WorldRFT: Latent World Model Planning with Reinforcement Fine-Tuning for Autonomous Driving

Pengxuan Yang[1,2,4*], Ben Lu[4*], Zhongpu Xia[1†], Chao Han[4], Yinfeng Gao[1], Teng Zhang[4], Kun Zhan[4], XianPeng Lang[4], Yupeng Zheng[1], Qichao Zhang[1,3‡]

[1] The State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, CAS
[2] School of Advanced Interdisciplinary Sciences, UCAS
[3] School of Artificial Intelligence, UCAS
[4]Li Auto

Abstract

Latent World Models enhance scene representation through temporal self-supervised learning, presenting a perception annotation-free paradigm for end-to-end autonomous driving. However, the reconstruction-oriented representation learning tangles perception with planning tasks, leading to suboptimal optimization for planning. To address this challenge, we propose WorldRFT, a planning-oriented latent world model framework that aligns scene representation learning with planning via a hierarchical planning decomposition and local-aware interactive refinement mechanism, augmented by reinforcement learning fine-tuning (RFT) to enhance safety-critical policy performance. Specifically, WorldRFT integrates a vision-geometry foundation model to improve 3D spatial awareness, employs hierarchical planning task decomposition to guide representation optimization, and utilizes local-aware iterative refinement to derive a planning-oriented driving policy. Furthermore, we introduce Group Relative Policy Optimization (GRPO), which applies trajectory Gaussianization and collision-aware rewards to fine-tune the driving policy, yielding systematic improvements in safety. WorldRFT achieves state-of-the-art (SOTA) performance on both open-loop nuScenes and closed-loop NavSim benchmarks. On nuScenes, it reduces collision rates by 83% (0.30% → 0.05%). On NavSim, using camera-only sensors input, it attains competitive performance with the LiDAR-based SOTA method DiffusionDrive (87.8 vs. 88.1 PDMS).

Code — https://github.com/pengxuanyang/WorldRFT

Introduction

Traditional end-to-end autonomous driving methods (Chen et al. 2024a; Jia et al. 2025; Jiang et al. 2023; Sun et al. 2024) often rely on multi-task architectures incorporating auxiliary perception modules such as object detection, online mapping, and occupancy prediction to improve scene understanding. Recently, a new paradigm of constructing unified latent world model (WM) representations from raw surround-view images (Zheng et al. 2025; Li, Fan et al. 2025) has enabled slight end-to-end autonomous driving through temporal self-supervised learning.

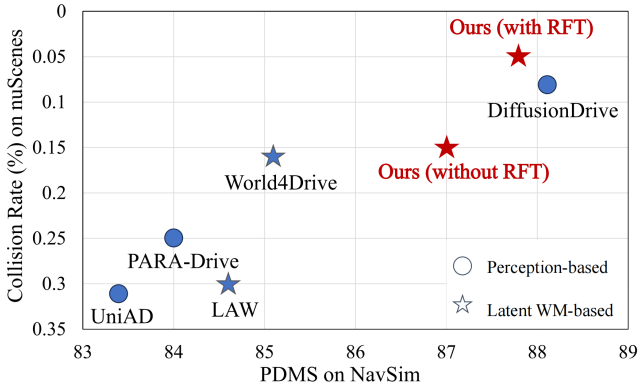

Figure 1: Performance comparison of SOTA methods on open-loop nuScenes and closed-loop NavSim benchmarks.

Despite this progress, existing latent WM suffers from a critical misalignment between reconstruction-oriented representations and planning requirements, manifesting in three key challenges: (1) Lacking spatial awareness: current reconstruction objectives produce generic representations with limited 3D spatial awareness, which is crucial for trajectory planning. Although World4Drive (Zheng et al. 2025) attempts to incorporate explicit 3D information (depth map) through monocular depth estimation models (Hu et al. 2024), this approach suffers from cross-view inconsistencies, limiting the model's global comprehension of complex driving scenarios. (2) Inefficient planning interaction mechanisms: existing models employ a single global planning query to generate trajectories from entire feature maps, inadequately capturing local structures essential for fine-grained planning. This results in dispersed attention and inability to effectively comprehend and extract planning-relevant scene representations, as shown in Figure 5, leading to suboptimal feature utilization for planning decisions. (3) Limited safety awareness: perception-based imitation learning approaches, such as PARA-Drive (Weng et al. 2024), DiffusionDrive(Liao, Chen et al. 2025), focus on minimizing tra-

*These authors contributed equally.
†Corresponding author.
‡Corresponding author.

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

jectory deviations from expert demonstrations without incorporating explicit safety objectives. This paradigm lacks active collision avoidance, merely treating all deviations from expert trajectories equally without understanding underlying safety principles or proactively identifying potential risks in novel scenarios.

To address these challenges, we propose **WorldRFT**, a planning-oriented latent world model framework that aligns representation learning with planning requirements. First, addressing the weak 3D spatial understanding, we design a **Spatial-aware World Encoder (SWE)** that constructs 3D spatially-aware representations by incorporating the visual-geometric foundation model VGGT (Wang et al. 2025), establishing a robust spatial understanding for planning tasks. Second, resolving the inefficiency of planning interaction mechanisms, we propose a **Hierarchical Planning Refinement (HPR)** module that decomposes complex end-to-end planning into three parallel subtasks which describe planning in hierarchical dimension: target region localization, spatial path planning, and temporal trajectory prediction. Building upon this decomposition, we develop a local-aware iterative refinement module that dynamically samples and fuses task-relevant local features from the latent space, ensuring both global consistency and local precision. Third, to enhance safety-critical planning capabilities, we design a safety-aware **Reinforcement learning Fine-Tuning (RFT)** phase based on Group Relative Policy Optimization (GRPO) (Shao et al. 2024) to optimize explicit safety objectives. By gaussianizing predicted trajectories and employing collision-aware rewards, our approach transitions from passive behavior cloning to active collision avoidance.

