples are generated by solving the Ordinary Differential Equation (ODE) $\mathrm{d}\mathbf{x}_t = v_t(\mathbf{x}_t, t)\mathrm{d}t$ from $t = 0$ to $t = 1$ starting from random noise [5, 26]. Compared to diffusion models, FM offers faster sampling and simpler training objectives [8, 26].

3.2 Classifier-Free Guidance.

Classifier-Free Guidance (CFG) [15] is a technique originally proposed for diffusion models to trade off diversity and fidelity in conditional generation. It jointly trains a conditional and an unconditional model by randomly dropping the condition during training, and extrapolates the score estimate during sampling. Recent works extend CFG to flow matching, forming Guided Flows [8, 14]. Analogous to diffusion CFG, the guided velocity field is computed as:

$$\tilde{v}_t(\mathbf{x}_t, t|\mathcal{C}) = v_t(\mathbf{x}_t, t) + w \cdot \big(v_t(\mathbf{x}_t, t|\mathcal{C}) - v_t(\mathbf{x}_t, t)\big), \tag{2}$$

where $v_t(\mathbf{x}_t, t|\mathcal{C})$ and $v_t(\mathbf{x}_t, t)$ are conditional and unconditional velocity estimators, respectively. The guidance scale $w \geq 0$ balances conditional fidelity and sample diversity: $w = 1$ gives standard conditional generation, $w > 1$ amplifies the condition for higher fidelity, and $w < 1$ increases diversity. This flexibility makes CFG particularly suitable for generating diverse trajectory candidates that remain close to expert demonstrations, a key requirement for constructing informative negative samples in our framework.

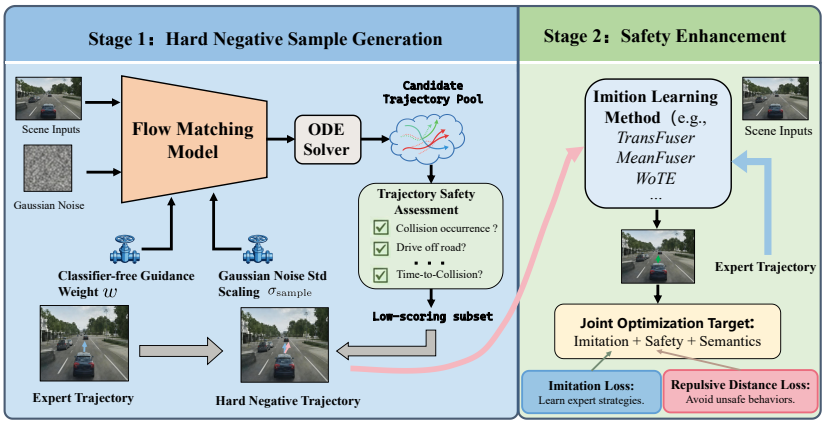

Fig. 2: Overview of the BeyondDrive framework. In Stage 1, classifier-free guidance and noise standard deviation scaling are employed to generate diverse trajectory candidates near expert demonstrations, followed by safety-aware and distance-aware filtering to construct hard negative samples that are unsafe yet expert-proximate. In Stage 2, the imitation learning loss and repulsive distance loss jointly optimize the policy to learn expert strategies while avoiding unsafe behaviors.

4 Method

In this section, we briefly revisit the end-to-end autonomous driving task and introduce our proposed framework, **BeyondDrive**, which enhances the safety of imitation learning. As shown in Fig. 2, BeyondDrive comprises two key stages: first, we generate diverse unsafe yet expert-proximate negative samples using a flow matching-based generator with a diversity enhancement strategy that combines CFG and noise std scaling to prevent mode collapse; second, we employ a RD loss to pull predictions toward expert demonstrations while pushing them away from hard negative, encouraging better separation between safe and unsafe behaviors through contrastive learning.

4.1 Problem Formulation

End-to-end autonomous driving systems take raw sensor data as input and directly plan a future trajectory. At each time step t, the ego vehicle observes multi-modal sensory data: a front-view RGB image $\mathbf{I}_t \in \mathbb{R}^{H\times W\times 3}$, a LiDAR point cloud $\mathbf{P}_t \in \mathbb{R}^{N\times 3}$ (with N points), and ego-vehicle states $\mathbf{s}_t = [v_t, a_t]$ consisting of velocity and acceleration. Additionally, a high-level navigation command c_t (e.g., go straight, turn left, turn right) is provided by a global planner. The model is required to output a planned trajectory $\hat{\tau}_t = \{(x_{t+k\Delta t}, y_{t+k\Delta t}, h_{t+k\Delta t})\}_{k=1}^{K}$ covering a 4-second horizon at a 2 Hz control frequency, where $K = 8$ and $\Delta t = 0.5\,\text{s}$. Each waypoint is represented in the ego-vehicle coordinate frame with position (x, y) and heading angle h.

4.2 Guided Trajectory Generation via Flow Matching

To generate high-quality negative samples, we first train a conditional flow matching model for trajectory generation. Let $\mathbf{x}_0 \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I})$ be a noise sample and let $\mathbf{x}_1 = \mathrm{vec}(\boldsymbol{\tau}_{\mathrm{exp}})$ be the vectorized representation of an expert demonstration $\boldsymbol{\tau}_{\mathrm{exp}}$ from the dataset. The context features $\mathcal{C} = \mathcal{E}_\theta(\mathbf{I}_t, s_t, c_t)$ are extracted by a scene encoder $\mathcal{E}_\theta$ [33] from the front-view image $\mathbf{I}_t$, ego states s_t, and navigation command c_t. We adopt the conditional linear interpolation $\mathbf{x}_t = (1-t)\mathbf{x}_0 + t\mathbf{x}_1$ with the corresponding target velocity $\mathbf{u}_t = \mathbf{x}_1 - \mathbf{x}_0$.

