
RLGF: Reinforcement Learning with Geometric Feedback for Autonomous Driving Video Generation

Tianyi Yan*
SKL-IOTSC, CIS
University of Macau
tianyi.yan123@gmail.com

Wencheng Han*
SKL-IOTSC, CIS
University of Macau

Xia Zhou
Li Auto Inc.

Xueyang Zhang
Li Auto Inc.

Kun Zhan
Li Auto Inc.

Cheng-zhong Xu
SKL-IOTSC, CIS
University of Macau

Jianbing Shen[†]
SKL-IOTSC, CIS
University of Macau

Abstract

Synthetic data is crucial for advancing autonomous driving (AD) systems, yet current state-of-the-art video generation models, despite their visual realism, suffer from subtle geometric distortions that limit their utility for downstream perception tasks. We identify and quantify this critical issue, demonstrating a significant performance gap in 3D object detection when using synthetic versus real data. To address this, we introduce Reinforcement Learning with Geometric Feedback (RLGF), RLGF uniquely refines video diffusion models by incorporating rewards from specialized latent-space AD perception models. Its core components include an efficient Latent-Space Windowing Optimization technique for targeted feedback during diffusion, and a Hierarchical Geometric Reward (HGR) system providing multi-level rewards for point-line-plane alignment, and scene occupancy coherence. To quantify these distortions, we propose GeoScores. Applied to models like DiVE on nuScenes, RLGF substantially reduces geometric errors (e.g., VP error by 21%, Depth error by 57%) and dramatically improves 3D object detection mAP by 12.7%, narrowing the gap to real-data performance. RLGF offers a plug-and-play solution for generating geometrically sound and reliable synthetic videos for AD development.

1 Introduction

The rapid progress of autonomous driving (AD) systems [15, 17, 6] has created a growing need for high-quality synthetic data. Recent diffusion-based video generation methods [57, 61, 30, 10, 9, 18, 19] have achieved state-of-the-art visual realism, measured by metrics like FVD [49]. However, we identify a critical yet underexplored limitation: the generated videos often contain subtle yet impactful incorrect geometric relationships. This flaw not only misleads downstream perception and planning tasks but also undermines the reliability of models trained or evaluated using such data, significantly constraining their applicability in essential use cases such as simulation-based training [30] and system validation [61, 67].

To investigate this geometric limitation, we conduct a series of targeted experiments. Evaluating 3D object detection on synthetic videos using BEVFusion [26] reveals a substantial performance drop compared to real data (mAP: 25.7 vs. 35.5). In contrast, 2D object detection using YOLOv5 [20]

*Equal contribution

[†]Corresponding author

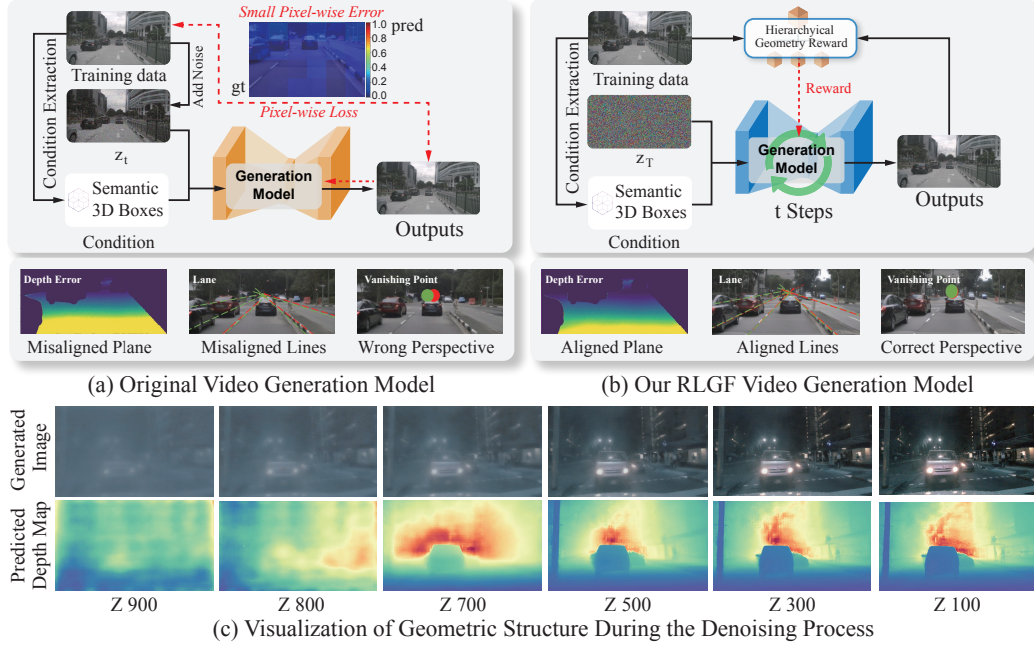


Figure 1: (a) **Original Video Generation Models**, optimized via pixel-level supervision (e.g., noise prediction error), often produce visually plausible videos that nonetheless suffer from severe geometric flaws (misaligned planes/lines, wrong perspective). This can degrade downstream tasks like 3D object detection (e.g., mAP drop from 35.5 to 25.7). (b) **Our RLGF** integrates a Hierarchical Geometry Reward directly into the multi-step denoising process. This reward, derived from perception models, guides the generation model to produce outputs with aligned planes, correct lane structures, and accurate perspective. (c) **Visualized depth maps** from noisy latents at various denoising stages (from noisy to less noisy) show coarse geometry emerging early and details later. This motivates our Latent-Space Windowing Optimization for targeted intermediate rewards.

on the same data yields results comparable to real-world samples (mAP: 43.8 vs. 44.7). These findings suggest that while current diffusion models [57, 56, 18, 9] preserve 2D appearance, indicating minimal image-level domain gap, yet they fail to capture accurate 3D scene structure. We attribute this primarily to underlying geometric inconsistencies. To further verify this hypothesis and systematically quantify these distortions, we introduce GeoScores. This metric suite evaluates geometric fidelity by applying pre-trained perception models [65, 42] to both synthetic videos and their corresponding real-world counterparts, using the outputs from real videos as reference ground truth. Significant discrepancies between the two highlight geometric errors. GeoScores (details are in appendix A.1) reveals three major issues: (1) Vanishing point shifts, indicating incorrect global perspective; (2) Lane topology inconsistencies, reflecting misaligned road markings and implausible lane structures; and (3) Depth errors, particularly on road surfaces, signifying incorrect placement. (e.g., for a typical baseline, an average VP shift of 0.086 normalized units, a Lane F1-Score of only 0.792, and an average depth error of 1.822). The significant deviations in these scores confirm that current "high-quality" synthetic data often suffers from pervasive geometric inaccuracies (fig. 1(a)).

Addressing these challenges, we present Reinforcement Learning with Geometric Feedback (RLGF), a novel framework that injects perception-model-driven geometric spatial constraints directly into the video generation process. Unlike conventional approaches [57, 56, 61, 18] that primarily rely on pixel-wise alignment, which often fail to explicitly enforce adherence to complex, underlying geometric principles, RLGF leverages dedicated, pre-trained AD perception models as reward providers to ensure geometric fidelity.

RLGF introduces two core technical innovations: Latent-Space Windowing Optimization and Hierarchical Geometric Alignment. First, we present Latent-Space Windowed Optimization. We observe that geometric structures in diffusion models emerge progressively across denoising steps: early steps

(e.g., before step 10 in flow matching [27, 28]) establish coarse global geometry, while later steps refine local details (fig. 1(c)). Training across the entire sampling process, as done in some prior RL-diffusion work [1] struggles to provide targeted guidance for these distinct phases. Therefore, we propose an efficient latent-space training strategy where rewards are applied directly to noisy latent features within a randomly sampled sliding window of intermediate diffusion steps. This approach significantly reduces computational (GPU memory) burden and, more importantly, allows for effective and targeted corrective feedback during both early (global structure formation) and late (detail refinement) stages of geometric synthesis.