Experiments demonstrate that WorldRFT achieves state-of-the-art (SOTA) performance on both open-loop nuScenes and closed-loop NavSim benchmarks, as shown in Figure 1. On the nuScenes dataset, compared to the baseline LAW (Li, Fan et al. 2025), average displacement error decreases by **21%** ($0.61\text{m} \rightarrow 0.48\text{m}$) and collision rate drops by **83%** ($0.30\% \rightarrow 0.05\%$). On the NavSim dataset, the PDMS metric improves by 3.2 points ($84.6 \rightarrow$ **87.8**), achieving the best performance among vision-only solutions and approaching the LiDAR-based SOTA method DiffusionDrive (88.1).

Our main contributions are summarized as follows:

- We propose the first planning-oriented latent world modeling paradigm that deeply aligns representation learning with planning tasks through the spatial geometric prior fusion and hierarchical planning interaction and local-aware refinement.
- We introduce a reinforcement fine-tuning phase that enhances safety planning via explicit safety rewards and GRPO method, transitioning from behavioral imitation to proactive collision avoidance.
- Our method achieves SOTA performance on both nuScenes and NavSim benchmarks, with substantial improvements in critical safety metrics.

Related Works

End-to-End Autonomous Driving

End-to-end autonomous driving models utilize the imitation learning framework with perception annotations to directly map sensor inputs to trajectory outputs(Gao et al. 2024; Yang et al. 2025). For instance, UniAD and VAD (Hu et al. 2023b; Jiang et al. 2023) fuse multiple perception and planning tasks within a cascaded BEV-based architecture, while PARA-Drive and ONE-Drive (Weng et al. 2024; Zheng et al. 2024b) leverage parallel architecture to enhance efficiency. Methods like SparseDrive (Sun et al. 2024) integrate detection, tracking, and mapping through symmetric sparse perception modules. Despite their advantages, these approaches require high-precision 3D perception annotations, leading to higher development complexity and costs.

World Models for Autonomous Driving

World models predict future scene states to improve dynamic environment understanding and support safer path planning. For instance, GAIA-1 (Hu et al. 2023a) creates realistic driving scenarios from multi-modal inputs and allows detailed control over vehicle behavior, while DriveDreamer4D (Zhao et al. 2025) generates novel trajectory videos from real driving data to enhance 4D reconstruction. Methods like DriveWorld and PreWorld (Min et al. 2024; Li et al. 2025; Gu et al. 2024; Jin et al. 2025) generate future occupancy and flow fields from BEV embeddings. In contrast, self-supervised approaches like LAW (Li, Fan et al. 2025) and SSR (Li and Cui 2025) model future scene dynamics without labeled data, but their relatively coarse planning frameworks can limit planning effectiveness, indicating a need for more sophisticated architectural designs.

Reinforcement Learning for Autonomous Driving

Reinforcement learning (RL) is increasingly important in autonomous driving for complex decision-making. RAD (Gao et al. 2025) builds virtual environments with 3D Gaussian Splatting to facilitate exploration and policy learning in diverse scenarios. AlphaDrive (Jiang et al. 2025) introduces RL to enhance planning and training of vision-language models. Given RL's advantage over imitation learning in handling causal confounding, we adopt RL for trajectory planning to improve safety and reduce collision risk in autonomous driving.

Methodology

Overview

As illustrated in Figure 2, WorldRFT comprises three key modules that form a complete pipeline of "scene understanding $\rightarrow$ planning decisions $\rightarrow$ safety optimization": (1) SWE constructs spatial-rich latent world representations from RGB images; (2) HPR efficiently extracts critical information strongly correlated with driving decisions; (3) RFT generates safer planning results through reinforcement learning optimization. These modules work synergistically to achieve safer end-to-end autonomous driving planning.

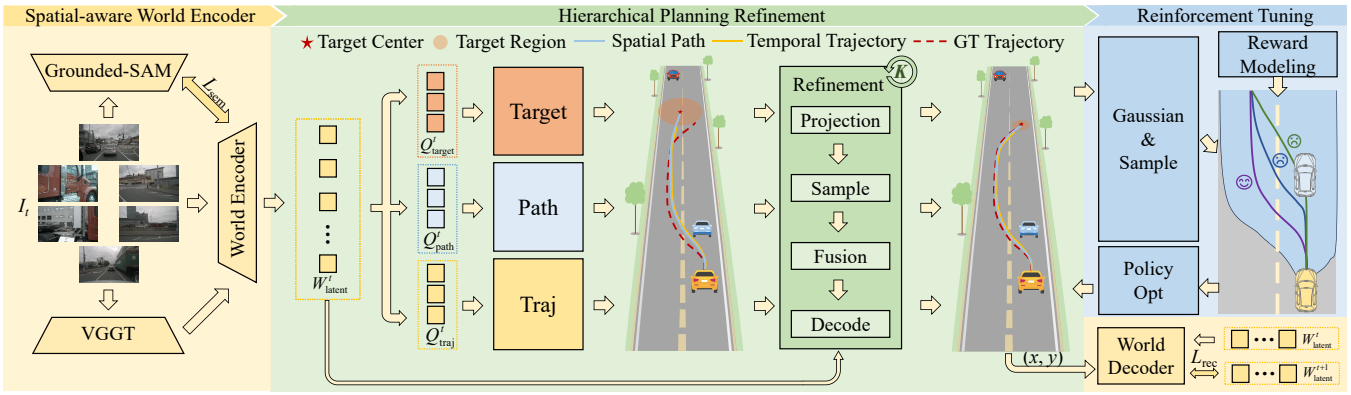

Figure 2: WorldRFT consists of three key modules: 1) Spatial-aware World Encoder (yellow) that extracts geometry-rich latent world representations from RGB images; 2) Hierarchical Planning Refinement Module (green) that efficiently captures critical information highly correlated with driving decisions through a refined planning task design; 3) Safety-aware Reinforcement Learning Fine-tuning phase (blue) that generates safer planning outcomes via reinforcement learning optimization. These modules work synergistically to deliver high-quality end-to-end autonomous driving planning.

Spatial-aware World Encoder

The spatial-aware world encoder constructs 3D spatially-aware latent world representations serving planning tasks from surround-view inputs through the geometric foundation model VGGT (Wang et al. 2025).