The model learns a time-dependent velocity field $v_\theta(\mathbf{x}_t, t|\mathcal{C}_{\mathrm{in}})$ conditioned on context features. During training, we randomly drop the condition with probability $p = 0.1$ to enable classifier-free guidance: $\mathcal{C}_{\mathrm{in}} = \mathcal{C}$ with probability $1-p$, and $\mathcal{C}_{\mathrm{in}} = \emptyset$ otherwise. We train the network v_θ with an ℓ_1 loss instead of the conventional mean squared error:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{t\sim\mathcal{U}(0,1),\mathbf{x}_0,\mathbf{x}_1} \left\| v_\theta(\mathbf{x}_t, t|\mathcal{C}_{\mathrm{in}}) - (\mathbf{x}_1 - \mathbf{x}_0) \right\|_1, \tag{3}$$

which yields better generation fidelity in planning. The network architecture v_θ consists of a Transformer Decoder, LayerNorm, and a Feed-Forward Network.

To achieve broader coverage of the high-quality hard negative space, we control the standard deviation by setting the prior noise scale to σ during training and increasing the sampling range to $\sigma_{sample} = 2.0\sigma$ during inference. At inference, we employ classifier-free guidance to steer the generation toward regions of high conditional likelihood. The guided velocity field is computed as:

$$\tilde{v}_\theta(\mathbf{x}_t, t|\mathcal{C}_{\mathrm{in}}) = v_\theta(\mathbf{x}_t, t) + w\big(v_\theta(\mathbf{x}_t, t|\mathcal{C}_{\mathrm{in}}) - v_\theta(\mathbf{x}_t, t)\big), \tag{4}$$

where $v_\theta(\mathbf{x}_t, t)$ is the unconditional velocity estimate obtained by dropping the condition, and $w \geq 0$ is the guidance scale. For each conditioning input $\mathcal{C}_{\mathrm{in}}$, we sample 64 candidate trajectories $\{\boldsymbol{\tau}_i\}_{i=1}^{64}$ in parallel by solving the ODE $\mathrm{d}\mathbf{x}_t = \tilde{v}_\theta(\mathbf{x}_t, t|\mathcal{C}_{\mathrm{in}})\mathrm{d}t$ from $t = 0$ to $t = 1$ and reshaping the resulting $\mathbf{x}_1$ back to trajectory form. We then assess each candidate using a scoring function $S(\boldsymbol{\tau})$ (*e.g.*, PDMS) that measures safety and driving quality. To construct informative negative samples, we first select a subset of trajectories with the lowest $S(\boldsymbol{\tau})$:

$$\mathcal{U}_{\mathrm{low}} = \{\boldsymbol{\tau} \in \{\boldsymbol{\tau}_i\} \mid S(\boldsymbol{\tau}) < \xi\}, \tag{5}$$

where ξ is a threshold that retains the most unsafe candidates. From this set, we pick the trajectory that is spatially closest to the expert demonstration $\boldsymbol{\tau}^{\mathrm{exp}}$:

$$\boldsymbol{\tau}^{\mathrm{neg}} = \underset{\boldsymbol{\tau}\in\mathcal{U}_{\mathrm{low}}}{\arg\min} \left\| \boldsymbol{\tau} - \boldsymbol{\tau}^{\mathrm{exp}} \right\|_2, \tag{6}$$

where $\|\cdot\|_2$ denotes the Euclidean distance summed over all waypoints. This two-stage selection ensures that the negative samples are both unsafe and semantically similar to expert demonstrations, providing challenging contrasts for the subsequent policy learning stage. The training algorithm follows the procedure outlined in **??**, while the hard negative sampling process is described in **??**.

4.3 Negative Sample Learning

We apply our BeyondDrive framework to the baseline TransFuser [33] model, obtaining an enhanced version denoted as TransFuser v7. TransFuser takes as input the front-view image $\mathbf{I}_t$, LiDAR point cloud $\mathbf{P}_t$, ego state $\mathbf{s}_t$, and navigation command c_t. It employs a ResNet-34 backbone to extract context features $\mathcal{C}$ and is trained with an auxiliary map reconstruction loss $\mathcal{L}_{\text{map}}$ [9] on rasterized HD maps, which facilitates learning better scene representations.

To smooth the data distribution, we normalize using the first-order difference of trajectory coordinates, denoted as $\Delta\boldsymbol{\tau} = \{(\Delta x_k, \Delta y_k, \sin h_k, \cos h_k)\}_{k=1}^{K}$. Here, Δx_k and Δy_k are the displacements over the k-th time interval (equivalent to average velocity scaled by $\Delta t = 0.5\,\text{s}$), while $\sin h_k$ and $\cos h_k$ encode the heading angle. This parameterization avoids the discontinuity inherent in direct angle regression, resulting in a smoother learning target that facilitates network convergence. The trajectory loss is then defined as the ℓ_1 distance between the predicted and ground-truth differentials:

$$\mathcal{L}_{\text{imi}} = \|\Delta\hat{\boldsymbol{\tau}} - \Delta\boldsymbol{\tau}^{\text{exp}}\|_1. \tag{7}$$