Secondly, RLGF features Hierarchical Geometric Reward (HGR), a multi-level feedback system designed to imbue generated videos with robust geometric fidelity and scene coherence. HGA integrates signals from two specialized latent-space perception networks: a Latent Geometry Perception Model assessing vanishing point, lane, and depth cues, and a Latent Occupancy Prediction Model inferring 3D scene occupancy. Both operate efficiently on noisy latents, aided by a lightweight micro-decode module to circumvent costly full decoding. Leveraging these perception models, HGA then constructs its multi-level reward system. For point-line-plane geometric feedback, we use the outputs of the perception model to: (1) enforce vanishing point consistency for accurate global perspective, (2) ensure lane topology validity for realistic road structure, and (3) promote depth coherence for correct surface and object geometry. For scene-level occupancy feedback, outputs from the occupancy model are used to: (4) align intermediate semantic features using KL divergence for plausible scene evolution, and (5) maximize 3D occupancy IoU for accurate volumetric object layout and dynamics. This structured approach, combining efficient latent-space perception with a multi-faceted reward design, ensures that comprehensive geometric and occupancy information is extracted and effectively fed back to guide the video diffusion model towards producing physically principled and geometrically sound autonomous driving scenarios.

Our contributions are fourfold:

- We are the first to systematically quantify the geometric distortion problem in autonomous driving video generation and propose the GeoScores metric for its evaluation.
- We introduce RLGF, a novel paradigm that uses reinforcement learning with perception-based rewards applied efficiently within a sliding window in latent space, enabling plug-and-play geometric correction.
- We design HGR which addresses geometric distortions by incorporating point-line-plane and scene-level occupancy multi-level geometric feedback derived from latent representations.
- Extensive experiments on nuScenes demonstrate RLGF’s plug-and-play effectiveness across two baselines [18, 9], boosting 3D detection mAP by 12.7% absolute while reduce geometry gap (via GeoScores) relative to real data. This work establishes a new paradigm for geometrically faithful synthetic data generation in autonomous driving systems.

2 Related Work

2.1 Video Diffusion for Autonomous Driving

The development of robust autonomous driving (AD) systems [15, 17, 6] necessitates large volumes of diverse and realistic training data. Learned generative model [22, 62, 11] have emerged as a powerful approach to synthesize such data by capturing the complex distributions of real-world driving scenarios. Early efforts explored Generative Adversarial Networks (GANs)[11, 38] and Variational Autoencoders (VAEs) [21] for generating driving-related imagery. More recently, diffusion models [12, 35, 44, 39, 59, 69, 68, 8], including latent diffusion models [44], have demonstrated state-of-the-art capabilities in generating high-fidelity images and videos [44, 4, 3, 77, 51, 54, 55, 75, 34]. Based on this technique, generative models in autonomous driving have greatly advanced. For example, BEVGen [46] and BEVControl [64] generate controllable street-view imagery from Bird’s-Eye View (BEV) layouts or other structural conditions. Further advancements, such as DriveDreamer [53], Magicdrive [10], DriveWM [56], Panacea [57], DriveSphere [61] and others [76, 30, 31, 58, 18, 23, 32, 73, 72], focus on generating coherent multi-camera driving videos, often conditioned on textual prompts, historical trajectories, or 3D assets. These models excel at producing visually compelling outputs that achieve high scores on perceptual metrics like Fréchet Video Distance (FVD) [49].

Despite achieving high visual realism, leading AD video generation models [57, 9, 18] frequently introduce subtle geometric and scene-level distortions (e.g., flawed perspectives, depth errors, unrealistic motion). These inconsistencies, often overlooked by human assessment, severely undermine performance on 3D perception tasks like object detection [26] and motion forecasting. RLGF addresses this critical gap by proposing a novel framework to instill multi-scale geometric and scene-level consistency within the generation process.

2.2 Reinforcement Learning for Video Generation

Fine-tuning generative models with reward signals, rooted in Reinforcement Learning from Human Feedback (RLHF) successes in LLMs (e.g., using PPO [45] or DPO [41]), is increasingly applied to diffusion models. For image generation, methods [2, 50, 63, 60] align models with human preferences, and approaches [25] use perception model feedback for content control. This trend extends to video diffusion, where DPO-based techniques [66, 74, 71] often utilize human preference data. While improving general video quality, these holistic preference scores typically lack the precise, local geometric feedback crucial for autonomous driving applications. Other methods [1, 24, 33, 7] employ explicit reward models for video fine-tuning, offering more detailed signals. However, these rewards are generally not tailored to the specific multi-scale 3D geometric and physical plausibility demands of AD simulation, failing to adequately address distortions like incorrect perspective or depth error.

Our RLGF framework advances this by introducing explicit, quantifiable geometric and scene-level rewards from dedicated AD perception models. This provides targeted, locally-aware feedback to correct specific geometric inaccuracies, producing data suitable for rigorous AD tasks.

3 Method

3.1 Preliminary: Limitations of Conditional Video Diffusion

Current video diffusion models [3, 57, 18] generate frames by gradually denoising latent variables through a Markov chain. Formally, given a condition c (e.g., road sketches or bounding boxes), the model learns to minimize the pixel-level reconstruction error during training:

$$\mathcal{L}_{\text{pixel}} = \mathbb{E}_{x_0, \epsilon, t} [\|\epsilon - \epsilon_\theta(x_t, t, c)\|^2], \quad (1)$$

where x_t is the noisy sample at timestep t , ϵ is the ground truth noise, and ϵ_θ is the denoising network. While effective for visual fidelity, this standard formulation inherently ignores crucial geometric-semantic correlations due to two limitations: (1) Pixel-wise Independence Assumption: The MSE loss treats each pixel as independent, failing to model higher-order geometric relationships (e.g., perspective consistency in 3D space). (2) Conditional Oversimplification: Existing methods concatenate geometric conditions c with noisy latents. However, this primarily enforces local pixel alignment corresponding to the condition rather than guaranteeing the global geometric integrity or plausibility of the underlying 3D scene structure. These limitations motivate the need for explicit geometric guidance during generation.

3.2 Reinforced Learning with Geometric Feedback (RLGF)

Our primary objective is to enhance the geometric and spatio-temporal consistency of videos generated by conditional diffusion models, addressing the critical gap left by conventional training objectives that primarily focus on pixel-level visual fidelity.

To achieve this, we introduce Reinforcement Learning with Geometric Feedback (RLGF). This framework refines a pre-trained video diffusion model, ϵ_θ , by guiding its generation process towards outputs that exhibit greater adherence to real-world geometric and physical principles. Unlike RLHF [37, 63], which typically captures broad subjective preferences, RLGF is designed to incorporate specific, model-interpretable geometric constraints. The core idea is to leverage dedicated, pre-trained perception models to evaluate the generated videos x_0 and provide a reward signal that steers ϵ_θ away from implausible outcomes.

Formally, given a well-trained video diffusion model ϵ_θ , the dataset D_v , the well-designed reward function $R(\cdot)$, we aim to maximize the objective:

$$J(\theta) = \mathbb{E}_{c, v, x_0} [R(x_0, v)] \quad (2)$$

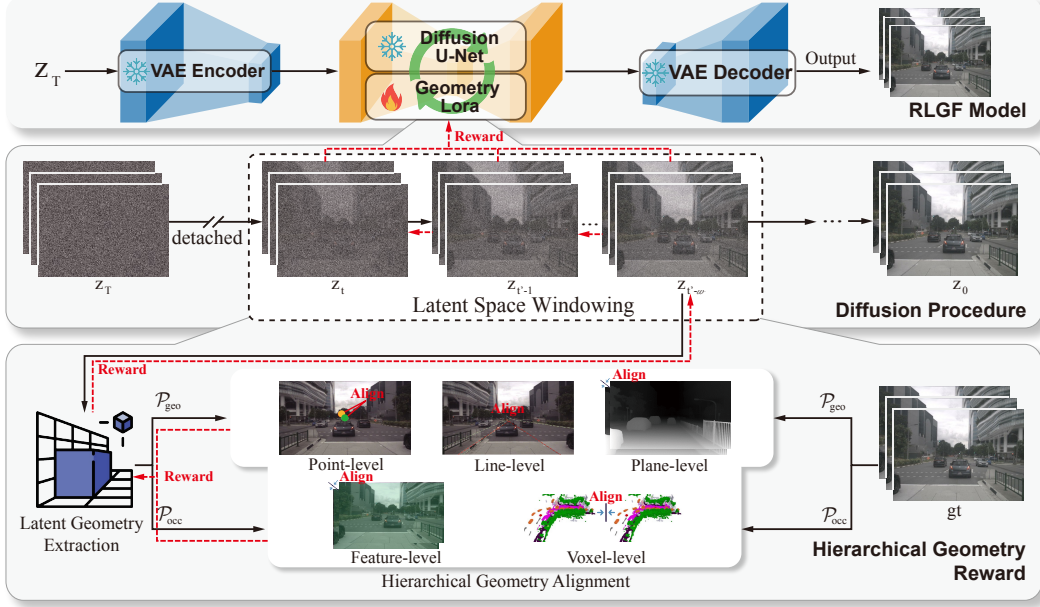


Figure 2: **Overview of the Reinforcement Learning with Geometric Feedback (RLGF) framework.** RLGF fine-tunes a frozen well-trained diffusion model via LoRA using rewards from a "Latent Space Windowing" scheme. Within this window, intermediate latents $z_{t'-w}$ are evaluated by frozen perception models $\mathcal{P}_{geo}$ (point-line-plane alignment) and $\mathcal{P}_{occ}$ (scene-level consistency) against a reference video. The resulting rewards (R_{geo}, R_{occ}) generate gradients (red arrows) to update LoRA, improving geometric and temporal consistency. Black arrows: feed forward; dashed red: gradients.

where c and v denote the condition and real video from the dataset D_v , x_0 the the generated sample by $\epsilon_\theta(\cdot|c)$ given c , and the total reward $R(x_0, v)$ encapsulates assessments from our Hierarchical Geometric Alignment system, as detailed in Section 3.4.

3.3 Latent-Space Windowing RL Optimization

Directly optimizing Equation 2 by unrolling the entire T -step diffusion sampling process to obtain x_0 and then backpropagating the reward gradient $\nabla_\theta R(x_0, v)$ is computationally prohibitive due to large memory requirements and long sampling chains.

To efficiently apply RLGF, we propose Latent-Space Windowed RL Optimization. This strategy addresses the computational challenge of full rollouts and leverages insights into the progressive nature of structure formation in diffusion models.

Visualizations (e.g., fig. 2(c) in Introduction) suggest that coarse global geometry is often established in earlier denoising steps (e.g. $t > T_{mid}$), while later steps refine details. Training across the entire sampling chain, as in some prior RL-diffusion work [1], may not provide sufficiently targeted guidance for these distinct phases.

Instead of rewarding only the final output x_0 , we provide feedback based on noisy latent features z_t at intermediate denoising steps t , compared against reference latents z_v derived from real videos. Specifically, during the T -step sampling process $z_T \rightarrow \dots \rightarrow z_0$, we apply our reward functions R within a sliding window of w steps (e.g. from random start step t' down to $k = t' - w$). Our perception models are designed to take z_k and k as input.

This approach is motivated by two factors: (1) Efficiency: It significantly reduces the computational graph for backpropagation. (2) Effectiveness: It allows targeted corrective signals at different stages of generation, crucial for both global structure and local detail.

The practical objective then becomes maximizing the expected reward obtained by evaluating the noisy latent z_k (at step k within the window) against the VAE-encoded latent z_v paired with real

video v :

$$J_{practical}(\theta_{LoRA}) = \mathbb{E}_{c,v,z_k} [R(z_k, z_v)] \quad (3)$$

where z_v is the encoded latent feature using VAE encoder [44]. The gradient with respect to the LoRA parameters θ_{LoRA} for a reward at step k is then:

$$\nabla_{\theta_{LoRA}} R_k(z_k, z_v) = \frac{\partial R_k(z_k, z_v)}{\partial z_k} \cdot \frac{\partial z_k}{\partial \theta_{LoRA}} \quad (4)$$

Throughout the RLGF fine-tuning, the perception models that constitute the reward function are pre-trained and their weights remain frozen, ensuring a consistent evaluation standard as ϵ_θ via θ_{LoRA} adapts to maximize geometric fidelity.

3.4 Hierarchical Geometric Reward (HGR)

The Hierarchical Geometric Reward (R) is designed to provide comprehensive, multi-level feedback on the geometric integrity and scene coherence of generated video latents. It evaluates consistency across point, line, plane, perceptual feature, and voxel-level representations, derived by applying specialized perception models to both the generated latent z_k and the reference real-video latent z_v .

3.4.1 Efficient Latent-Space Perception

To avoid computationally expensive full VAE decoding at each step k for perception, our perception models operate directly on latent features. Given a noisy video latent $z_k \in \mathbb{R}^{L \times C \times H' \times W'}$ (where L is frames, C channels, H', W' latent dimensions) at diffusion step k , we first employ a lightweight Micro-Decode Module, $\mathcal{F}_{micro}$. This module, constructed using shallow layers from the VAE decoder, processes z_k and k to produce enhanced per-frame features:

$$\mathbf{f}_k^f = \mathcal{F}_{micro}(z_k^f, k) \quad (5)$$

where k is processed by the Fourier Embedding [77, 48]. $\mathbf{f}_k^f$ is suitable for downstream perception tasks, significantly reducing memory and computation. The same $\mathcal{F}_{micro}$ is applied to z_v (with $k = 0$) to obtain reference features $\mathbf{f}_v^f$.

3.4.2 Perception Model Architectures

We train two models that take these micro-decoded features $\{\mathbf{f}_k^f\}$ or $\{\mathbf{f}_v^f\}$ and timestep k as input.

Latent Geometry Perception Model ($\mathcal{P}_{geo}$): This multi-task model processes per-frame features $\mathbf{f}^f$ to assess static 2.5D geometry. It uses a DINOv2 [36] backbone followed by task-specific heads:

(1) For **vanishing point detection**, we reformulate the task as heatmap regression to better capture positional uncertainty, following [13]. The network head takes $\mathbf{g}$ as input and predicts a probability heatmap $H \in \mathbb{R}^{h \times w}$. The ground truth heatmap H^{gt} is constructed as a 2D Gaussian centered at the annotated VP location, balancing localization precision and learning difficulty. We optimize using MSE:

$$\mathcal{L}_{vp} = \|H - H^{gt}\|^2 \quad (6)$$

(2) The **lane parsing** branch complements this by detecting road markings through a topology-aware segmentation head, following [42].

(3) As for the **depth estimation** task, we fine-tune the weight of [65] using a scale-invariant logarithmic (SiLog) loss:

$$\mathcal{L}_{depth} = \frac{1}{n} \sum_i d_i^2 - \frac{\lambda^2}{n} \left(\sum_i d_i \right)^2 \quad (7)$$

where $d_i = \log y_i - \log y_i^{gt}$, y_i is the predicted results and $\lambda \in [0, 1]$

Finally, we optimize $\mathcal{P}_{geo}$ using the total loss $\mathcal{L}_{geo} = \mathcal{L}_{vp} + \mathcal{L}_{lane} + \mathcal{L}_{depth}$.

Latent Occupancy Prediction Model ($\mathcal{P}_{occ}$): This model processes per-frame features $\mathbf{f}^f$ using an image backbone followed by a prediction head to infer the intermediate backbone features $\mathbf{feat}_{occ}$ and the 3D occupancy grid $O^i \in \mathbb{R}^{X \times Y \times Z}$ representing the occupied space around the ego vehicle for that frame. $\mathcal{P}_{occ}$ is trained based on [16].

3.4.3 Multi-Granularity Reward Signals

The hierarchical geometric reward $R = R_{geo} + R_{occ}$ quantifies geometric fidelity and scene coherence through a multi-scale decomposition.