Basic Visual Feature Encoding. Given multi-view images $I_t \in \mathbb{R}^{M\times H\times W\times 3}$ at time t, we first extract features $F_t \in \mathbb{R}^{M\times h\times w\times D}$ through the image backbone. Following World4Drive (Zheng et al. 2025), we then utilize the vision-language foundation model Grounded-SAM for semantic supervised learning through the loss function $\mathcal{L}_{\text{sem}}$.

3D Spatial Encoding. Spatial understanding capability is crucial for planning tasks. VGGT, benefiting from its diverse training data and elegant feed-forward architecture, provides multi-view consistent geometric-rich prior from 2D surround images I_t. We leveraging VGGT's structural priors to enhance spatial awareness. Specifically, to maintain generic 3D aware capacity, we employ a frozen VGGT as a 3D spatial encoder and extract 3D tokens from its final layer:

$$\varepsilon_l(I_t) = \{t_c, t_r, t_{3D}\} \tag{1}$$

where l denotes the layer index, and t_c, t_r, and t_{3D} represent camera, register, and 3D tokens, respectively. We subsequently incorporate the 3D tokens t_{3D} into the image features F_t to construct spatial-aware latent world representations. Specifically, we propose a lightweight fusion module that integrates ResNet and VGGT features via a single cross-attention layer, where the 2D visual features F_t act as queries and the VGGT-derived 3D features t_{3D} are used as keys and values. This process ultimately yields the unified latent space visual representation W^t_{latent}, enabling effective spatial understanding without requiring explicit 3D inputs:

$$W^t_{\text{latent}} = \text{C-A}(F_t, t_{3D}) \tag{2}$$

where C-A denotes Cross-Attention layers.

Hierarchical Planning Refinement

In this section, we decompose end-to-end planning into three parallel subtasks that extract hierarchical features with distinct supervision: target region localization, spatial path planning, and temporal trajectory prediction. These tasks capture complementary aspects of planning and are unified through a query interaction framework and refined via local-aware iterative refinement.

Unified Query Interaction Framework. To facilitate coordination among parallel subtasks while maintaining their unique functionalities, we design a unified query interaction framework. Specifically, each subtask is initialized with dedicated queries: Q_{target}, Q_{path}, and Q_{traj}, which then independently aggregate hierarchical planning-relevant features from the latent world representation:

$$[Q'_{\text{target}}, Q'_{\text{path}}, Q'_{\text{traj}}] = \text{C-A}([Q_{\text{target}}, Q_{\text{path}}, Q_{\text{traj}}], W^t_{\text{latent}}) \tag{3}$$

To facilitate inter-task communication, these queries undergo concatenation and self-attention, enabling mutual awareness of planning intentions:

$$[Q''_{\text{target}}, Q''_{\text{path}}, Q''_{\text{traj}}] = \text{S-A}(\text{Concat}[Q'_{\text{target}}, Q'_{\text{path}}, Q'_{\text{traj}}]) \tag{4}$$

Target Region Localization. Predicting a deterministic target point for navigation (Xing et al. 2025) is an ill-posed problem due to the inherent uncertainty of valid target points in planning scenarios. Therefore, rather than predicting deterministic target points, we model targets as probabilistic regions using Laplace distributions. In detail, we parameterize the target region as:

$$(\mu, b) = \text{MLP}(Q''_{\text{target}}) \tag{5}$$

where $\mu \in \mathbb{R}^2$ denotes the region center and $b \in \mathbb{R}^2$ represents the scale parameters, quantifying the spatial extent of the target region. This formulation enables the model to learn target region distributions rather than fitting to a single deterministic point. Moreover, the probabilistic representation naturally captures scene complexity, thereby providing adaptive information that can be leveraged by the iterative refinement module to guide feature fusion.

The probabilistic representation is trained using the negative log-likelihood loss:

$$\mathcal{L}_{\text{Laplace-NLL}} = \log(2b) + \frac{\|\mathbf{y} - \boldsymbol{\mu}\|_1}{b} \tag{6}$$

Here, the scale parameter b naturally reflects scene complexity, with larger values corresponding to more challenging scenarios that demand cautious planning. As a conditioning signal, b modulates feature fusion within the uncertainty-aware local adaptation module, thereby enabling adaptive planning in accordance with scene uncertainty.

Spatial Path Planning. This module generates a spatial path connecting the current position to the target region. To ensure spatial consistency, we construct the spatial path ground truth by uniformly sampling future trajectory points at fixed spatial intervals, rather than temporal intervals (Li, Fan et al. 2025; Li and Cui 2025; Zheng et al. 2025; Jiang et al. 2023). Furthermore, we decode N spatial path points through an MLP network:

$$T_{\text{path}} = \text{MLP}(Q''_{\text{path}}) \tag{7}$$

where $T_{\text{path}} \in \mathbb{R}^{N\times 2}$ denotes path points sampled at 2-meter intervals, with each point denoting coordinates (x, y).

Temporal Trajectory Prediction. In this module, we decode future temporal trajectories for T timesteps through a trajectory decoder:

$$T_{\text{traj}} = \text{MLP}(Q''_{\text{traj}}) \tag{8}$$

where $T_{\text{traj}} \in \mathbb{R}^{T\times 2}$ represents trajectory points at fixed temporal intervals (0.5 seconds), with each point containing coordinates (x, y).

