Bridging Simulation and Reality: Geometry and Decision Alignment for Autonomous Driving

Dapeng Zhang[1] * **Zhenlong Yuan**[2] * **Chenyang Li**[3] **Hongtao Nie**[3]
Yinda Chen[4] **Rui Zhou**[3] **Peng Zhi**[5] ‡ **Qingguo Zhou**[3] ‡

[1] National University of Singapore [2] Dream-X Team [3] Lanzhou University
[4] University of Science and Technology of China [5] Shanghai Jiao Tong University

* *Equal contribution* ‡ *Corresponding Author*

Abstract

Bridging the gap between simulation and reality remains a fundamental challenge for end-to-end autonomous driving. Existing approaches primarily focus on appearance-level features. This often leads to suboptimal transfer, where visually aligned models still produce inconsistent or unsafe behaviors in the real world. In this paper, we propose a unified domain adaptation framework that jointly aligns perception and decision processes through geometry-aware and vector-space representations. At the perception level, we introduce a Geometry-Aware Perception Alignment (GAPA) module that enforces cross-domain consistency in both explicit geometry and implicit structural representations extracted from latent features. This reduces domain discrepancy by targeting scene structure rather than appearance. At the decision level, we propose a Latent Vector Space Guidance Decision Alignment (LVDA) module that aligns vector-conditioned policy distributions. By modeling driving decisions as conditioned on structured representations of map topology and multi-agent interactions, we minimize discrepancies between source and target action distributions via adversarial and structural alignment. To further enhance stability, we introduce a progressive adversarial transfer strategy that improves cross-domain feature alignment and training stability. Extensive experiments on a newly partitioned nuScenes dataset demonstrate that Bridge-AD achieves excellent performance, effectively narrowing the gap between simulation and real-world autonomous driving.

1 Introduction

Autonomous driving aims to develop intelligent systems capable of making safe, reliable, and adaptive decisions in complex and dynamic environments. Recent advances in end-to-end learning have demonstrated that driving policies can be learned directly from raw sensory inputs and expert demonstrations, bypassing hand-engineered modular pipelines. Despite these successes, a fundamental challenge remains: models trained in one domain often fail to generalize to another, particularly when transferring from simulation to the real world.

Simulation environments provide scalable, cost-effective, and safe platforms for generating large-scale training data, including rare and safety-critical scenarios that are difficult to capture in real-world datasets. In contrast, real-world data inherently captures authentic sensor noise, environmental variability, and complex vehicle dynamics. While both domains are indispensable, the discrepancy between them, commonly referred to as the simulation-to-reality (sim-to-real) gap, significantly