For R_{geo} , it addresses vanishing point consistency, lane topology validity, and depth coherence. Given the outputs $\{p_{vp}, L_{pred}, D_{pred}\}$ from $\mathcal{P}_{geo}(z_k)$ and reference conditions $\{v_{ref}, L_{ref}, D_{ref}\}$ from $\mathcal{P}_{geo}(z_v)$, we define:

$$R_{geo}(z_k, z_v) = \underbrace{\lambda_{vp} r_{vp}(p_{vp}, v_{ref})}_{\text{Point-level}} + \underbrace{\lambda_{lane} r_{lane}(L_{pred}, L_{ref})}_{\text{Line-level}} + \underbrace{\lambda_{depth} r_{depth}(D_{pred}, D_{ref})}_{\text{Plane-level}} \quad (8)$$

where p_{vp}, p_{vp}, p_{vp} are the predictions from $\mathcal{P}_{geo}$ for vanishing point, lanes and depth, respectively. $\lambda_{vp}, \lambda_{vp}$ and λ_{vp} weights balancing the contribution of each task. The individual reward functions $r(\cdot)$ are designed to be high for geometrically accurate samples and low (or negative) for inconsistent ones:

$$r_{vp}(p_{vp}, v_{ref}) = -\|p_{vp} - v_{ref}\|_2^2 \quad (9)$$

where v_{ref} is the ground-truth vanishing point in conditions.

$$r_{lane} = \text{F1-Score}(L_{pred}, L_{ref}) \quad (10)$$

where F1-Score indicates the F1-Scores.

$$r_{depth} = -(\sqrt{(D_{pred} \odot M_{road} - D_{ref} \odot M_{road})^2} + \sqrt{(D_{pred} \odot M_{vehicle} - D_{ref} \odot M_{vehicle})^2}) \quad (11)$$

where pixel masks M_{road} and $M_{vehicle}$ are generated from real data and identify road and vehicle regions, respectively.

After that, we define $R_{occ}(z_k, z_v) = r_{align} + r_{iou}$. r_{align} encourages similarity in high-level scene interpretation by aligning distributions of intermediate occupancy features. For each frame f :

$$r_{align_f} = -D_{KL}(p(\text{feat}_{occ,f}^{real}) \| p(\text{feat}_{occ,f}^{gen})), \quad (12)$$

where $\text{feat}_{occ,f}$ are backbone features from $\mathcal{P}_{occ}$. Distributions $p(\cdot)$ can be estimated (e.g., Gaussian fit over a batch).

r_{occ} promotes accurate 3D structure and object layout. For each frame f :

$$r_{iou_f} = \text{IoU}(O_f^{gen}, O_f^{real}) = \frac{|O_f^{gen} \cap O_f^{real}|}{|O_f^{gen} \cup O_f^{real}|}, \quad (13)$$

where O_f are the 3D occupancy grids from $\mathcal{P}_{occ}$.

4 Experiments

4.1 Experimental Setup

We conduct comprehensive experiments to validate the efficacy of our proposed Reinforcement Learning with Geometric Feedback (RLGF) framework. This section details the datasets, baseline models, evaluation metrics, and implementation specifics used in our evaluation.

Datasets. Our experiments are primarily conducted on nuScenes [5] using the official validation split. Its multi-camera setup and comprehensive annotations, including 3D object labels and HD maps, provide a challenging benchmark. For this dataset, conditions c for the diffusion models are extracted from ground truth annotations, simulating realistic inputs for controllable generation.

Baselines. We demonstrate the plug-and-play nature of RLGF by integrating it with two representative video diffusion models, referred to as MagicDrive-V2 [9] and DiVE [18].

Evaluation Metrics. GeoScores: We use our proposed GeoScores suite: Vanishing Point Error: NormDist between the predicted VP and the pseudo-ground truth VP derived from real data. Lower is better. Lane Topology Score: F1-score for semantic segmentation of lane markings against ground

Table 1: **Performance of our Latent Geometry Perception Model ($\mathcal{P}_{geo}$) on the nuScenes validation split.** $\mathcal{P}_{geo}$ operates directly on micro-decoded latent features and is compared against representative pixel-space baselines.

Task	Metric	Model / Method	Input Space	Performance
VP Detection	NormDist ↓	URVP [29]	Pixel	0.045
		VPD [14]	Pixel	0.032
		$\mathcal{P}_{geo}$	Latent	0.024
Lane Parsing	F1-Score ↑	LaneATT [47]	Pixel	0.822
		PriorLane [40]	Pixel	0.879
		$\mathcal{P}_{geo}$	Latent	0.865
Depth Estimation	RMSE ↓	DepthAnything-v2 [65]	Pixel	1.798
		$\mathcal{P}_{geo}$	Latent	2.596

Table 2: **Comparison with state-of-the-art video generation methods on the nuScenes validation set.** We evaluate visual quality (FVD), 3D Object Detection (3DOD) performance, and geometric fidelity (GeoScores).

Methods	Quality	3DOD		GeoScore		
	FVD	mAP	NDS	VP	Lane	Depth
Real Data	-	35.53	41.20	-	-	-
Panacea [57]	139.0	11.58	22.31	-	-	-
Drive-WM [57]	122.7	20.66	-	-	-	-
MagicDrive-v2 [18]	101.2	18.95	21.10	0.092	0.787	1.732
DiVE [18]	68.4	25.75	33.61	0.086	0.792	1.822
MagicDrive-v2+Ours	99.8	23.21	27.80	0.079	0.854	0.983
DiVE+Ours	67.6	31.42	36.07	0.068	0.879	0.772

truth lane masks. Higher is better. Depth Error: We use RMSE between predicted depth for road surface regions and pseudo-ground truth depth. Lower is better.

Downstream Task Performance: 3D Object Detection: We employ a strong BEV-based detector, BEVFusion [26], trained solely on synthetic data generated by different methods or on real data. We report the standard nuScenes detection score (NDS) and mean Average Precision (mAP).

Visual Realism: We report Fréchet Video Distance (FVD) [49] to ensure RLGF does not degrade the visual quality achieved by the baseline diffusion models.

Implementation Details are included in the supplemental materials.

4.2 Performance of Pre-trained Perception Models

We first verify the capabilities of our perception models ($\mathcal{P}_{geo}$ and $\mathcal{P}_{occ}$ which form the basis of our reward functions. These models operate directly on latent features z_t and timestep t . Our goal here is to demonstrate their competence in extracting meaningful geometric and scene information from the latent domain.

Latent Geometry Perception Model ($\mathcal{P}_{geo}$) is a multi-task model responsible for assessing fine-grained geometric properties from latent features. Table 1 presents its performance on the nuScenes validation split for vanishing point (VP) detection, lane parsing, and depth estimation, compared against established pixel-space methods. The results indicate that $\mathcal{P}_{geo}$ effectively captures these geometric cues from latent features. For instance, it achieves a normalized distance error of 0.024 for VP detection and an F1-score of 0.865 for lane parsing. Although its depth estimation RMSE (2.596) is higher than the specialized pixel space model DepthAnything-v2, it still provides a consistent measure of depth relationships.

Latent Occupancy Model ($\mathcal{P}_{occ}$) is tasked with understanding 3D scene layout from sequences of latent features. Its performance on the Occ3D-nuScenes is shown in Table 3. $\mathcal{P}_{occ}$ achieves an overall mIoU of 29.96 when operating from latent representations. This level of performance, compared to a pixel-based method like FlashOcc (32.08 mIoU), demonstrates a solid capability to infer volumetric scene structure from the latent domain.