Local-aware Iterative Refinement. This module iteratively refines the initial planning outputs by integrating local scene information. It takes the preliminary results (μ, b), T_{path}, and T_{traj} as inputs, and refines the outputs of all three subtasks over K iterations. Using temporal trajectory refinement as an example, each iteration involves the following key steps as shown in Figure 3:

First, we encode current planning information (μ, b), T_{path}, and T_{traj} into a unified state representation F_s:

$$F_s = \text{MLP}(\text{Concat}[(\mu^{(k)}, b^{(k)}), T^{(k)}_{\text{path}}, T^{(k)}_{\text{traj}}]) \tag{9}$$

Subsequently, the trajectory points are projected onto the latent space feature map using the camera parameters, yielding corresponding positions P_{proj}. Deformable convolution is then applied to adaptively sample the local features F_{local} at these positions:

$$P_{\text{proj}} = \text{CameraProject}(T^{(k)}_{\text{traj}}) \tag{10}$$

$$F_{\text{local}} = \text{DeformConv}(W^t_{\text{latent}}, P_{\text{proj}}) \tag{11}$$

Next, local and global features are fused, guided by the scale parameter b obtained from target region modeling, which acts as a conditioning signal:

$$F_b = \text{MLP}(b^{(k)}) \tag{12}$$

$$F_{\text{fusion}} = \text{MLP}(\text{Concat}[F_{\text{local}}, Q''_{\text{traj}}, F_s, F_b]) \tag{13}$$

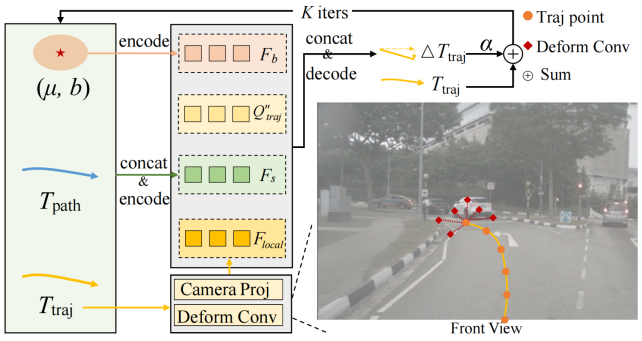

Figure 3: Architecture of the Local-aware Iterative Refinement module. The module refines preliminary planning results $(\mu, b), T_{\text{path}}, T_{\text{traj}}$ through K iterations. Using T_{traj} as an example, each iteration: (1) encodes global planning states, (2) projects trajectory points via camera parameters, (3) samples local features using deformable convolution, and (4) fuses them with global information and uncertainty representation. Residual connections enable incremental updates for adaptive local adjustment.

where Q''_{traj} represents the global planning intention.

Finally, independent prediction heads generate the corresponding trajectory offsets $\Delta T^{(k)}_{\text{traj}}$, which are then used to incrementally update the trajectory via residual connections:

$$\Delta T^{(k)}_{\text{traj}} = \text{MLP}(F_{\text{fusion}}) \tag{14}$$

$$T^{(k+1)}_{\text{traj}} = T^{(k)}_{\text{traj}} + \alpha \Delta T^{(k)}_{\text{traj}} \tag{15}$$

where $\alpha = 0.1$ is the update step size.

Training Loss. We pretrain our model guided by the latent world modeling, which enables self-supervised learning without requiring perception annotations by leveraging temporal consistency in future world prediction. Following (Li, Fan et al. 2025), we use a world decoder to predict future latent world representations $\widehat{W}^{t+1}_{\text{latent}}$ from the current latent representation W^t_{latent} and the temporal trajectory T_{traj}. We self-supervise world prediction through the MSE loss:

$$\mathcal{L}_{\text{rec}} = \text{MSE}(\widehat{W}^{t+1}_{\text{latent}}, W^{t+1}_{\text{latent}}) \tag{16}$$

Additionally, we employ L1 loss $\mathcal{L}_{\text{traj}}$ to align both the spatial path T_{path} and temporal trajectory T_{traj} with expert demonstrations. The final training objective combines multiple loss terms:

$$\mathcal{L}_{\text{Total}} = \alpha\mathcal{L}_{\text{sem}} + \beta\mathcal{L}_{\text{rec}} + \gamma\mathcal{L}_{\text{target}} + \eta\mathcal{L}_{\text{traj}} \tag{17}$$

where α, β, γ and η are hyper-parameters.

Reinforcement Learning Fine-tuning

In this phase, the module introduces collision-aware rewards, extends trajectories to probability distributions for exploration, and applies GRPO to improve safety while maintaining planning accuracy.

Collision-aware Reward Design. Our RL framework targets collision risks between the ego vehicle and surrounding

agents. The collision-aware reward is based on the distance between the ego vehicle's bounding box and those of nearby agents along the trajectory: negative distances (collisions) incur penalties, while non-collisions yield zero reward:

$$r = \begin{cases} -1 & \text{if collision happens} \\ 0 & \text{if no collision happens} \end{cases} \tag{18}$$

Gaussianized-Trajectory Modeling. Since the above trajectory prediction is formulated as a regression task, effective application of RL requires recasting it as a probabilistic, distribution-based problem. To this end, we model trajectories using Gaussian distributions, where the predicted trajectory serve as the means $\boldsymbol{\mu}_\theta$, and an auxiliary variance network adaptively estimates the trajectory variances $\boldsymbol{\Sigma}_\theta$.

Policy Optimization. Based on the world model structure, we optimize generated trajectories using GRPO (Shao et al. 2024). Following the GRPO framework, we estimate advantage functions based on relative rewards within sample groups. Specifically, a group of G trajectories is sampled using the previously described latent world model, which serves as the policy π_θ: $\left\{T_{\text{traj}_0}, T_{\text{traj}_1}, ..., T_{\text{traj}_{G-1}}\right\}$. For each point along these trajectories, the relative reward is defined as the normalized per-point reward value:

$$\tilde{r}_j^{T_{\text{traj}_i}} = \frac{r_j^{T_{\text{traj}_i}} - \text{mean}\left(\boldsymbol{r}_j\right)}{\text{std}\left(\boldsymbol{r}_j\right)} \tag{19}$$

where $r_j^{T_{\text{traj}_i}}$ denotes the value of the original reward function, and $\boldsymbol{r}_j = \left\{r_j^{T_{\text{traj}_0}}, r_j^{T_{\text{traj}_1}}, ..., r_j^{T_{\text{traj}_{G-1}}}\right\}$ denotes the values of the group-based reward function.