To inject safety awareness, we introduce a RD loss that encourages the predicted trajectory to stay away from unsafe yet spatially similar negative samples generated by our flow matching module. Given a negative trajectory $\boldsymbol{\tau}_{\text{neg}} = \{(x_k^{\text{neg}}, y_k^{\text{neg}}, h_k^{\text{neg}})\}_{k=1}^{K}$ associated with the same condition, we compute its differential counterpart $\Delta\boldsymbol{\tau}^{\text{neg}}$ in the same manner. The RD loss is then defined as the negative clipped ℓ_1 distance between the predicted and the hard negative differentials, where the distance is upper-bounded by a constant C :

$$\mathcal{L}_{\text{rd}} = -\min(\|\Delta\hat{\boldsymbol{\tau}} - \Delta\boldsymbol{\tau}^{\text{neg}}\|_1, C), \tag{8}$$

this formulation pushes the prediction away from unsafe regions in the trajectory space, while the trajectory imitation loss $\mathcal{L}_{\text{imi}}$ simultaneously pulls it toward the expert demonstration, thereby improving the separability between safe and unsafe trajectories through contrastive optimization. The overall training objective combines all components:

$$\mathcal{L} = \lambda_{\text{imi}}\mathcal{L}_{\text{imi}} + \lambda_{\text{rd}}\mathcal{L}_{\text{rd}} + \lambda_{\text{sem}}\mathcal{L}_{\text{sem}}, \tag{9}$$

where λ_{imi}, λ_{rd}, and λ_{sem} are weighting coefficients balancing the contributions of each term.

5 Experiments

In this section, we conduct extensive experiments to validate the effectiveness of our proposed method. We first introduce the evaluation benchmark and dataset used in our experiments. Then, through ablation studies and qualitative visualizations, we analyze the contribution of each key component in our framework. Finally, we demonstrate the generality of our approach by integrating it with other advanced driving models, showing consistent performance improvements.

5.1 Benchmark

NAVSIM v1. [9] This is a data-driven non-reactive simulation benchmark that bridges the gap between open-loop and closed-loop evaluation. NAVSIM unrolls bird's-eye-view abstractions of real-world scenes over a short simulation horizon to compute simulation-based metrics such as progress and time-to-collision, offering better alignment with closed-loop performance than traditional displacement errors while maintaining computational efficiency. The benchmark is built upon the OpenScene dataset (a redistribution of nuPlan [3]) and provides standardized filtered splits to focus on challenging driving scenarios. The dataset is split into two subsets: **navtrain** with approximately 85k scenes and **navtest** with around 12k scenes. The evaluation metric, PDM Score (**PMDS**), is a weighted combination of five sub-metrics: no collisions (NC), drivable area compliance (DAC), time-to-collision (TTC), comfort (Comf.), and Ego Progress (EP). The formal computation is defined as:

$$\text{PDMS} = \text{NC} \times \text{DAC} \times \frac{5 \cdot \text{EP} + 5 \cdot \text{TTC} + 2 \cdot \text{Comf.}}{12}, \tag{10}$$

NAVSIM v2. [4] Building upon v1, this version introduces a reactive simulation environment where background traffic agents dynamically respond to the ego vehicle's behavior, enabling more realistic evaluation of interactive driving scenarios . To comprehensively assess performance in this reactive setting, NAVSIM v2 employs the Extended PDM Score (**EPDMS**), which expands the metric set from five to nine sub-metrics . The multiplier metrics are extended with Driving Direction Compliance (DDC) and Traffic Light Compliance (TLC) , while the weighted metrics are augmented with Lane Keeping (LK) , History Comfort (HC) (an improved version of the basic Comfort metric), and Extended Comfort (EC).

HUGSIM. [52] HUGSIM is a real-time, photo-realistic closed-loop simulator that lifts captured 2D RGB images into 3D space via 3D Gaussian Splatting, enabling dynamic updates of ego and actor states based on control commands. Unlike log-replay or reactive-agent benchmarks, HUGSIM supports fully interactive closed-loop evaluation: the ego agent's decisions influence the environment in real time, and surrounding vehicles are simulated reactively. **HD-Score** is used for closed-loop evaluation, It aggregates Route Completion (RC) with NC, DAC, TTC, Comf. across an episode of length T:

$$\text{HD} - \text{Score} = \text{RC} \times \frac{1}{T} \sum_{t=1}^{T} \left(\text{NC}_t \times \text{DAC}_t \times \frac{5 \cdot \text{TTC}_t + 2 \cdot \text{Comf.}_t}{7} \right). \tag{11}$$

5.2 Implementation Details

We trained all models with a batch size of 32 for a total of 100 epochs. We employed a cosine annealing learning rate scheduler with three warmup epochs and a peak learning rate of 2e-4. The Flow Matching model uses the exact

Table 1: Performance on the NAVSIMv1 navtest Benchmark. "C" dontes Camera, and "L" denotes Lidar. * Indicates the optimized version.