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

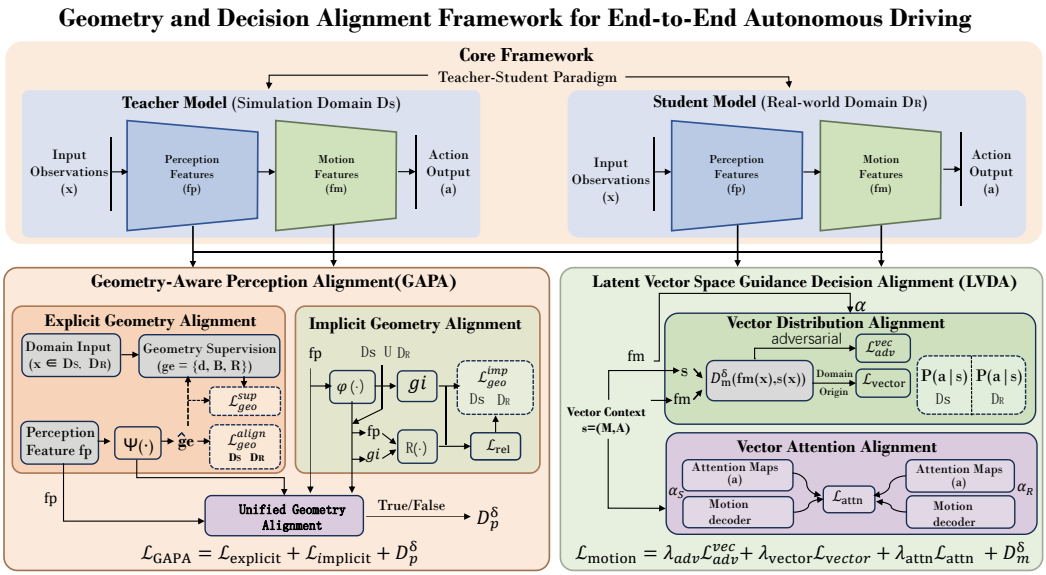


Figure 1: Overview of Bridge-AD for simulation-to-real autonomous driving. The framework performs domain-specific pre-training followed by adversarial alignment. GAPA aligns geometric representations at the perception level, while LVDA aligns vector-conditioned policy and attention at the decision level. Dual discriminators enforce domain invariance, enabling robust transfer from simulation to real-world driving.

limits the deployment of learned driving policies. Bridging this gap is therefore a central problem in autonomous driving research.

Existing domain adaptation methods predominantly focus on aligning appearance-level features between domains, such as textures, illumination, and style. However, such approaches often overlook two critical aspects. First, autonomous driving decisions fundamentally depend on the geometric structure of the scene, including 3D layout, object configurations, and spatial relationships, which are more invariant across domains than appearance. Second, driving policies operate over structured decision spaces, where vectorized representations of road topology and multi-agent interactions govern behavior. Misalignment at this decision level can lead to inconsistent or unsafe actions, even when perception features appear well-aligned.

To address these limitations, we propose **Bridge-AD**, a unified domain adaptation framework that explicitly aligns both perception and decision processes for end-to-end autonomous driving. Our approach is built upon two key components. First, we introduce a Geometry-Aware Perception Alignment (GAPA) module, which enforces cross-domain consistency in both explicit geometric representations (3D bounding boxes, spatial relations) and implicit structural features extracted from latent representations. By focusing on geometry rather than appearance, GAPA reduces domain discrepancy at a more fundamental level. Second, we propose a Latent Vector Space Guidance Decision Alignment (LVDA) module, which aligns vector-conditioned policy distributions across domains. Specifically, we model driving decisions as conditioned on structured vector representations capturing map topology and multi-agent interactions, and minimize discrepancies between source and target action distributions via adversarial learning. Additionally, we align attention distributions within the policy network to enforce consistent relational reasoning, further improving robustness under domain shift. To effectively integrate these components, Bridge-AD adopts a progressive adversarial transfer strategy. We first pre-train domain-specific models on simulation and real-world datasets, and then employ dual discriminators to guide cross-domain alignment at both perception and decision levels. A tailored backpropagation mechanism is introduced to stabilize training and enhance feature alignment across domains.

We validate our method on a newly partitioned nuScenes benchmark designed for sim-to-real transfer. Extensive experiments demonstrate that Bridge-AD significantly outperforms state-of-the-art domain adaptation approaches, achieving more robust and consistent driving behavior in real-world scenarios.

Our main contributions are summarized as follows:

- We propose Bridge-AD, a unified framework for sim-to-real transfer that jointly aligns perception and decision processes in end-to-end autonomous driving.
- We introduce Geometry-Aware Perception Alignment (GAPA), which enforces cross-domain consistency in both explicit and implicit geometric representations.
- We propose Latent Vector Space Guidance Decision Alignment (LVDA), which aligns vector-conditioned policy distributions and attention structures across domains.
- We develop a progressive adversarial transfer strategy with dual discriminators and a stabilized training mechanism for effective cross-domain learning, our method demonstrates state-of-the-art performance on nuScenes.