Given that the predicted trajectories are represented as time-differential increments, each increment at the current time step influences both the current and all subsequent points, leading to potential error accumulation. Accordingly, the relative advantage function for the j-th trajectory point is defined as:

$$Adv_j^{T_{\text{traj}_i}} = \sum_{t \geq j} \tilde{r}_t^{T_{\text{traj}_i}} \tag{20}$$

The policy optimization objective function becomes:

$$J\left(\theta\right) = \frac{1}{G} \sum_i \sum_j \left\{ \min\left[f_1, f_2\right] - \beta D_{KL}\left(\pi_\theta || \pi_{\theta_{ref}}\right) \right\} \tag{21}$$

where:

$$f_1 = \left(\frac{\pi_\theta}{\pi_{\theta old}}\right) Adv_j^{T_{\text{traj}_i}} \tag{22}$$

$$f_2 = \text{clip}\left(\frac{\pi_\theta}{\pi_{\theta_{old}}}, 1-\varepsilon, 1+\varepsilon\right) Adv_j^{T_{\text{traj}_i}} \tag{23}$$

where ε limits the update step size of the policy function to ensure algorithmic stability, and β is the coefficient for the Kullback-Leibler (KL) divergence. $\pi_{\theta_{old}}$ denotes the old policy, and $\pi_{\theta_{ref}}$ represents the reference policy, which is the pre-trained model. As mentioned above, we model the output of the trajectory as a Gaussian distribution with mean vector $\boldsymbol{\mu}_\theta$ and covariance matrix $\boldsymbol{\Sigma}_\theta$. The reference model provides a deterministic result $\boldsymbol{\mu}_{ref} = T_{\text{traj}_{ref}}$, which is used to compute the negative log-likelihood (NLL) loss under the Gaussian distribution. Thus, the KL divergence is defined as:

$$D_{KL} = \frac{1}{2} \left[\begin{array}{l} \log|\boldsymbol{\Sigma}_\theta| + (\boldsymbol{\mu}_{ref} - \boldsymbol{\mu}_\theta)^T \boldsymbol{\Sigma}_\theta^{-1} (\boldsymbol{\mu}_{ref} - \boldsymbol{\mu}_\theta) \\ +2\log(2\pi) \end{array} \right] \tag{24}$$

RFT Training Loss. We fine-tune the model using the GRPO objective function and KL loss. Therefore, the loss for reinforcement learning can be written as:

$$\mathcal{L}_{\text{RL}} = -J\left(\theta\right) + \lambda D_{KL} \tag{25}$$

Experiment

Overview

This section presents the experimental results of the proposed method, including ablation studies for each component. For completeness, detailed descriptions of the dataset, experimental setup, evaluation metrics, hyperparameter settings, scalability analysis, and additional visualizations are provided in the supplementary material.

Benchmark

We comprehensively evaluate our method on the open-loop nuScenes benchmark (Caesar et al. 2020) and the closed-loop NavSim benchmark (Dauner et al. 2024). On nuScenes, we report displacement error (L2) and collision rate (CR) for trajectory prediction. For NavSim, we adopt the PDM Score (PDMS), which combines performance across five dimensions: no at-fault collision (NC), drivable area compliance (DAC), time-to-collision (TTC), ride comfort (Comf.), and ego progress (EP).

Main Results

As demonstrated in Table 1, we compare our framework with several SOTA methods. Specifically, WorldRFT achieves SOTA performance among perception annotation-free approaches, demonstrating a 21% reduction in L2 error (0.61 m $\rightarrow$ 0.48 m) and an impressive 83% reduction in collision rate (0.30% $\rightarrow$ 0.05%) compared to the strong baseline LAW. Moreover, WorldRFT achieves the lowest collision rate among all evaluated methods, even surpassing perception-based approaches, which underscores the superior safety performance of our planning-oriented design. We also provide results incorporating the ego status, with specific descriptions detailed in the supplementary materials.

In addition, as demonstrated in Table 2, WorldRFT also achieves SOTA performance in closed-loop evaluation with PDMS of 87.8 using only camera inputs. Compared to the baseline LAW (84.6), our approach demonstrates significant improvements across all safety-critical metrics: NC (No At-fault Collision) (96.4 $\rightarrow$ 97.8), TTC (Time-to-Collision) (88.7 $\rightarrow$ 94.0), and particularly DAC (Drivable Area Compliance) (95.4 $\rightarrow$ 96.8). The exceptional DAC score of 96.8, which is the highest among all methods including LiDAR-based approaches, demonstrates that incorporating VGGT

Method	Training Approach	L2 (m) ↓				Collision Rate (%) ↓			
		1s	2s	3s	Avg.	1s	2s	3s	Avg.
ST-P3 (Hu et al. 2022)	P-IL	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71
OccNet (Tong et al. 2023)	P-IL	1.29	2.13	2.99	2.13	0.21	0.59	1.37	0.72
UniAD (Hu et al. 2023b)	P-IL	0.48	0.96	1.65	1.03	0.05	0.17	0.71	0.31
VAD (Jiang et al. 2023)	P-IL	0.41	0.70	1.05	0.72	0.07	0.18	0.43	0.23
UncAD (Yang et al. 2025)	P-IL	0.33	0.59	0.94	0.62	0.10	0.14	0.28	0.17
PPAD (Chen et al. 2024b)	P-IL	0.31	0.56	0.87	0.58	0.08	0.12	0.38	0.19
PARA-Drive (Weng et al. 2024)	P-IL	0.25	0.46	**0.74**	0.48	0.14	0.23	0.39	0.25
GenAD (Zheng et al. 2024a)	P-IL	0.28	0.49	0.78	0.52	0.08	0.14	0.34	0.19
LAW (Li, Fan et al. 2025) (Perception-based)	P-IL	0.24	0.46	0.76	0.49	0.08	0.10	0.39	0.19
DiffusionDrive (Liao, Chen et al. 2025)	P-IL	0.27	0.54	0.90	0.57	0.03	0.05	**0.16**	0.08
Epona (Zhang et al. 2025)	SS-L	0.61	1.17	1.98	1.25	0.01	0.22	0.85	0.36
LAW (Li, Fan et al. 2025) (Perception-free)	SS-L	0.26	0.57	1.01	0.61	0.14	0.21	0.54	0.30
World4Drive (Zheng et al. 2025)	SS-L	0.23	0.47	0.81	0.50	0.02	0.12	0.33	0.16
Ours (without RFT)	SS-L	**0.21**	**0.44**	0.76	**0.47**	0.10	0.11	0.23	0.15
Ours (with RFT)	SS-L & RL	0.22	**0.44**	0.77	0.48	**0.00**	**0.00**	**0.16**	**0.05**
SSR† (Li and Cui 2025)	SS-L	0.19	0.36	0.62	0.39	0.00	0.10	0.20	0.13
Ours (with RFT)†	SS-L & RL	0.15	0.30	0.56	0.33	0.00	0.00	0.12	0.04

P-IL: Perception-based Imitation Learning; SS-L: Self-Supervised Learning; RL: Reinforcement Learning.
† Represents methods incorporating ego status.