Method	Venue	Input	NC↑	DAC↑	EP↑	TTC↑	Comf.↑	PDMS↑
Hydra-MDP [22]	arXiv'24	C&L	98.3	96.0	78.7	94.6	**100.0**	86.5
DiffusionDrive [24]	CVPR'25	C&L	98.2	96.2	82.2	94.7	**100.0**	88.1
WoTE [21]	ICCV'25	C&L	98.5	96.8	81.9	94.4	99.9	88.3
Hydra-NeXt [23]	ICCV'25	C&L	98.1	97.7	81.8	94.6	**100.0**	88.6
TransFuser [33]	CVPR'21	C&L	97.7	92.8	79.2	92.8	**100.0**	84.0
TransFuser v7	-	C&L	98.7	97.5	83.2	95.5	**100.0**	89.6
UniAD [17]	CVPR'23	C	97.8	91.9	78.8	92.9	**100.0**	83.4
PARA-Drive [43]	CVPR'24	C	97.9	92.4	79.3	93.0	99.8	84.0
LAW [20]	ICLR'25	C	96.4	95.4	81.7	88.7	99.9	84.6
Epona [49]	ICCV'25	C	97.9	95.1	80.4	93.8	99.9	86.2
DrivingGPT [7]	ICCV'25	C	98.9	90.7	79.7	94.9	95.6	82.4
PWM [50]	NeurIPS'25	C	98.6	95.9	81.8	95.4	**100.0**	88.1
AutoVLA [53]	NeurIPS'25	C	98.4	95.6	81.9	98.0	99.9	89.1
WorldRFT [46]	AAAI'26	C	97.8	96.8	81.7	94.0	**100.0**	87.8
VADv2 [6]	ICLR'26	C	98.3	97.4	82.3	95.7	**100.0**	89.3
BridgeDrive [28]	ICLR'26	C	98.2	96.1	82.3	94.5	**100.0**	88.0
SNG-VLA [18]	ICRA'26	C	98.9	96.5	**83.8**	92.9	**100.0**	88.2
VGGDrive [40]	CVPR'26	C	98.6	96.3	82.9	95.6	99.9	88.8
DriveLaW [44]	CVPR'26	C	**99.0**	97.1	81.3	**96.7**	**100.0**	89.1
MeanFuser [41]	CVPR'26	C	98.6	97.0	82.8	95.0	**100.0**	89.0
LTF [33]	NeurIPS'24	C	97.4	92.8	79.0	92.4	**100.0**	83.8
LTFv6 [32]	CVPR'26	C	97.5	95.4	80.9	93.8	**100.0**	86.4
LTF*	-	C	98.2	97.0	83.0	94.6	**100.0**	88.7
LTFv7	-	C	98.4	**97.9**	83.7	95.0	**100.0**	**89.7**

same perception module as LTF [9]. For negative sample generation, we used a classifier-free guidance (CFG) scale of 0.5 and 5 sampling steps in the flow matching process.

5.3 Main Results

Results on NAVSIM. We train two variants of our model under the same setting: **TransFuser v7**, which takes both camera and point cloud inputs, and **LTFv7**, which uses camera inputs only. As shown in Tab. 1, our method achieves promising performance in both settings, with PDMS scores of 89.6 and 89.7, respectively, surpassing many recent high-performing multimodal evaluators [6, 22–24, 41], world models [20, 21, 44, 49, 50], and Vision-Language-Action (VLA) based [7, 40, 53] approaches. The comparable performance between TransFuser v7 and LTFv7 is consistent with the official NAVSIM benchmark results. Moreover, compared to LTF, LTFv7 improves PDMS by **5.9**, with NC and DAC increasing by **1.0** and **5.1**, respectively. These gains indicate a significant reduction in collisions and drivable area departures, while also enhancing traffic efficiency (EP

Table 2: Performence on the NAVSIMv2 navtest Benchmark.

Method	NC↑	DAC↑	DDC↑	TLC↑	EP↑	TTC↑	LK↑	HC↑	EC↑	EPDMS↑
DriveSuprim [47]	97.5	96.5	99.4	99.6	**88.4**	96.6	95.5	**98.3**	77.0	83.1
ARTEMIS [10]	98.3	95.1	98.6	**99.8**	81.5	97.4	96.5	-	**98.3**	83.1
DiffusionDrive [24]	98.2	96.3	99.4	**99.8**	87.4	97.4	97.0	**98.3**	87.7	88.3
MeanFuser [41]	98.3	97.4	**99.6**	**99.8**	87.5	97.6	**97.5**	**98.3**	88.0	89.6
TransFuser [33]	97.7	92.8	99.2	99.7	87.5	96.5	95.9	**98.3**	88.3	84.4
TransFuser v7	98.3	**98.0**	99.5	**99.8**	87.3	97.5	97.3	**98.3**	88.3	90.0
LTF [9]	97.6	91.9	99.2	**99.8**	87.6	97.2	96.6	**98.3**	86.3	83.6
LTF*	98.3	97.0	99.5	**99.8**	87.5	97.3	97.3	**98.3**	88.2	89.1
LTFv7	**98.4**	97.9	99.5	**99.8**	87.8	**98.0**	97.3	**98.3**	**88.5**	**90.1**

Table 3: Zero-shot Performance on the HUGSIM Benchmark.

Method	Easy		Medium		Hard		Extreme		Overall	
	RC	HD-Score	RC	HD-Score	RC	HD-Score	RC	HD-Score	RC	HD-Score
UniAD [17]	58.6	48.7	41.2	29.5	**40.4**	**27.3**	26.0	14.3	40.6	28.9
VAD [19]	38.7	24.3	27.0	9.9	25.5	10.4	23.0	8.2	27.9	12.3
MeanFuser [41]	76.7	**67.1**	41.1	24.3	37.9	21.5	28.4	11.0	27.9	12.3
LTF [9]	68.4	52.8	40.7	24.6	36.9	19.8	25.5	8.1	41.4	24.8
LTFv7	**76.8**	65.6	**43.0**	**31.4**	35.5	26.3	**29.6**	**16.2**	**46.2**	**34.8**

+**4.7**) and maintaining a safer distance (TTC +**2.6**). Table 2 reports results on the reactive closed-loop benchmark NAVSIM v2, where both TransFuser v7 and LTFv7 exceed 90 EPDMS. Similarly, substantial gains are observed in the DAC metric, and the improved LK (+**0.7**) metric demonstrates that our approach enhances long-range planning capability.