2 Related Works

2.1 End-to-End Autonomous Driving Model Trained on Real-World Scenario Data

Most industry solutions rely on real-world data for training, reducing the gap between training and deployment. Early work such as ALVINN [1] uses a three-layer network trained on camera data to predict driving direction. Some methods directly predict trajectories without explicit perception or motion prediction [2], which is less suitable for real-world settings. To address this, Casas et al. [3] improve planning by leveraging intermediate representations and forecasting dynamic agents. Later approaches enhance safety and performance through different strategies. LookOut [4] increases planner diversity, while others incorporate accumulated features or semantic occupancy supervision [5, 6]. UniAD [7] introduces a unified framework that jointly models perception, prediction, and planning via query-based interactions. Additionally, methods that integrate perception outputs with planning constraints show strong real-world performance [8–10]. More efficient designs have also been proposed. SparseDrive [11] replaces costly BEV representations with sparse feature sampling and adopts a parallel planning architecture with collision-aware rescoring. Similarly, MapExpert [12] employs a unified module with expert-specific designs for improved efficiency. Recently, vision-language model (VLM)-based approaches [13–17] have emerged, reframing autonomous driving as a vision-language task.

2.2 End-to-End Autonomous Driving Model Trained on Simulated Scenario Data

Collecting real-world data is costly and often lacks sufficient corner cases. To address this, simulation platforms such as CARLA [18] and LGSVL [19] are widely used for training and evaluation. For example, CILRS [20] introduces an auxiliary speed prediction task to assist control, while LbC [21] transfers privileged information from a teacher model to a vision-based student. To overcome limitations of camera-only inputs, multi-modal methods integrate additional sensors for improved spatial understanding [22, 23]. Building on prior work [8, 24], VADv2 [10] leverages large planning vocabularies and simulation data for trajectory prediction. Similarly, PlanT [25] adopts a transformer-based design for efficient and scalable planning, while other works refine transformer representations to enhance performance [26, 12]. ThinkTwice [27] further improves trajectory prediction by combining coarse future estimation with salient feature retrieval. Simulation has also accelerated reinforcement learning approaches, where teacher models are trained under diverse scenarios (e.g., extreme weather, accidents) to guide student models in handling corner cases [28–30]. Recently, LLM-based methods such as DriveGPT4 [31] and related works [32, 13, 33, 34] enable closed-loop driving by integrating perception and decision-making into a unified framework. LLM4AD [35] further introduces a benchmark to evaluate these approaches across diverse scenarios.

2.3 Domain Adversarial Transfer

Numerous studies demonstrate the effectiveness of transfer learning by examining the transferability of deep neural networks trained on public datasets [36]. Hendrycks et al. [37] investigates the transfer pre-training mechanism, showing that pre-training improves adversarial robustness by 10% over the base model. However, a major focus of transfer learning is unsupervised domain adaptation, which adapts a model trained on a labeled dataset to an unlabeled domain. Among these approaches, Long et al. [38] introduces marginal and conditional distributions to construct new feature representations

for unsupervised domains. MetaAlign [39] proposes an effective meta-optimization strategy that maximizes the neural network gradients across two tasks during training. Yang et al. [40] proposes an adaption transfer method to grasp novel target objects, this method leverages object attributes to facilitate robotic grasping and rapid adaptation to new domains.

3 Preliminaries: Discrepancy Between Simulation and Real-world

Let simulation and real-world datasets be denoted as D_S and D_R, δ is the learnable parameters. The hypothesis spaces are defined as:

$$H_s = \{h_s : D_S\}, \quad H_r = \{h_r : D_R\} \tag{1}$$

T_{s2r} means the distance between simulation and real-world domains.

$$h_r = T_{s2r} \cdot h_s \tag{2}$$

We define discrepancy as:

$$D_h^\delta(D_S||D_R), \quad D_h^\delta = D_p^\delta + D_m^\delta \tag{3}$$

D_p and D_m are perception and motion discrepancy. Existing discrepancy modeling focuses on feature alignment but ignores underlying geometric structure and decision-level.

4 Method

As depicted in Figure 1, we propose a novel domain transfer learning framework for end-to-end autonomous driving. Our method comprises three core modules: (1) an End-to-End Base Model following a teacher-student paradigm, (2) Geometry-Aware Perception Alignment (GAPA) to bridge the structural gap in scene understanding, and (3) Latent Vector Space Guidance Decision Alignment (LVDA) to ensure policy consistency in the vector space.