Table 1: End-to-end planning results on nuScenes benchmark (Caesar et al. 2020)

Method	Training Approach	Input	NC ↑	DAC ↑	TTC ↑	Comf. ↑	EP ↑	PDMS ↑
UniAD (Hu et al. 2023b)	P-IL	C	97.8	91.9	92.9	**100.0**	78.8	83.4
PARA-Drive (Weng et al. 2024)	P-IL	C	97.9	92.4	93.0	99.8	79.3	84.0
LTF (Prakash, Chitta, and Geiger 2021)	P-IL	C	97.4	92.8	92.4	**100.0**	79.0	83.8
Transfuser (Prakash, Chitta, and Geiger 2021)	P-IL	C & L	97.7	92.8	92.8	**100.0**	79.2	84.0
VADv2 (Chen et al. 2024a)	P-IL	C & L	97.2	89.1	91.6	**100.0**	76.0	80.9
Hydra-MDP (Li et al. 2024)	P-IL	C & L	97.9	91.7	92.9	**100.0**	77.6	83.0
DiffusionDrive (Liao, Chen et al. 2025)	P-IL	C & L	**98.2**	96.2	**94.7**	**100.0**	**82.2**	**88.1**
Epona (Zhang et al. 2025)	SS-L	C	97.9	95.1	93.8	**99.9**	80.4	86.2
LAW (Perception-free) (Li, Fan et al. 2025)	SS-L	C	96.4	95.4	88.7	99.9	81.7	84.6
DriveX (Shi et al. 2025)	SS-L	C	97.5	94.0	93.0	**100.0**	79.7	84.5
World4Drive (Zheng et al. 2025)	SS-L	C	97.4	94.3	92.8	**100.0**	79.9	85.1
Ours (without RFT)	SS-L	C	97.5	96.0	94.0	**100.0**	80.9	87.0
Ours (with RFT)	SS-L & RL	C	97.8	**96.8**	94.0	**100.0**	81.7	87.8

P-IL: Perception-based Imitation Learning; SS-L: Self-Supervised Learning; RL: Reinforcement Learning.
C: Camera modality; L: LiDAR modality.

Table 2: End-to-end planning results on NavSim benchmark (Dauner et al. 2024)

geometric priors substantially enhances the model's 3D spatial understanding without explicit depth supervision.

Notably, our closed-loop performance exceeds that of most methods relying on LiDAR inputs, trailing DiffusionDrive (88.1) by only 0.3 points. This result demonstrates that our planning-oriented latent world modeling, combined with safety-oriented reinforcement fine-tuning, achieves outstanding performance and holds strong potential for real-world deployment.

Ablation Study

In this section, we conduct comprehensive ablation studies to investigate the effectiveness of each component in our proposed WorldRFT framework. All ablation experiments incorporate our safety-oriented reinforcement fine-tuning by default to ensure consistent safety-aware evaluation. All of the ablation experiments are conducted on the nuScenes and NavSim benchmark. For clarity, we present detailed results on nuScenes here, while NavSim results are provided in the supplementary material.

Effectiveness of VGGT Spatial Encoding. As demonstrated in Table 3, incorporating VGGT geometric priors shows consistent improvements across different configurations. Comparing ID 4 with ID 8, adding VGGT reduces L2 error by 7.7% (0.52 → 0.48) and collision rate by 37.5% (0.08 → 0.05). The performance gains remain substantial even with partial task designs (ID 5-6), demonstrating that VGGT-enhanced spatial understanding benefits planning re-

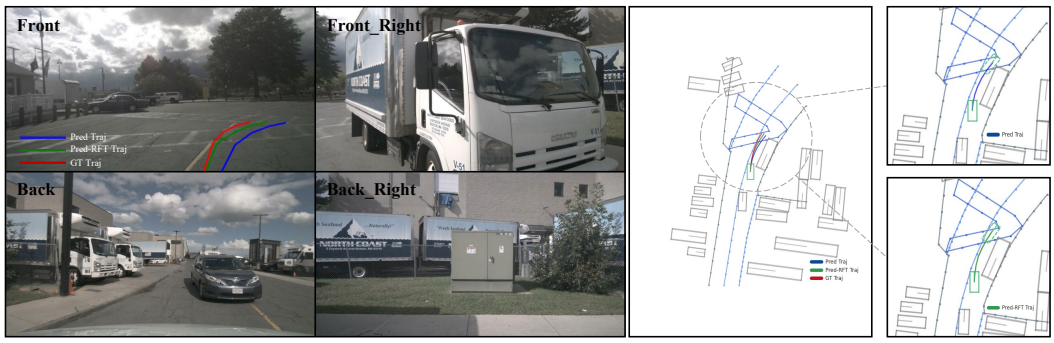

Figure 4: Visualization of planning trajectories, where the red line is the ground truth, the blue line is the pretrained-only trajectory, and the green line shows the RFT-trajectory. Obviously, the pre-trained trajectory dangerously approaches obstacles, risking collisions. In contrast, the RFT-trajectory proactively adjusts to maintain safe distances, avoiding collisions.