Results on HUGSIM. As shown in Tab. 3, we evaluate the zero-shot closed-loop performance of our method on four official datasets (KITTI-360 [25], Waymo [38], nuScenes [2], and Pandaset [45]). Our approach achieves state-of-the-art results in the levels of difficulty of easy, Medium, and extreme. Compared to LTF, it improves the overall Rc and HD-Score by +**4.8** and +**10**, respectively, demonstrating improved driving performance and safety while exhibiting a strong generalizability. Notably, consistent improvements are also observed on metrics such as DDC and LK within EPDMS, which are not present in the PDMS metric used for negative sample selection. This further confirms that our method learns a transferable notion of safety beyond mere optimization of a single simulator metric, and that the performance gains are not contingent on the specific scoring function used during training.

5.4 Ablation Studies and Case Studies

We ablate the key components and hyperparameters to investigate their impact on model performance.

Table 4: Ablation studies on core components for both uni-modal and multi-modal methods. Hyperparameter Tuning includes replacing the constant scheduler with a cosine annealing scheduler with warmup, as well as adjusting the loss weights. Normalization is performed by using first-order differences of coordinates instead of raw coordinates.

Model	Components	NC↑	DAC↑	EP↑	TTC↑	Comf.↑	PDMS↑	EPDMS↑
Uni-Modal Planner								
Tranfsuer [33]	-	97.7	92.8	79.2	92.8	100	84.0	84.4
LTF [9]	Baseline	97.4	92.8	79.0	92.4	100	83.8	83.6
$\mathcal{M}_{s1}$	LTF + Hyperparameter Tuning	98.1	95.7	81.7	94.3	100	87.4	87.8
LTF*	$\mathcal{M}_{s1}$ + Trajectory Normalization	98.2	97.0	83.0	94.6	100	88.7	89.1
LTFv7	LTF* + BeyondDrive	98.4	97.9	83.7	95.0	100	$89.7_{+1.0}$	$90.1_{+1.0}$
Diffusion-based Planner								
DiffusionDrive [24]	Baseline	98.2	96.2	82.2	94.7	100	88.1	88.3
$\mathcal{M}_{m1}$	DiffusionDrive + BeyondDrive	98.2	97.4	83.4	94.8	100	$89.2_{+1.1}$	$88.9_{+0.6}$
Flow-based Planner								
MeanFuser [41]	Baseline	98.6	97.0	82.8	95.0	100	89.0	89.6
$\mathcal{M}_{m2}$	MeanFuser + BeyondDrive	98.5	98.4	84.2	95.1	100	$90.3_{+1.3}$	$90.5_{+0.9}$
World Model Planner								
WoTE [21]	Baseline	98.5	96.8	81.9	94.4	99.9	88.3	87.6
$\mathcal{M}_{w1}$	WoTE + BeyondDrive	98.4	97.3	83.1	95.1	100	$89.2_{+0.9}$	$88.7_{+1.1}$

Table 5: Ablation on key hyperparameters. N: number of flow matching samples per scene. w: CFG scale. σ_{sample}: noise scale at sampling. $\text{Prop}_{\text{sample}}$: proportion of scenes with valid negative samples. PDMS_{max} and PDMS_{std}: average (over scenes) of the maximum and standard deviation of PDMS across N trajectories. $\text{Dist}_{\text{nega}}$ and $\text{PDMS}_{\text{nega}}$: average distance with expert demonstration and average PDMS of the selected negatives. PDMS_{LTF}: PDMS of the LTF baseline.

N	w	σ_{sample}	$\text{Prop}_{\text{sample}}$	PDMS_{max}	PDMS_{std}	$\text{Dist}_{\text{nega}}$	$\text{PDMS}_{\text{nega}}$	PDMS_{LTF}
1	1.0	1.0σ	-	87.1	0	-	-	-
64	1.0	1.0σ	8.0%	94.8	0.0008	0.40	23.9	88.8
64	0.5	1.0σ	78.8%	98.2	0.1900	1.83	22.8	89.1
64	0.5	2.0σ	96.4%	98.9	0.3392	1.68	19.8	89.7

The ablation of each core module. As shown in Tab. 4, we ablate the key components that contribute to the performance improvement of the uni-modal method LTFv7. With hyperparameter tuning alone, *i.e.*, replacing the official constant learning rate scheduler with a cosine annealing scheduler with warmup and reducing the loss weights of auxiliary tasks, the model achieves a performance gain of +3.6 PDMS. After applying normalization, the model further improves by +1.3 PDMS, as the smoothed data distribution eases network learning. Finally, incorporating BeyondDrive boosts the model to 89.7 PDMS, with EPDMS exceeding 90 and clear improvements in safety metrics such as NC, DAC, and TTC, validating the effectiveness of our method. We also applied BeyondDrive to the multimodal methods DiffusionDrive and MeanFuser, as well