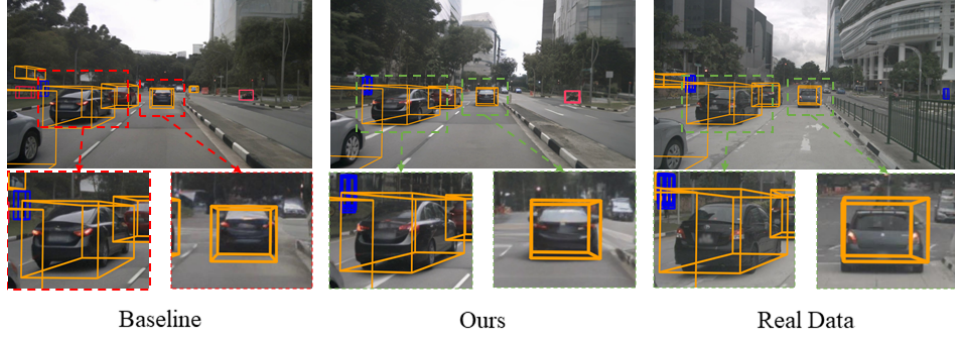


Figure 3: **Qualitative comparison of 3D object bounding box alignment.** RLGF-enhanced video exhibits much-improved 3D box alignment, closely matching the geometry implied by the scene.

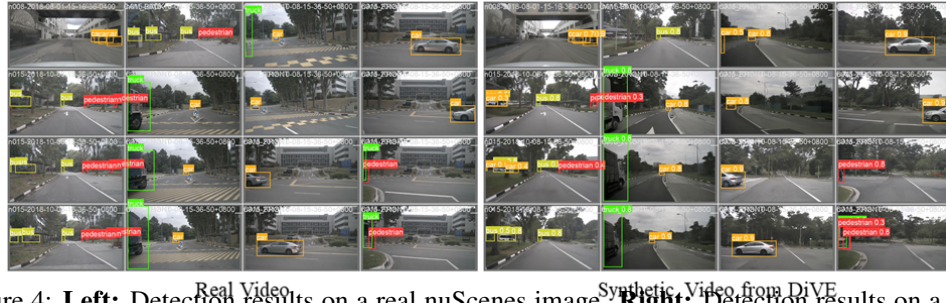


Figure 4: **Left:** Detection results on a real nuScenes image. **Right:** Detection results on a corresponding synthetic image generated by the DiVE baseline. Bounding boxes indicate detected objects (primarily vehicles).

4.3 Main Results: Improving Geometric Fidelity and Downstream Tasks

Table 2 shows RLGF’s impact on the nuScenes validation set. RLGF consistently enhances geometric integrity (GeoScores) across baselines while maintaining or improving visual quality (FVD). For instance, DiVE + RLGF significantly improves all GeoScore components (VP error: 0.086 $\rightarrow$ 0.068; Lane F1: 0.792 $\rightarrow$ 0.879; Depth RMSE: 1.822 $\rightarrow$ 0.772).

Crucially, this geometric enhancement translates to substantial 3DOD gains: DiVE + RLGF boosts mAP from 25.75 to 31.42 and NDS from 33.61 to 36.07, markedly closing the gap to real data performance. Similar improvements are seen for MagicDrive-v2 + RLGF (mAP: 18.95 $\rightarrow$ 23.21). These results underscore RLGF’s effectiveness as a plug-and-play module for improving both geometric soundness and downstream utility of synthetic videos. We also present the qualitative result in fig. 3 about the detection results on our synthetic data and baseline [18] and the real data. (More qualitative visualization results and further analysis are included in the supplementary materials).

4.4 2D Object Detection Results

To illustrate that current diffusion models like DiVE can generate visually realistic data with minimal 2D domain gap for certain tasks, we present qualitative 2D object detection results. Figure 4 shows outputs from a YOLOv5 [20] detector applied to (a) real nuScenes data and (b) synthetic data generated by the DiVE baseline. The detector is pre-trained on a large-scale dataset (e.g., COCO) and then fine-tuned on real nuScenes training data.

As observed in Figure 4, the 2D detection performance on DiVE-generated synthetic data is qualitatively very similar to that on real data. Objects are generally detected with comparable confidence and bounding box accuracy. This visual consistency aligns with our quantitative findings (mAP: 43.8 on synthetic vs. 44.7 on real, as mentioned in the Introduction), suggesting that the semantic content and 2D appearance features necessary for 2D detection are well-preserved in the synthetic videos. This further reinforces our hypothesis that the primary limitation of such synthetic data lies in its 3D geometric fidelity, which is specifically addressed by our RLGF framework.

Table 3: **Performance of our Latent Occupancy Prediction Model ($\mathcal{P}_{occ}$) on the Occ3D-nuScenes.** $\mathcal{P}_{occ}$ predicts 3D occupancy grids from sequences of micro-decoded latent features. **Vehicle** and **Dri. Sur** indicate IoU of vehicles and driving surface.

Method	Vehicle	Dri. Sur	mIoU $\uparrow$
FlashOcc [70]	43.2	72.2	32.08
$\mathcal{P}_{occ}$	37.9	65.7	29.96

Table 4: **Ablation study of HGA reward components within RLGF on nuScenes.**

ID	HGA Rewards					3DOD	
	r_{vp}	r_{lane}	r_{depth}	r_{align}	r_{iou}	mAP $\uparrow$	NDS $\uparrow$
DiVE [18]						25.75	33.61
1	✓					26.31	33.66
2	✓	✓				26.93	33.98
3	✓	✓	✓			27.12	34.82
4				✓	✓	28.06	35.11
Full	✓	✓	✓	✓	✓	31.42	36.07

Table 5: **Ablation studies on key RLGF hyperparameters.**

Hyperparameter	Value	3D Detection mAP
Window Size (w)	3	30.89
	5 (Ours)	31.42
	8	31.25
Reward Weights (λ)	Equal Weights (all 0.2)	30.76
	Balanced (Ours)	31.42
Window Range Position (t')	Early (Random in [20, 30])	30.55
	Mid (Random in [8, 30]) (Ours)	31.42
	Late (Random in [1, 15])	29.91

4.5 Ablation Study

4.5.1 Ablations about HGA Reward Components

We conduct ablation studies on nuScenes using DiVE as the baseline to understand the contributions of different HGA reward components. Results are shown in Table 4. Incrementally adding reward components ($r_{vp}, r_{lane}, r_{depth}$, and occupancy rewards ($r_{align} + r_{iou}$) progressively improves 3DOD performance (Table 4). For example, adding point-line-plane rewards (ID: 3) boosts DiVE’s mAP to 27.12. Incorporating only occupancy rewards (ID: 4) yields 28.06 mAP. The full HGA system, combining all five components, achieves the highest mAP (31.42) and NDS (36.07), highlighting the synergistic benefits of our comprehensive multi-level feedback. (More qualitative visualization results and further analysis are included in the supplementary materials).

4.5.2 Ablations about Hyperparameters.

We provide the detailed ablation results in table 5 for clarity, including the RL window size, reward weights, and the starting step of the sliding window. All ablations are performed on the DiVE baseline and evaluated with BEVFusion.

5 Conclusion

In this work, we addressed the critical issue of geometric distortions in diffusion-based video generation for autonomous driving. We introduced GeoScores for quantitative evaluation and proposed Reinforcement Learning with Geometric Feedback (RLGF), a novel framework to enhance the geometric integrity of synthetic videos. RLGF, through its Hierarchical Geometric Alignment (HGA) module which incorporates multi-level geometric and scene occupancy feedback, effectively injects perception-driven constraints into the generation process by fine-tuning pre-trained diffusion models. Our experiments demonstrate that RLGF significantly improves geometric fidelity across multiple baselines and, crucially, boosts downstream 3D object detection performance by up to , substantially closing the gap with real-data performance. This work establishes a new direction for generating more reliable and task-aware synthetic data for autonomous systems.

Acknowledgments

This work was supported by the Science and Technology Development Fund of Macau SAR (FDCT) under grants 0102/2023/RIA2 and 0154/2022/A3 and 001/2024/SKL, and the Jiangyin Hi-tech Industrial Development Zone under the Taihu Innovation Scheme (EF2025-00003-SKL-IOTSC).