4.1 End-to-End Base Models

We formalize the problem as transferring a model from a teacher domain D_S (Simulation) to a student domain D_R (Real-world). We adopt a **Teacher-Student Framework** where both models share identical architectures (SparseDrive). **Teacher Model:** Pre-trained on D_S, providing dense supervision for safety-critical and long-tail scenarios. This model follows the end-to-end architecture to generate high-fidelity trajectories. **Student Model:** It is optimized on D_R. This model acts as the final deployable agent, learning to map real-world observations to driving actions while retaining the structural knowledge of the teacher.

4.2 Geometry-Aware Perception Alignment

A key limitation of transfer learning methods is that they primarily align appearance-level features while neglecting the underlying geometric structure of the scene. However, for autonomous driving, the decision process depends critically on geometry, such as 3D layout, object positions, and spatial relations, which are largely invariant across domains.

To address this, we propose a Geometry-Aware Perception Alignment (GAPA) module that enforces cross-domain consistency at the geometric level. Our approach consists of two complementary components: **Explicit Geometry-Aware Alignment**: leveraging structured geometric representations when supervision is available; **Implicit Geometry-Aware Alignment**: extracting geometry from latent representations.

4.2.1 Explicit Geometry-Aware Perception Alignment

In the simulation and real-world domain, we have access to explicit geometric attributes such as BEV coordinates d, 3D bounding boxes $\mathcal{B}$, and pairwise spatial relations $\mathcal{R}$. We define an explicit geometry representation as:

$$g^e(x) := \{d(x), \mathcal{B}(x), \mathcal{R}(x)\} \tag{4}$$

Let $f_p(x)$ is the latent features extracted from perception features with several transformer blocks. A geometry prediction head $\psi(\cdot)$ maps features to geometry:

$$\hat{g}^e(x) = \psi(f_p(x)) \tag{5}$$

For our datasets, geometry is available or can be generated accurately. We apply supervised learning:

$$\mathcal{L}_{geo}^{sup} = \mathbb{E}_{x\sim D_S}\left[\|\hat{g}^e(x) - g^e(x)\|\right] \tag{6}$$

To bridge the domain gap, we minimize the distribution divergence $\mathcal{D}$ between the projected geometric distributions of both domains:

$$\mathcal{L}_{geo}^{align} = \mathcal{D}\big(P_S(\hat{g}^e),\, P_R(\hat{g}^e)\big) \tag{7}$$

where $\mathcal{D}(\cdot,\cdot)$ is a distribution divergence, P_S and P_R are the probability measures.

Finally, we have

$$\mathcal{L}_{explicit} = \mathcal{L}_{geo}^{sup} + \lambda_{align}\mathcal{L}_{geo}^{align} \tag{8}$$

4.2.2 Implicit Geometry-Aware Perception Alignment

In real-world scenarios, explicit geometry annotations may be noisy. To alleviate this, we propose an implicit geometry alignment strategy.

We define an implicit geometry extractor:

$$g^i(x) = \phi(f_p(x)) \tag{9}$$

where $\phi(\cdot)$ extracts spatial structure from latent representations (transformer attention, feature correlations). Unlike explicit geometry, $g^i(x)$ is not directly supervised.

We assume that geometry corresponds to relational structure. Thus, we enforce consistency via pairwise relations:

$$\mathcal{L}_{rel} = \mathbb{E}_{x\sim D_S\cup D_R}\left[\|\mathcal{R}(f_p(x)) - \mathcal{R}(g^i(x))\|_2\right] \tag{10}$$

where $\mathcal{R}(\cdot)$ computes relational matrices use cosine similarity.

We align implicit geometry distributions:

$$\mathcal{L}_{geo}^{imp} = \mathcal{D}\big(P_S(g^i),\, P_R(g^i)\big) \tag{11}$$

$$\mathcal{L}_{implicit} = \lambda_{imp}\mathcal{L}_{geo}^{imp} + \lambda_{rel}\mathcal{L}_{rel} \tag{12}$$

4.2.3 Unified Geometry-Aware Alignment

We extend the perception discriminator to operate on joint representations (δ represents the learnable parameters of the discriminator):

$$D_p^{\delta}(f_p(x), \hat{g}^e(x), g^i(x)) \tag{13}$$

which enforces alignment in the joint space of appearance and geometry.