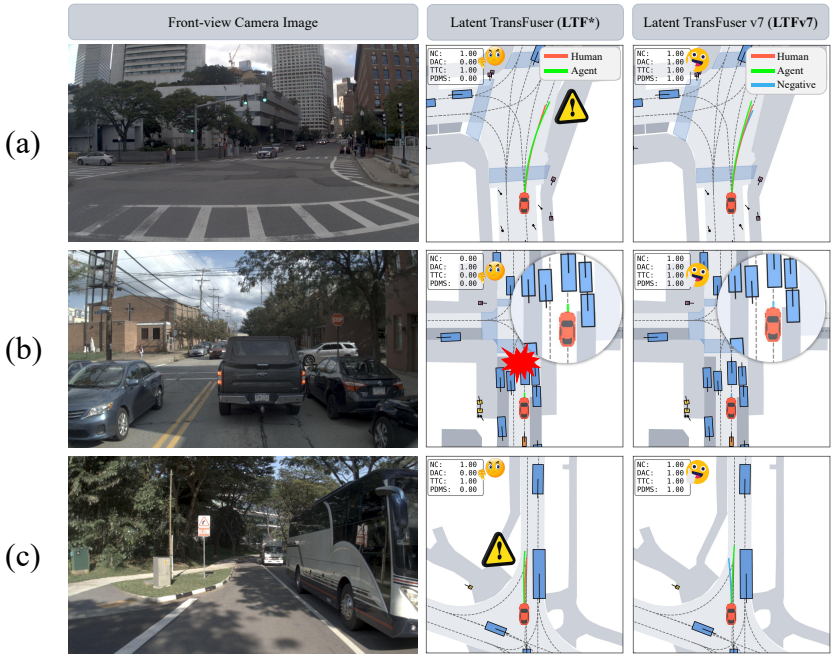

Fig. 3: Case studies on the navtest benchmark. Left: front-view camera images. Middle: BEV scenes with expert demonstrations and LTF* planned trajectories (with scores). Right: expert demonstrations, LTFv7 planned trajectories and negative samples.

as the world model method WoTE. In all three methods, simply by adding one line of training loss code, we achieved PDMS gains of +1.1, +1.3, and +0.9 respectively, demonstrating the generalizability of our approach.

The ablation of key hyperparameters. As shown in Tab. 5, when $N = 1$, the flow matching model achieves a PDMS_{max} of 87.1, reflecting its baseline capability. Without CFG and noise scaling ($w = 1.0$, $\sigma_{\text{sample}} = 1.0\sigma$), only 8.0% of scenes yield valid negative samples. Reducing the CFG scale to $w = 0.5$ substantially increases this proportion to 78.8%. Further enlarging the sampling noise to $\sigma_{\text{sample}} = 2.0\sigma$ pushes the proportion to 96.4%, while the average distance to the expert demonstration ($\text{Dist}_{\text{nega}}$) drops to 1.68, and the LTF performance improves to 89.7 PDMS. These results highlight the importance of both CFG and noise scaling in generating diverse and proximal hard negatives for safety-aware learning.

The ablation of loss weights. We ablate the influence of the RD loss weight λ_{rd} on model performance. As shown in Fig. 4, gradually increasing λ_{rd} yields significant performance improvements, provided that its value remains lower than λ_{imi}; otherwise, the model fails to converge. More details are presented in **??**.

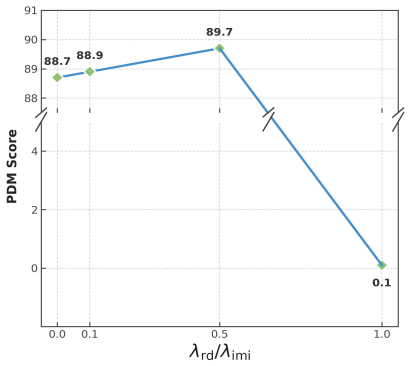

Fig. 4: Impact of $\lambda_{rd}/\lambda_{imi}$ on model performance.

Case Studies. Specifically, we present three representative scenarios in Fig. 3 to demonstrate the improvements of LTFv7 over LTF* in terms of planning quality and safety. (a) At a T-shaped intersection, the trajectory planned by LTF* deviates from the drivable area at the far end, resulting in a DAC score of 0. Our generated negative samples indicate that such maneuvers are unsafe, which guides the optimized LTFv7 to learn a safer driving strategy that steers clear of the road boundary. (b) At a crossroad requiring a stop, LTF* plans a forward-moving trajectory, leading to NC and TTC scores of 0, indicating an imminent collision with the front vehicle. Our negative samples demonstrate the danger of proceeding, and thus LTFv7 maintains a stationary policy. (c) In a bus-merging scenario, LTF* adopts an overly cautious yielding strategy that steers off the road edge. Corrected by the negative samples, the model learns the proper planning policy of staying near the lane center.

6 Conclusion

We presented BeyondDrive, a failure-aware contrastive learning framework for end-to-end autonomous driving that enforces safety boundaries via expert-proximate negative trajectories. By synthesizing unsafe yet expert-proximate trajectories with flow matching and optimizing a contrastive distance-based objective, BeyondDrive enables policies to preserve imitation fidelity while explicitly avoiding safety-critical behaviors. Experiments on NAVSIM and HUGSIM show consistent improvements across uni-modal and multi-modal planners, demonstrating strong generalization and improved closed-loop driving performance.

7 Limitations

BeyondDrive focuses on trajectory-level safety learning and does not explicitly model long-horizon interactions or high-level semantic constraints, both of which are deferred to future work. Additionally, its integration with vocabulary-based approaches such as VADv2 is limited by the need for an additional refinement module following the scoring phase to facilitate safety-oriented training.