References

- [1] Alexander Black, Simon Jenni, Tu Bui, Md Mehrab Tanjim, Stefano Petrangeli, Ritwik Sinha, Viswanathan Swaminathan, and John Collomosse. Vader: video alignment differencing and retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22357–22367, 2023.
- [2] Kevin Black, Michael Janner, Yilun Du, Ilya Kostrikov, and Sergey Levine. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023.
- [3] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.
- [4] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023.
- [5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.
- [6] Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024.
- [7] Zhangquan Chen, Xufang Luo, and Dongsheng Li. Visrl: Intention-driven visual perception via reinforced reasoning. *arXiv preprint arXiv:2503.07523*, 2025.
- [8] Zeyu Dong, Yuyang Yin, Yuqi Li, Eric Li, Hao-Xiang Guo, and Yikai Wang. Panolora: Bridging perspective and panoramic video generation with lora adaptation. *arXiv preprint arXiv:2509.11092*, 2025.
- [9] Ruiyuan Gao, Kai Chen, Bo Xiao, Lanqing Hong, Zhenguo Li, and Qiang Xu. Magicdrivedit: High-resolution long video generation for autonomous driving with adaptive control. *arXiv preprint arXiv:2411.13807*, 2024.
- [10] Ruiyuan Gao, Kai Chen, Enze Xie, Lanqing Hong, Zhenguo Li, Dit-Yan Yeung, and Qiang Xu. Magicdrive: Street view generation with diverse 3d geometry control. *arXiv preprint arXiv:2310.02601*, 2023.
- [11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.
- [12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.
- [13] Hiroto Honda, Motoki Kimura, Takumi Karasawa, and Yusuke Uchida. End-to-end monocular vanishing point detection exploiting lane annotations. *arXiv preprint arXiv:2108.13699*, 2021.
- [14] Hiroto Honda, Motoki Kimura, Takumi Karasawa, and Yusuke Uchida. End-to-end monocular vanishing point detection exploiting lane annotations. *arXiv preprint arXiv:2108.13699*, 2021.

- [15] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17853–17862, 2023.
- [16] Yuanhui Huang, Amonnut Thammatadatrakoon, Wenzhao Zheng, Yunpeng Zhang, Dalong Du, and Jiwen Lu. Probabilistic gaussian superposition for efficient 3d occupancy prediction. *arXiv preprint arXiv:2412.04384*, 2024.
- [17] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023.
- [18] Junpeng Jiang, Gangyi Hong, Miao Zhang, Hengtong Hu, Kun Zhan, Rui Shao, and Liqiang Nie. Dive: Efficient multi-view driving scenes generation based on video diffusion transformer. *arXiv preprint arXiv:2504.19614*, 2025.
- [19] Junpeng Jiang, Gangyi Hong, Miao Zhang, Hengtong Hu, Kun Zhan, Rui Shao, and Liqiang Nie. Dive: Efficient multi-view driving scenes generation based on video diffusion transformer. *arXiv preprint arXiv:2504.19614*, 2025.
- [20] Glenn Jocher, Alex Stoken, Jirka Borovec, Liu Changyu, Adam Hogan, Laurentiu Diaconu, Jake Poznanski, Lijun Yu, Prashant Rai, Russ Ferriday, et al. ultralytics/yolov5: v3. 0. *Zenodo*, 2020.
- [21] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013.
- [22] Lingdong Kong, Wesley Yang, Jianbiao Mei, Youquan Liu, Ao Liang, Dekai Zhu, Dongyue Lu, Wei Yin, Xiaotao Hu, Mingkai Jia, et al. 3d and 4d world modeling: A survey. *arXiv preprint arXiv:2509.07996*, 2025.
- [23] Bohan Li, Jiazhe Guo, Hongsi Liu, Yingshuang Zou, Yikang Ding, Xiwu Chen, Hu Zhu, Feiyang Tan, Chi Zhang, Tiancai Wang, et al. Uniscene: Unified occupancy-centric driving scene generation. *arXiv preprint arXiv:2412.05435*, 2024.
- [24] Jiachen Li, Weixi Feng, Tsu-Jui Fu, Xinyi Wang, Sugato Basu, Wenhui Chen, and William Yang Wang. T2v-turbo: Breaking the quality bottleneck of video consistency model with mixed reward feedback. *arXiv preprint arXiv:2405.18750*, 2024.
- [25] Ming Li, Taojiannan Yang, Huafeng Kuang, Jie Wu, Zhaoning Wang, Xuefeng Xiao, and Chen Chen. Controlnet++: Improving conditional controls with efficient consistency feedback: Project page: liming-ai. github. io/controlnet_plus_plus. In *European Conference on Computer Vision*, pages 129–147. Springer, 2024.
- [26] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. *Advances in Neural Information Processing Systems*, 35:10421–10434, 2022.
- [27] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.
- [28] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.
- [29] Yin-Bo Liu, Ming Zeng, and Qing-Hao Meng. Unstructured road vanishing point detection using convolutional neural networks and heatmap regression. *IEEE Transactions on Instrumentation and Measurement*, 70:1–8, 2020.
- [30] Enhui Ma, Lijun Zhou, Tao Tang, Zhan Zhang, Dong Han, Junpeng Jiang, Kun Zhan, Peng Jia, Xianpeng Lang, Haiyang Sun, et al. Unleashing generalization of end-to-end autonomous driving with controllable long video generation. *arXiv preprint arXiv:2406.01349*, 2024.
- [31] Jianbiao Mei, Tao Hu, Xuemeng Yang, Licheng Wen, Yu Yang, Tiantian Wei, Yukai Ma, Min Dou, Botian Shi, and Yong Liu. Dreamforge: Motion-aware autoregressive video generation for multi-view driving scenes. *arXiv preprint arXiv:2409.04003*, 2024.

- [32] Chaojun Ni, Xiaofeng Wang, Zheng Zhu, Weijie Wang, Haoyun Li, Guosheng Zhao, Jie Li, Wenkang Qin, Guan Huang, and Wenjun Mei. Wonderturbo: Generating interactive 3d world in 0.72 seconds. *arXiv preprint arXiv:2504.02261*, 2025.
- [33] Chaojun Ni, Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Wenkang Qin, Xinze Chen, Guanghong Jia, Guan Huang, and Wenjun Mei. Recondreamer-rl: Enhancing reinforcement learning via diffusion-based scene reconstruction. *arXiv preprint arXiv:2508.08170*, 2025.
- [34] Chaojun Ni, Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Wenkang Qin, Guan Huang, Chen Liu, Yuyin Chen, Yida Wang, Xueyang Zhang, et al. Recondreamer: Crafting world models for driving scene reconstruction via online restoration. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 1559–1569, 2025.
- [35] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021.
- [36] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.
- [37] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.
- [38] Yu Qian, Xunhao Li, Jian Zhang, Xiaolin Meng, Yongfu Li, Heng Ding, and Maoze Wang. A diffusion-tgan framework for spatio-temporal speed imputation and trajectory reconstruction. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–15, 2025.
- [39] Yu Qian, Jian Zhang, Zhanyu Feng, Xunhao Li, Zhiyuan Liu, and Hua Wang. Multi-task itransformer: A saturation-based model for short-term highway traffic congestion prediction considering event-induced capacity variability. *IEEE Transactions on Vehicular Technology*, pages 1–15, 2025.
- [40] Qibo Qiu, Haiming Gao, Wei Hua, Gang Huang, and Xiaofei He. Priorlane: A prior knowledge enhanced lane detection approach based on transformer. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5618–5624. IEEE, 2023.
- [41] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.
- [42] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.
- [43] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.
- [44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.
- [45] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