The final perception alignment objective combines both components:

$$\mathcal{L}_{GAPA} = \mathcal{L}_{explicit} + \mathcal{L}_{implicit} + D_p^{\delta} \tag{14}$$

We assume that the conditional label distribution depends on geometry:

$$P(y|x) = P(y|g(x)) \tag{15}$$

Under this assumption, domain shift arises from mismatch in geometry distribution:

$$P_S(g(x)) \neq P_R(g(x)) \tag{16}$$

By minimizing:

$$\mathcal{D}(P_S(g), P_R(g)) \to 0 \quad \Rightarrow \quad P_S(y|g) \approx P_R(y|g) \tag{17}$$

Aligning geometry is sufficient for reducing domain gap when decisions depend primarily on vector structure rather than appearance. Combine with the other label/distance related objective components ($\mathcal{L}_{geo}^{sup}, \mathcal{L}_{rel}, D_p^{\delta}$), we could optimize the total function.

4.3 Latent Vector Space Guidance Decision Alignment

While geometry-aware perception alignment enforce geometry-level discrepancy, a critical challenge remains at the **decision level**. In autonomous driving, policies must reason over vector decision context such as road topology and multi-agent interactions. Even with aligned perception features, mismatch in vector distributions can lead to inconsistent or unsafe planning behaviors. Thus, we propose a Latent Vector Space Guidance Decision Alignment framework, which enforces domain invariance at the policy level by aligning vector-conditioned decision distributions.

Let $x \in \mathcal{X}$ denote perception features, $y \in \mathcal{Y}$ denote trajectories, and $a \in \mathcal{A}$ denote policy outputs. We define a structured vector representation:

$$s(x) := \big(M(x), A(x)\big) \tag{18}$$

where $M(x)$ is the map representation (lane graph, topology), $A(x)$ denotes surrounding agent states (positions, velocities).

Let $f_m(x) \in \mathbb{R}^{d_m}$ denote motion features extracted from perception features, and let π_θ denote the policy parameterized by θ:

$$a = \pi_\theta\big(f_m(x), s(x)\big) \tag{19}$$

We assume that the optimal policy depends on x only through $s(x)$:

$$P(y \mid x) = P(y \mid s(x)) \tag{20}$$

This assumption reflects that driving decisions are governed by structured scene context rather than raw appearance.

The expected real-world domain risk is:

$$\epsilon_R(\pi_\theta) = \mathbb{E}_{(x,y)\sim D_R}\big[\ell(\pi_\theta(f_m(x), s(x)), y)\big] \tag{21}$$

where $\ell(\cdot,\cdot)$ is a cross-entropy loss. Following domain adaptation theory, for a policy class Π:

$$\epsilon_R(\pi_\theta) \leq \epsilon_S(\pi_\theta) + d_{\mathcal{S}}(\pi_\theta) + d_{\mathcal{X}}(P_S, P_R) + \lambda^* \tag{22}$$

$\epsilon_S(\pi_\theta)$ is the simulation domain risk, $\lambda^* = \min_{\pi\in\Pi}\big(\epsilon_S(\pi) + \epsilon_R(\pi)\big)$ is the optimal joint error, $d_{\mathcal{S}}(\pi_\theta)$ measures the discrepancy in conditional action distributions:

$$d_{\mathcal{S}}(\pi_\theta) = \mathbb{E}_{s\sim P_S}\left[\mathcal{D}\big(P_S(a \mid s), P_R(a \mid s)\big)\right] \tag{23}$$

$d_{\mathcal{X}}(P_S, P_R)$ measures the discrepancy between marginal state distributions:

$$d_{\mathcal{X}}(P_S, P_R) = \mathcal{D}_{\mathcal{X}}\big(P_S(s), P_R(s)\big) \tag{24}$$

4.3.1 Vector Distribution Alignment

Direct sample-wise alignment across domains is ill-posed due to lack of correspondence. Instead, we align conditional action distributions:

$$P_S(a \mid s) \quad \text{and} \quad P_R(a \mid s) \tag{25}$$

We define the vector alignment objective:

$$\mathcal{L}_{vector} = \mathbb{E}_{s\sim\mathcal{S}}\left[\mathcal{D}\big(P_S(a \mid s),\, P_R(a \mid s)\big)\right] \tag{26}$$