ID	Latent Encoder	Planning Task		Refine -ment	L2	CR
	VGGT	Target	Path			
1					0.59	0.16
2		✓		✓	0.56	0.10
3			✓	✓	0.54	0.11
4		✓	✓	✓	0.52	0.08
5	✓		✓	✓	0.51	0.08
6	✓	✓		✓	0.49	0.09
7	✓	✓	✓		0.50	0.06
8	✓	✓	✓	✓	0.48	0.05

Table 3: Ablation study of each proposed component

gardless of the specific task configuration.

Impact of Hierarchical Planning Tasks. Starting from the baseline (ID 1 in Table 3), introducing target region localization and spatial path planning with refinement (ID 4) reduces L2 error by 11.9% (0.59 → 0.52) and collision rate by 50.0% (0.16 → 0.08). This validates our hypothesis that decomposing the complex planning task into hierarchical subtasks enables more effective feature extraction from latent representations, thereby enhancing overall planning performance.

Local-aware Iterative Refinement. As shown in Table 3, comparing ID 7 (without refinement) and ID 8 (with refinement), the refinement module reduces L2 error by 4.0% (0.50 → 0.48) and collision rate by 16.7% (0.06 → 0.05). This demonstrates that incorporating local features based on initial trajectories enables fine-grained trajectory adjustments that improve both trajectory accuracy and safety.

Safety-Oriented Reinforcement Fine-tuning. As shown in Table 1, comparing our method without RFT (Ours without RFT) and with RFT (Ours with RFT), the RL module significantly reduces collision rate by 66.7% (0.15 → 0.05) while L2 error slightly increases from 0.47 to 0.48. This selective improvement in safety metrics validates that our collision-aware reward design guides the model from behavior imitation toward understanding safety principles.

Qualitative Results In this section, we quantitatively evaluate WorldRFT on nuScenes. Figure 4 shows predicted trajectories: ground truth (red), without RFT (blue), and with RFT (green). The without-RFT model approaches surrounding vehicles too closely, while the with-RFT model maintains safer distances and better matches the ground truth. As shown in Figure 5, attention maps (LAW middle, WorldRFT below) demonstrate that WorldRFT focuses more precisely on critical regions like surrounding vehicles, improving perception-planning integration and yielding more robust, human-like behavior.

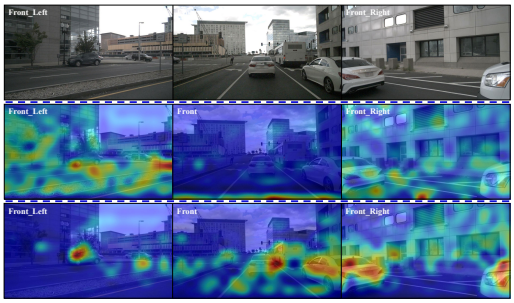

Figure 5: Visualization of the original image (top), along with attention maps for the baseline (middle) and WorldRFT (bottom), where blue represents weak attention and red indicates strong attention. The maps reveal that WorldRFT focuses on surrounding planning-critical agents such as vehicles, which are largely overlooked by the baseline method.

Conclusion

We present WorldRFT, a planning-oriented latent world framework for end-to-end autonomous driving. Our key insight is that existing latent world models suffer from misalignment between reconstruction-oriented learning and planning requirements. To bridge this gap, we introduced three innovations: (1) SWE with VGGT for spatial-aware representations; (2) HPR combining hierarchical task decomposition with iterative refinement for planning-relevant features; and (3) RFT enabling active collision avoidance beyond passive imitation. Experiments on nuScenes and NavSim demonstrate that our planning-oriented design with geometric priors and safety optimization effectively enables slight end-to-end autonomous driving.

Acknowledgments

This work is supported by the National Key Research and Development Program of China under Grant 2022YFA1004000, Beijing Natural Science Foundation-Xiaomi Innovation Joint Fund L253007, Beijing Natural Science Foundation under Grant 4242052, and the National Natural Science Foundation of China (NSFC) under Grant 62173325.

References

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuScenes: A Multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11621–11631.

Chen, S.; Jiang, B.; Gao, H.; Liao, B.; Xu, Q.; Zhang, Q.; Huang, C.; Liu, W.; and Wang, X. 2024a. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*.

Chen, Z.; Ye, M.; Xu, S.; Cao, T.; and Chen, Q. 2024b. Ppad: Iterative interactions of prediction and planning for end-to-end autonomous driving. In *Proceedings of the European Conference on Computer Vision*, 239–256. Springer.

Dauner, D.; Hallgarten, M.; Li, T.; Weng, X.; Huang, Z.; Yang, Z.; Li, H.; Gilitschenski, I.; Ivanovic, B.; Pavone, M.; Geiger, A.; and Chitta, K. 2024. NAVSIM: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *arXiv preprint arXiv:2406.15349*.

Gao, H.; Chen, S.; Jiang, B.; Liao, B.; Shi, Y.; Guo, X.; Pu, Y.; Yin, H.; Li, X.; Zhang, X.; Zhang, Y.; Liu, W.; Zhang, Q.; and Wang, X. 2025. Rad: Training an end-to-end driving policy via large-scale 3dgs-based reinforcement learning. *arXiv preprint arXiv:2502.13144*.

Gao, Y.; Zhang, Q.; Ding, D.-W.; and Zhao, D. 2024. Dream to Drive With Predictive Individual World Model. *IEEE Transactions on Intelligent Vehicles*, 9(12): 8224–8238.

Gu, S.; Yin, W.; Jin, B.; Guo, X.; Wang, J.; Li, H.; Zhang, Q.; and Long, X. 2024. Dome: Taming diffusion model into high-fidelity controllable occupancy world model. *arXiv preprint arXiv:2410.10429*.

Hu, A.; Russell, L.; Yeo, H.; Murez, Z.; Fedoseev, G.; Kendall, A.; Shotton, J.; and Corrado, G. 2023a. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*.

Hu, M.; Yin, W.; Zhang, C.; Cai, Z.; Long, X.; Chen, H.; Wang, K.; Yu, G.; Shen, C.; and Shen, S. 2024. Metric3d v2: A versatile monocular geometric foundation model for zero-shot metric depth and surface normal estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Hu, S.; Chen, L.; Wu, P.; Li, H.; Yan, J.; and Tao, D. 2022. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *Proceedings of the European Conference on Computer Vision*, 533–549. Springer.