- [46] Alexander Swerdlow, Runsheng Xu, and Bolei Zhou. Street-view image generation from a bird’s-eye view layout. *IEEE Robotics and Automation Letters*, 2024.
- [47] Lucas Tabelini, Rodrigo Berriel, Thiago M. Paix ao, Claudine Badue, Alberto Ferreira De Souza, and Thiago Oliveira-Santos. Keep your Eyes on the Lane: Real-time Attention-guided Lane Detection. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.
- [48] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.
- [49] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Fvd: A new metric for video generation. 2019.
- [50] Bram Wallace, Meihua Dang, Rafael Rafailov, Linqi Zhou, Aaron Lou, Senthil Purushwalkam, Stefano Ermon, Caiming Xiong, Shafiq Joty, and Nikhil Naik. Diffusion model alignment using direct preference optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8228–8238, 2024.
- [51] Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghai Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wenten Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.
- [52] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. *arXiv preprint arXiv:2303.11926*, 2023.
- [53] Xiaofeng Wang, Zheng Zhu, Guan Huang, Xinze Chen, Jiagang Zhu, and Jiwen Lu. Drive-dreamer: Towards real-world-drive world models for autonomous driving. In *European Conference on Computer Vision*, pages 55–72. Springer, 2024.
- [54] Yujia Wang, Fang-Lue Zhang, and Neil A Dodgson. Scantd: 360° scanpath prediction based on time-series diffusion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7764–7773, 2024.
- [55] Yujia Wang, Fang-Lue Zhang, and Neil A Dodgson. Target scanpath-guided 360-degree image enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 8169–8177, 2025.
- [56] Yuqi Wang, Jiawei He, Lue Fan, Hongxin Li, Yuntao Chen, and Zhaoxiang Zhang. Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14749–14759, 2024.
- [57] Yuqing Wen, Yucheng Zhao, Yingfei Liu, Fan Jia, Yanhui Wang, Chong Luo, Chi Zhang, Tiancai Wang, Xiaoyan Sun, and Xiangyu Zhang. Panacea: Panoramic and controllable video generation for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6902–6912, 2024.
- [58] Yichen Xie, Chenfeng Xu, Chensheng Peng, Shuqi Zhao, Nhat Ho, Alexander T Pham, Mingyu Ding, Masayoshi Tomizuka, and Wei Zhan. X-drive: Cross-modality consistent multi-sensor data synthesis for driving scenarios. *arXiv preprint arXiv:2411.01123*, 2024.
- [59] Yi Xin, Qi Qin, Siqi Luo, Kaiwen Zhu, Juncheng Yan, Yan Tai, Jiayi Lei, Yuewen Cao, Keqi Wang, Yibin Wang, et al. Lumina-dimoo: An omni diffusion large language model for multi-modal generation and understanding. *arXiv preprint arXiv:2510.06308*, 2025.

- [60] Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. Imagereward: Learning and evaluating human preferences for text-to-image generation. *Advances in Neural Information Processing Systems*, 36:15903–15935, 2023.
- [61] Tianyi Yan, Dongming Wu, Wencheng Han, Junpeng Jiang, Xia Zhou, Kun Zhan, Cheng-zhong Xu, and Jianbing Shen. Drivingsphere: Building a high-fidelity 4d world for closed-loop simulation. *arXiv preprint arXiv:2411.11252*, 2024.
- [62] Tianyi Yan, Junbo Yin, Xianpeng Lang, Ruigang Yang, Cheng-Zhong Xu, and Jianbing Shen. Olidm: Object-aware lidar diffusion models for autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 9121–9129, 2025.
- [63] Kai Yang, Jian Tao, Jiafei Lyu, Chunjiang Ge, Jiaxin Chen, Weihang Shen, Xiaolong Zhu, and Xiu Li. Using human feedback to fine-tune diffusion models without any reward model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8941–8951, 2024.
- [64] Kairui Yang, Enhui Ma, Jibin Peng, Qing Guo, Di Lin, and Kaicheng Yu. Bevcontrol: Accurately controlling street-view elements with multi-perspective consistency via bev sketch layout. *arXiv preprint arXiv:2308.01661*, 2023.
- [65] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024.
- [66] Xiaomeng Yang, Zhiyu Tan, and Hao Li. Ipo: Iterative preference optimization for text-to-video generation. *arXiv preprint arXiv:2502.02088*, 2025.
- [67] Xuemeng Yang, Licheng Wen, Yukai Ma, Jianbiao Mei, Xin Li, Tiantian Wei, Wenjie Lei, Daocheng Fu, Pinlong Cai, Min Dou, et al. Drivearena: A closed-loop generative simulation platform for autonomous driving. *arXiv preprint arXiv:2408.00415*, 2024.
- [68] Zhongqi Yang, Wenhong Ge, Yuqi Li, Jiaqi Chen, Haoyuan Li, Mengyin An, Fei Kang, Hua Xue, Baixin Xu, Yuyang Yin, et al. Matrix-3d: Omnidirectional explorable 3d world generation. *arXiv preprint arXiv:2508.08086*, 2025.
- [69] Mingyang Yi, Aoxue Li, Yi Xin, and Zhenguo Li. Towards understanding the working mechanism of text-to-image diffusion model. *Advances in Neural Information Processing Systems*, 37:55342–55369, 2024.
- [70] Zichen Yu, Changyong Shu, Jiajun Deng, Kangjie Lu, Zongdai Liu, Jiangyong Yu, Dawei Yang, Hui Li, and Yan Chen. Flashocc: Fast and memory-efficient occupancy prediction via channel-to-height plugin. 2023.
- [71] Hangjie Yuan, Shiwei Zhang, Xiang Wang, Yujie Wei, Tao Feng, Yining Pan, Yingya Zhang, Ziwei Liu, Samuel Albanie, and Dong Ni. Instructvideo: Instructing video diffusion models with human feedback. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6463–6474, 2024.
- [72] Shuang Zeng, Xinyuan Chang, Xinran Liu, Zheng Pan, and Xing Wei. Driving with prior maps: Unified vector prior encoding for autonomous vehicle mapping. *arXiv preprint arXiv:2409.05352*, 2024.
- [73] Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, and Xing Wei. Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving. *arXiv preprint arXiv:2505.17685*, 2025.
- [74] Jiacheng Zhang, Jie Wu, Weifeng Chen, Yatai Ji, Xuefeng Xiao, Weilin Huang, and Kai Han. Onlinevpo: Align video diffusion model with online video-centric preference optimization. *arXiv preprint arXiv:2412.15159*, 2024.
- [75] Rongchao Zhang, Yu Huang, Yiwei Lou, Weiping Ding, Yongzhi Cao, and Hanpin Wang. Synergistic attention-guided cascaded graph diffusion model for complementarity determining region synthesis. *IEEE Trans. Neural Networks Learn. Syst.*, 36(7):11875–11886, 2025.

- [76] Guosheng Zhao, Xiaofeng Wang, Zheng Zhu, Xinze Chen, Guan Huang, Xiaoyi Bao, and Xingang Wang. Drivedreamer-2: Llm-enhanced world models for diverse driving video generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 10412–10420, 2025.
- [77] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

Appendix

This supplementary material provides additional details to support the findings presented in our main paper. We include: (1) comprehensive implementation specifics for our RLGF framework and the pre-trained perception models; further details on the GeoScores metric computation; (2) additional experimental results (3) a discussion on the limitations of our current work and potential future directions.

A Detailed Implementation Details

This section elaborates on the implementation details of our proposed RLGF framework, the pre-trained perception models ($\mathcal{P}_{geo}$ and $\mathcal{P}_{occ}$), and the experimental setup.

A.1 Dataset Preparation

All experiments, including the pre-training of our perception models ($\mathcal{P}_{geo}$ and $\mathcal{P}_{occ}$) and the fine-tuning of diffusion models with RLGF, are conducted using the nuScenes dataset [5]. We primarily utilize the official training and validation splits. While nuScenes provides rich annotations like 3D bounding boxes and HD maps, it does not directly offer ground truth labels for vanishing points (VP), dense segmentation masks for all relevant classes (like fine-grained lanes beyond HD map polylines), or per-pixel depth maps required by our $\mathcal{P}_{geo}$. Therefore, we generate high-quality pseudo-labels for these tasks using strong, pre-existing perception models, as detailed below. These pseudo-labels serve as the training targets for our latent-space perception models.

Depth Pseudo-Labels: To obtain dense depth information for training the depth estimation component of $\mathcal{P}_{geo}$, we utilize Depth Anything V2 (vit-l version) [65]. This state-of-the-art monocular depth estimation model is applied to all images in the nuScenes training set to generate per-pixel depth maps. These output depth maps serve as the pseudo-ground truth for our latent depth estimation task.