In practice, $P(a \mid s)$ is implicitly defined by the policy π_θ. We approximate the divergence via an adversarial discriminator:

$$D_m(\pi_\theta(f_m(x), s(x)); \delta) \tag{27}$$

where $D_m(\cdot;\delta)$ is a discriminator with parameters δ, it predicts whether a vector-conditioned action originates from D_S or D_R.

The adversarial loss is:

$$\mathcal{L}_{adv}^{vec} = \mathbb{E}_{x\sim D_S}\big[\log D_m(a(x);\delta)\big] + \mathbb{E}_{x\sim D_R}\big[\log(1 - D_m(a(x);\delta))\big] \tag{28}$$

where $a(x) = \pi_\theta(f_m(x), s(x))$.

Minimizing this loss with respect to θ and maximizing with respect to δ aligns $P_S(a|s)$ and $P_R(a|s)$.

4.3.2 Vector Attention Alignment

Modern motion planners rely on attention mechanisms to reason over vector tokens. Let:

$$\alpha(x) = \mathrm{Softmax}\left(\frac{Q(x)K(x)^\top}{\sqrt{d_k}}\right) \tag{29}$$

$Q(x), K(x) \in \mathbb{R}^{N\times d_k}$ are query and key matrices, d_k is the feature dimension, $\alpha(x) \in \mathbb{R}^{N\times N}$ is a row-stochastic attention matrix. We treat $\alpha(x)$ as a probability distribution over vectors and minimize divergence:

$$\mathcal{L}_{attn} = \mathbb{E}_{x\sim D_S\cup D_R}\left[\mathcal{D}\big(\alpha_S(x), \alpha_R(x)\big)\right] \tag{30}$$

The final motion-level alignment objective is:

$$\mathcal{L}_{motion} = \lambda_{adv}\mathcal{L}_{adv}^{vec} + \lambda_{vector}\mathcal{L}_{vector} + \lambda_{attn}\mathcal{L}_{attn} + D_m^\delta \tag{31}$$

λ_{adv}, λ_{vector}, and λ_{attn} are balancing coefficients. Similar D_p^δ, D_m^δ is the motion discriminator.

5 Experiments

5.1 Datasets

Real-world Datasets. We use the nuScenes [41] dataset, which contains 1000 multi-modal annotated scenes. Unlike the official split with significant scene overlap, we adopt a new split to better evaluate generalization. Specifically, simple scenarios (straight roads) are used for training, while challenging cases (intersections and unmarked roads) are reserved for validation, enabling assessment of robustness and transfer to hard real-world scenarios.

Simulation Datasets. This dataset is built on the CARLA Leaderboard v2 simulator [18], referred to as the CARLA dataset. We align sensor settings with nuScenes (GNSS, LiDAR, six cameras, and HD maps) and organize data into clips. Following [42], we simulate diverse hard cases (accidents, violations, irregular roads, pedestrian crossings) to facilitate transfer to real-world models.

5.2 Implementation Details

We use [11] as our baseline. The training process is complex and challenging in terms of convergence; therefore, we utilize eight A800 GPUs with a large total batch size to prevent the model from getting trapped in local optima or experiencing complete divergence. Additionally, we employ different optimizers for the baseline model and discriminators to enhance performance.

5.3 Comparisons with State-of-the-art Methods

Since our goal is to explore transfer learning from simulation to real-world scenarios and address the scarcity of rare and challenging scenarios in real-world datasets, we employ a new split strategy for the nuScenes dataset. This new split strategy assigns challenging scenarios (hard cases) to the

Table 1: **Open-loop planning performance on the new split nuScenes dataset.** Bridge-AD achieves a significant performance improvement over other methods across evaluation metrics. The best results achieved by our approach are highlighted in bold. *: Reproduced with simulation dataset pre-trained, real-world data finetuned. These experiments are reproduced on the new split nuScenes and simulation dataset, with all evaluations performed using NVIDIA Tesla A800 GPUs.