Hu, Y.; Yang, J.; Chen, L.; Li, K.; Sima, C.; Zhu, X.; Chai, S.; Du, S.; Lin, T.; Wang, W.; Lu, L.; Jia, X.; Liu, Q.; Dai, J.; Qiao, Y.; and Li, H. 2023b. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 17853–17862.

Jia, X.; You, J.; Zhang, Z.; and Yan, J. 2025. Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. In *Proceedings of the International Conference on Learning Representations*.

Jiang, B.; Chen, S.; Xu, Q.; Liao, B.; Chen, J.; Zhou, H.; Zhang, Q.; Liu, W.; Huang, C.; and Wang, X. 2023. VAD: Vectorized scene representation for efficient autonomous driving. *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Jiang, B.; Chen, S.; Zhang, Q.; Liu, W.; and Wang, X. 2025. AlphaDrive: Unleashing the power of VLMs in autonomous driving via reinforcement learning and reasoning. *arXiv preprint arXiv:2503.07608*.

Jin, B.; Gu, S.; Hu, X.; Zheng, Y.; Guo, X.; Zhang, Q.; Long, X.; and Yin, W. 2025. OccTENS: 3D Occupancy World Model via Temporal Next-Scale Prediction. *arXiv preprint arXiv:2509.03887*.

Li, P.; and Cui, D. 2025. Navigation-guided sparse scene representation for end-to-end autonomous driving. In *Proceedings of the International Conference on Learning Representations*, 15522–15533.

Li, X.; Li, P.; Zheng, Y.; Sun, W.; Wang, Y.; and Chen, Y. 2025. Semi-supervised vision-centric 3d occupancy world model for autonomous driving. *arXiv preprint arXiv:2502.07309*.

Li, Y.; Fan, L.; et al. 2025. Enhancing end-to-end autonomous driving with latent world model. *Proceedings of the International Conference on Learning Representations*.

Li, Z.; Li, K.; Wang, S.; Lan, S.; Yu, Z.; Ji, Y.; Li, Z.; Zhu, Z.; Kautz, J.; Wu, Z.; Jiang, Y.; and Alvarez, J. M. 2024. Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. *arXiv preprint arXiv:2406.06978*.

Liao, B.; Chen, S.; et al. 2025. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Min, C.; Zhao, D.; Xiao, L.; Zhao, J.; Xu, X.; Zhu, Z.; Jin, L.; Li, J.; Guo, Y.; Xing, J.; Jing, L.; Nie, Y.; and Dai, B. 2024. Driveworld: 4D pre-trained scene understanding via world models for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15522–15533.

Prakash, A.; Chitta, K.; and Geiger, A. 2021. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Shao, Z.; Wang, P.; Zhu, Q.; Xu, R.; Song, J.; Bi, X.; Zhang, H.; Zhang, M.; Li, Y., Y.K.and Wu; and Guo, D. 2024. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Shi, C.; Shi, S.; Sheng, K.; Zhang, B.; and Jiang, L. 2025. DriveX: Omni scene modeling for learning generalizable

world knowledge in autonomous driving. *arXiv preprint arXiv:2505.19239*.

Sun, W.; Lin, X.; Shi, Y.; Zhang, C.; Wu, H.; and Zheng, S. 2024. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620*.

Tong, W.; Sima, C.; Wang, T.; Chen, L.; Wu, S.; Deng, H.; Gu, Y.; Lu, L.; Luo, P.; Lin, D.; and H., L. 2023. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8406–8415.

Wang, J.; Chen, M.; Karaev, N.; Vedaldi, A.; Rupprecht, C.; and Novotny, D. 2025. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 5294–5306.

Weng, X.; Ivanovic, B.; Wang, Y.; Wang, Y.; and Pavone, M. 2024. PARA-Drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15449–15458.

Xing, Z.; Zhang, X.; Hu, Y.; Jiang, B.; He, T.; Zhang, Q.; Long, X.; and Yin, W. 2025. Goalflow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 1602–1611.

Yang, P.; Zheng, Y.; Zhang, Q.; Zhu, K.; Xing, Z.; Lin, Q.; Liu, Y.-F.; Su, Z.; and Zhao, D. 2025. UncAD: Towards Safe End-to-end Autonomous Driving via Online Map Uncertainty. *arXiv preprint arXiv:2504.12826*.

Zhang, K.; Tang, Z.; Hu, X.; Pan, X.; Guo, X.; Liu, Y.; Huang, J.; Yuan, L.; Zhang, Q.; Long, X.; Cao, X.; and Yin, W. 2025. Epona: Autoregressive diffusion world model for autonomous driving. *arXiv preprint arXiv:2506.24113*.

Zhao, G.; Ni, C.; Wang, X.; Zhu, Z.; Zhang, X.; Wang, Y.; Huang, G.; Chen, X.; Wang, B.; Zhang, Y.; Mei, W.; and Wang, X. 2025. DriveDreamer4D: World models are effective data machines for 4d driving scene representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12015–12026.

Zheng, W.; Song, R.; Guo, X.; Zhang, C.; and Chen, L. 2024a. Genad: Generative end-to-end autonomous driving. In *European Conference on Computer Vision*, 87–104. Springer.

Zheng, Y.; Xia, Z.; Zhang, Q.; Zhang, T.; Lu, B.; Huo, X.; Han, C.; Li, Y.; Yu, M.; Jin, B.; et al. 2024b. Preliminary investigation into data scaling laws for imitation learning-based end-to-end autonomous driving. *arXiv preprint arXiv:2412.02689*.

Zheng, Y.; Yang, P.; Xing, Z.; Zhang, Q.; Zheng, Y.; Gao, Y.; Li, P.; Zhang, T.; Xia, Z.; Jia, P.; and Zhao, D. 2025. World4drive: End-to-end autonomous driving via intention-aware physical latent world model. *arXiv preprint arXiv:2507.00603*.