Semantic Segmentation Pseudo-Labels (Lanes, Road Surface, Vehicles): For precise segmentation masks of various scene elements, we employ Grounded-SAM-2[43, 42]. For the lanes, the model is prompted to accurately segment visible lane markings. The resulting binary segmentation masks are used as pseudo-ground truth for training the lane parsing head of $\mathcal{P}_{geo}$. For the road surface and vehicle masks, SAM-2 is also utilized to generate segmentation masks for road surfaces and vehicles.

Vanishing Point Pseudo-Labels from Lane Masks, following [13]: With accurate lane segmentation masks obtained via SAM-2 (as described above), we derive vanishing point pseudo-labels through a geometric procedure. For each detected lane marking in a frame: The center point of the lane marking is calculated from its left and right edges (derived from the SAM-2 segmentation mask) for every horizontal image line at 5-pixel intervals. These extracted center points are grouped to represent the centerline of each individual lane marking. Robust curve fitting (e.g., RANSAC with a line model) is applied to these centerlines. The intersection point of multiple fitted lane centerlines is then computed to determine the scene’s vanishing point. This computed VP serves as the pseudo-ground truth for the VP detection task.

The use of these high-quality pseudo-labels enables us to train effective latent-space perception models tailored to the nuScenes domain, which subsequently provide the nuanced reward signals for our RLGF framework. The conditions c for the main diffusion models (e.g., semantic 3D boxes for some baselines) are derived from the original nuScenes ground truth annotations.

A.2 Perception Model Architectures and Pre-training

Micro-Decode Module($\mathcal{F}_{micro}$): The $\mathcal{F}_{micro}$ module is constructed using the first upper block of the official VAE decoder from the OpenSora [77] used by our baseline video diffusion models. The input of $\mathcal{F}_{micro}$ is the noisy latent feature z_k^f for a frame f with the timestep k . The same $\mathcal{F}_{micro}$ architecture is used when processing the reference real-video latent z_v (with k typically set to 0).

Latent Geometry Perception Model($\mathcal{P}_{geo}$): We use a pre-trained DINOv2-ViT-S/14 [36] as the backbone feature extractor and the pre-trained weight from DepthAnything-V2 [65]. $\mathcal{P}_{geo}$ is trained

for 50 epochs on the nuScenes training split using the AdamW optimizer with a learning rate of 5×10^{-5} and a batchsize of 16. We use $8 \times$ NVIDIA A100 GPUs to cover the experiment.

Latent Occupancy Prediction Model($\mathcal{P}_{occ}$): $\mathcal{P}_{occ}$ is trained on occ3D-nuscenes dataset for 24 epochs using AdamW optimizer following [70, 16].

A.3 RLGF Fine-tuning Details

Baselines: We used publicly available checkpoints for MagicDrive-V2 [9] and DiVE [18].

LoRA Configuration: For LoRA, we applied it to the attention layers (Q, K, V projections) of the DiT backbone in the diffusion models. We used a rank $r = 16$ following [1].

Latent-Space Windowed Optimization: The window size w is set to 5. The starting step t' for the window was randomly sampled from the range $[8, 30]$, with $T = 30$ is the total number of diffusion steps.

Reward Weights: We set $\lambda_{vp} = 0.1$, $\lambda_{lane} = 0.1$, $\lambda_{depth} = 0.5$. These weights were determined empirically based on early experiments on a small validation subset, aiming to balance the scale of individual reward components and their perceived impact on generation quality.

We use AdamW with a learning rate of 1×10^{-4} and a batchsize of 1 with 8 frames per video clip.

A.4 GeoScores Metric Details

This section provides further clarification on the computation of our GeoScores components. For all GeoScores, the "reference ground truth" is derived by applying the corresponding pre-trained perception model to the *real* video data, following appendix A.1. The score then measures the deviation of the synthetic video's perception output from this real-data-derived reference.

Vanishing Point Error (VP $\downarrow$): Calculated as the L2 Normalized Distance (NormDist) between the calculated VP on a synthetic frame and the VP calculated on a real frame. **Lane Topology Score (Lane $\uparrow$):** Calculated as the F1 score for semantic segmentation of lane markings. The predictions is from Grounded-SAM2 [43, 42] on the synthetic frame, and the target is applied to the real frame. **Depth Error (Depth $\downarrow$):** Calculated as the Root Mean Squared Error (RMSE) between the depth map predicted by Depth Anything V2 [65] for road surface regions on a synthetic frame and the depth map for the same regions on the real frame. Road surface masks are obtained from SAM-2 [42].

B Additional Experiment Results

B.1 Extended 3D Object Detection Results on Multiple Detectors

To further demonstrate the generalizability of the improvements conferred by RLGF, we evaluated the generated synthetic data using an additional state-of-the-art 3D object detector, StreamPETR [52], alongside the BEVFusion results presented in the main paper. Table 6 presents the performance (mAP and NDS on nuScenes validation) for StreamPETR and the average performance across both BEVFusion and StreamPETR. Both detectors were trained from scratch solely on the respective synthetic data or real data.

B.2 Robustness to Different Downstream Detectors:

To demonstrate generality, we evaluated our method on the stronger **StreamPETR** detector. RLGF achieved a substantial +4.75% mAP gain, proving our geometric improvements are robust and benefit diverse downstream architectures.

C Limitations and Future Work

This section discusses the current limitations of our RLGF framework and GeoScores metric, alongside potential avenues for future research.

Dependence on Perception Models: RLGF's performance is inherently tied to the accuracy and robustness of the pre-trained perception models ($\mathcal{P}_{geo}, \mathcal{P}_{occ}$). Biases or errors in these models could

Table 6: Detailed 3D Object Detection (3DOD) performance on nuScenes validation using Stream-PETR [52]. RLGF is applied to MagicDrive-v2 and DiVE.

Methods	Quality	BevFusion		StreamPETR	
	FVD	mAP	NDS	mAP	NDS
Real Data	-	35.53	41.20	38.01	49.02
Panacea [57]	139.0	11.58	22.31	-	-
Drive-WM [57]	122.7	20.66	-	-	-
MagicDrive-v2 [18]	101.2	18.95	21.10	22.77	28.93
DiVE [18]	68.4	25.75	33.61	29.19	36.23
MagicDrive-v2+Ours	99.8	23.21	27.80	26.01	35.64
DiVE+Ours	67.6	31.42	36.07	33.94	39.68

Method	mAP (BEVFusion)
DiVE [16] (Baseline)	25.75
+ Detector Reward	26.51 (+0.76)
+ Ours (RLGF)	31.42 (+5.67)

Table 7: Comparison of reward signal effectiveness on the DiVE baseline. Our Hierarchical Geometric Reward (HGR) is significantly more effective than a high-level detector-based reward.

propagate into the reward signal and mislead the generation process. Future work could explore jointly training or adapting perception models during RLGF, or using ensembles.

Computational Cost: While Latent-Space Windowed Optimization significantly reduces costs compared to full rollouts, RL-based fine-tuning remains more computationally intensive than standard diffusion model training. Exploring more sample-efficient RL algorithms or distillation techniques could be beneficial.

Reward Design and Balancing: The current HGA reward combines five components with manually tuned weights. Optimizing these weights automatically or learning a more adaptive reward function is a promising direction. Furthermore, incorporating even more diverse geometric or physical constraints (e.g., collision avoidance, traffic rule adherence) could further enhance realism.

Generalization: While demonstrated on nuScenes, further investigation is needed to assess RLGF’s generalization capabilities across diverse datasets, environmental conditions (e.g., adverse weather, night scenes not well-represented in training), and different diffusion model architectures.

GeoScores Scope: Current GeoScores focus on camera-based geometric aspects. Expanding them to include LiDAR consistency or multi-modal geometric agreement could provide a more holistic evaluation.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification:

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: In the supplementary materials

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[NA\]](#)

Justification: not include theoretical results

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: Detailed implementation details are included in the supplementary materials.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We will open source the code once we are ready.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Details will be included in the included in the supplementary materials.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification:

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification:

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer:[NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional review board (IRB) approvals or equivalent for research with human subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. Declaration of LLM usage

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [No]

Justification: Only for editing.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.