References

1. Albergo, M.S., Vanden-Eijnden, E.: Building normalizing flows with stochastic interpolants. In: The Eleventh International Conference on Learning Representations, ICLR 2023 5
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020) 11
3. Caesar, H., Kabzan, J., Tan, K.S., Fong, W.K., Wolff, E., Lang, A., Fletcher, L., Beijbom, O., Omari, S.: nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. arXiv preprint arXiv:2106.11810 (2021) 9
4. Cao, W., Hallgarten, M., Li, T., Dauner, D., Gu, X., Wang, C., Miron, Y., Aiello, M., Li, H., Gilitschenski, I., et al.: Pseudo-simulation for autonomous driving. In: 9th Annual Conference on Robot Learning 9
5. Chen, R.T., Rubanova, Y., Bettencourt, J., Duvenaud, D.K.: Neural ordinary differential equations. Advances in neural information processing systems **31** (2018) 5
6. Chen, S., Jiang, B., Gao, H., Liao, B., Xu, Q., Zhang, Q., Huang, C., Liu, W., Wang, X.: Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. arXiv preprint arXiv:2402.13243 (2024) 2, 4, 10
7. Chen, Y., Wang, Y., Zhang, Z.: 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) 10
8. Dao, Q., Phung, H., Nguyen, B., Tran, A.: Flow matching in latent space. arXiv preprint arXiv:2307.08698 (2023) 5
9. Dauner, D., Hallgarten, M., Li, T., Weng, X., Huang, Z., Yang, Z., Li, H., Gilitschenski, I., Ivanovic, B., Pavone, M., et al.: Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. Advances in Neural Information Processing Systems **37**, 28706–28719 (2024) 3, 8, 9, 10, 11, 12
10. Feng, R., Xi, N., Chu, D., Wang, R., Deng, Z., Wang, A., Lu, L., Wang, J., Huang, Y.: Artemis: Autoregressive end-to-end trajectory planning with mixture of experts for autonomous driving. IEEE Robotics and Automation Letters **11**(1), 226–233 (2025) 11
11. Gan, W., Dao, M.S., Zettsu, K.: Drive-clip: Cross-modal contrastive safety-critical driving scenario representation learning and zero-shot driving risk analysis. In: International Conference on Multimedia Modeling. pp. 82–97. Springer (2024) 4
12. Gao, Y., Liu, D., Zhang, Q., Zheng, Y., Tian, H., Li, G., Ye, H., Chen, L., Ding, D.W., Zhao, D.: Learning from mistakes: Post-training for driving vla with takeover data. arXiv preprint arXiv:2603.14972 (2026) 4
13. Gao, Y., Zhang, Q., Liu, D., Xia, Z., Li, G., Ma, K., Chen, G., Ye, H., Chen, L., Ding, D.W., et al.: Perlad: Towards enhanced closed-loop end-to-end autonomous driving with pseudo-simulation-based reinforcement learning. IEEE Robotics and Automation Letters (2026) 4
14. Geng, Z., Deng, M., Bai, X., Kolter, Z., He, K.: Mean flows for one-step generative modeling. Advances in Neural Information Processing Systems **38**, 75460–75482 (2026) 5
15. Ho, J., Salimans, T.: Classifier-free diffusion guidance. In: NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications 5

16. Hu, H., Wang, X., Zhang, Y., Chen, Q., Guan, Q.: A comprehensive survey on contrastive learning. Neurocomputing **610**, 128645 (2024) 3
17. Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al.: Planning-oriented autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 17853–17862 (2023) 2, 4, 10, 11
18. Hua, Z., Wang, J., Li, P., Jin, Q., Zhang, B., Sheng, K., Chen, Y., Gan, Z., Ding, W.: Unveiling the surprising efficacy of navigation understanding in end-to-end autonomous driving. arXiv preprint arXiv:2604.12208 (2026) 10
19. Jiang, B., Chen, S., Xu, Q., Liao, B., Chen, J., Zhou, H., Zhang, Q., Liu, W., Huang, C., Wang, X.: Vad: Vectorized scene representation for efficient autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8340–8350 (2023) 2, 4, 11
20. Li, Y., Fan, L., He, J., Wang, Y., Chen, Y., Zhang, Z., Tan, T.: Enhancing end-to-end autonomous driving with latent world model. In: The Thirteenth International Conference on Learning Representations 10
21. Li, Y., Wang, Y., Liu, Y., He, J., Fan, L., Zhang, Z.: End-to-end driving with online trajectory evaluation via bev world model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27137–27146 (2025) 2, 3, 4, 10, 12
22. Li, Z., Li, K., Wang, S., Lan, S., Yu, Z., Ji, Y., Li, Z., Zhu, Z., Kautz, J., Wu, Z., et al.: Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. arXiv preprint arXiv:2406.06978 (2024) 4, 10
23. Li, Z., Wang, S., Lan, S., Yu, Z., Wu, Z., Alvarez, J.M.: Hydra-next: Robust closed-loop driving with open-loop training. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27305–27314 (2025) 10
24. Liao, B., Chen, S., Yin, H., Jiang, B., Wang, C., Yan, S., Zhang, X., Li, X., Zhang, Y., Zhang, Q., et al.: Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 12037–12047 (2025) 2, 3, 4, 10, 11, 12
25. Liao, Y., Xie, J., Geiger, A.: Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(3), 3292–3310 (2022) 11
26. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. In: The Eleventh International Conference on Learning Representations, ICLR 2023 (2023) 2, 5
27. Liu, D., Gao, Y., Qian, D., Zhang, Q., Ye, X., Han, J., Zheng, Y., Liu, X., Xia, Z., Ding, D., et al.: Takead: Preference-based post-optimization for end-to-end autonomous driving with expert takeover data. IEEE Robotics and Automation Letters **11**(2), 1738–1745 (2025) 4
28. Liu, S., Chen, W., Li, W., Wang, Z., Yang, L., Huang, J., Zhang, Y., Huang, Z., Cheng, Z., Yang, H.: Bridgedrive: Diffusion bridge policy for closed-loop trajectory planning in autonomous driving. arXiv preprint arXiv:2509.23589 (2025) 10
29. Liu, X., Gong, C., et al.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: The Eleventh International Conference on Learning Representations, ICLR 2023 5
30. Liu, X., Zhong, Z., Zhang, Q., Guo, Y., Zheng, Y., Wang, J., Zhao, D., Liu, Y.F., Su, Z., Gao, Y., et al.: Reasonplan: Unified scene prediction and decision reasoning for closed-loop autonomous driving. In: Conference on Robot Learning. pp. 3051–3068. PMLR (2025) 4

31. Lu, J., Guan, J., Huang, Z., Li, J., Li, G., Kong, L., Li, Y., Wang, H., Xu, S., Luo, Y., et al.: Onevl: One-step latent reasoning and planning with vision-language explanation. arXiv preprint arXiv:2604.18486 (2026) 4
32. Nguyen, L., Fauth, M., Jaeger, B., Dauner, D., Igl, M., Geiger, A., Chitta, K.: Lead: Minimizing learner-expert asymmetry in end-to-end driving. arXiv preprint arXiv:2512.20563 (2025) 10
33. Prakash, A., Chitta, K., Geiger, A.: Multi-modal fusion transformer for end-to-end autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7077–7087 (2021) 2, 4, 7, 8, 10, 11, 12
34. Rao, B., Liao, H., Guan, Y., Wang, C., Wang, B., Zhang, J., Li, Z.: Amd: Adaptive momentum and decoupled contrastive learning framework for robust long-tail trajectory prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 28849–28858 (2025) 4
35. Rosin, G., Rahman, M.R.U., Vascon, S.: Ecam: A contrastive learning approach to avoid environmental collision in trajectory forecasting. In: 2025 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2025) 4
36. Shang, S., Chen, Y., Wang, Y., Li, Y., Zhang, Z.: Drivedpo: Policy learning via safety dpo for end-to-end autonomous driving. arXiv preprint arXiv:2509.17940 (2025) 4
37. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020) 2
38. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 2446–2454 (2020) 11
39. Tian, H., Li, T., Liu, H., Yang, J., Qiu, Y., Li, G., Wang, J., Gao, Y., Zhang, Z., Wang, L., et al.: Simscale: Learning to drive via real-world simulation at scale. arXiv preprint arXiv:2511.23369 (2025) 4
40. Wang, J., Li, G., Huang, Z., Dang, C., Ye, H., Han, Y., Chen, L.: Vggdrive: Empowering vision-language models with cross-view geometric grounding for autonomous driving. arXiv preprint arXiv:2602.20794 (2026) 10
41. Wang, J., Liu, X., Zheng, Y., Xing, Z., Li, P., Li, G., Ma, K., Chen, G., Ye, H., Xia, Z., et al.: Meanfuser: Fast one-step multi-modal trajectory generation and adaptive reconstruction via meanflow for end-to-end autonomous driving. arXiv preprint arXiv:2602.20060 (2026) 2, 3, 4, 10, 11, 12
42. Wang, J., Zhang, Q.: Consistencydrive: Efficient end-to-end autonomous driving with consistency models. In: 2025 10th International Conference on Control, Robotics and Cybernetics (CRC). pp. 57–62. IEEE (2025) 4
43. Weng, X., Ivanovic, B., Wang, Y., Wang, Y., Pavone, M.: Para-drive: Parallelized architecture for real-time autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15449–15458 (2024) 10
44. Xia, T., Li, Y., Zhou, L., Yao, J., Xiong, K., Sun, H., Wang, B., Ma, K., Chen, G., Ye, H., et al.: Drivelaw: Unifying planning and video generation in a latent driving world. arXiv preprint arXiv:2512.23421 (2025) 10
45. Xiao, P., Shao, Z., Hao, S., Zhang, Z., Chai, X., Jiao, J., Li, Z., Wu, J., Sun, K., Jiang, K., et al.: Pandaset: Advanced sensor suite dataset for autonomous driving. In: 2021 IEEE international intelligent transportation systems conference (ITSC). pp. 3095–3101. IEEE (2021) 11

46. Yang, P., Lu, B., Xia, Z., Han, C., Gao, Y., Zhang, T., Zhan, K., Lang, X., Zheng, Y., Zhang, Q.: Worldrft: Latent world model planning with reinforcement fine-tuning for autonomous driving. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 40(14): 11649–11657 (2026) 10
47. Yao, W., Li, Z., Lan, S., Wang, Z., Sun, X., Alvarez, J.M., Wu, Z.: Drivesuprim: Towards precise trajectory selection for end-to-end planning. arXiv preprint arXiv:2506.06659 (2025) 11
48. Zhang, J., Pourkeshavarz, M., Rasouli, A.: Tract: A training dynamics aware contrastive learning framework for long-tail trajectory prediction. In: 2024 IEEE Intelligent Vehicles Symposium (IV). pp. 3282–3288. IEEE (2024) 4
49. Zhang, K., Tang, Z., Hu, X., Pan, X., Guo, X., Liu, Y., Huang, J., Yuan, L., Zhang, Q., Long, X.X., et al.: Epona: Autoregressive diffusion world model for autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27220–27230 (2025) 10
50. Zhao, Z., Fu, T., Wang, Y., Wang, L., Lu, H.: From forecasting to planning: Policy world model for collaborative state-action prediction. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems 10
51. Zheng, Y., Yang, P., Xing, Z., Zhang, Q., et al.: World4drive: End-to-end autonomous driving via intention-aware physical latent world model. In: IEEE/CVF International Conference on Computer Vision. pp. 28632–28642 (2025) 2
52. Zhou, H., Lin, L., Wang, J., Lu, Y., Bai, D., Liu, B., Wang, Y., Geiger, A., Liao, Y.: Hugsim: A real-time, photo-realistic and closed-loop simulator for autonomous driving. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025) 9
53. Zhou, Z., Cai, T., Zhao, S.Z., Zhang, Y., Huang, Z., Zhou, B., Ma, J.: Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems 10