Methods	**L2 (m) ↓**				**Collision Rate (%) ↓**			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
UniAD [7]	0.57	0.86	1.40	0.94	0.67	0.75	1.03	0.82
VAD [8]	0.55	0.78	1.23	0.85	0.26	0.33	0.54	0.38
VADv2 [43]	0.55	0.75	1.21	0.84	0.24	0.34	0.50	0.36
SparseDrive [11]	0.56	0.75	1.19	0.83	0.18	0.31	0.47	0.32
SparseDrive-finetune*	0.59	0.71	1.17	0.82	0.18	0.30	0.50	0.33
Ours	**0.31**	**0.60**	**0.86**	**0.59**	**0.06**	**0.10**	**0.18**	**0.11**

Table 2: **Closed-loop Experiments on NAVSIM.** Bridge-AD achieves signicant improvement on the closed-loop experiments compare to similar methods.

Methods	**NC↑**	**DAC↑**	**TTC↑**	**Comfort↑**	**EP↑**	**PDMS↑**
UniAD[7]	97.8	91.9	92.9	100	78.8	83.4
Para-Drive[44]	97.9	92.4	93.0	99.8	79.3	84.0
Hydra-MDP-W [45]	98.1	96.1	93.9	100	77.8	85.7
AutoVLA (One-shot)[46]	96.89	94.43	88.06	99.94	75.82	80.54
Ours	**98.30**	**94.64**	**96.17**	**100**	**81.95**	**89.03**
AutoVLA (Post-RFT)[46]	98.41	95.40	98.04	99.94	81.87	89.11
AutoVLA (Best-of-N)[46]	99.14	95.51	97.12	99.98	87.55	92.12

Table 3: **Open-loop planning performance on the original split nuScenes dataset.** Bridge-AD achieves a slight improvement over other methods across evaluation metrics. These experiments are reproduced on the original split nuScenes and simulation dataset. *: Official checkpoint re-validated with corrected metrics.

Methods	**L2 (m) ↓**				**Collision Rate (%) ↓**			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
UniAD[7]*	0.45	0.70	1.04	0.73	0.62	0.58	0.63	0.61
VAD[8]*	0.41	0.70	1.05	0.72	0.07	0.17	0.41	0.22
PPAD[47]	0.30	0.69	1.26	0.75	0.03	0.22	0.73	0.33
ParaDrive [44]	0.26	0.59	1.12	0.66	0.00	0.12	0.65	0.26
SparseDrive[11]	0.29	0.63	0.97	0.63	0.03	0.09	0.19	0.10
AlignDrive[48]	0.38	0.73	1.23	0.78	0.01	0.04	0.14	0.06
HiP-AD[49]	0.28	0.53	0.87	0.56	0.01	0.05	0.15	0.07
Ours	**0.23**	**0.50**	**0.82**	**0.52**	0.02	0.05	**0.12**	**0.06**

Table 4: Ablation studies were conducted on the key design elements of Bridge-AD using the new split real-world dataset and simulation dataset. The results prove the effectiveness of our design.

Index	**Module Components**	**L2(m) ↓**	**Collision Rate(%) ↓**
1	Simulation baseline: Simulation dataset trained, real-world dataset tested	14.45	28.73
2	Real-world baseline: Real-world dataset trained, real-world dataset tested	0.83	0.32
3	Simulation dataset pre-trained real-world dataset finetuned	0.82	0.33
4	Based on two baselines + LVDA	0.69	0.21
5	Based on two baselines + GAPA	0.63	0.18
6	Ours (Based on two baseline + LVDA +GAPA)	0.59	0.11

evaluation dataset to assess the ability of our method to transfer hard cases effectively. To ensure a fair comparison with other SOTA methods, we retrained these models using our new split strategy on the nuScenes dataset. As shown in Table 1, we compare the performance of our approach with existing methods. All methods are evaluated on the newly split nuScenes dataset. Our approach demonstrates superior performance on the newly split dataset. Specifically, Bridge-AD surpasses competing methods by more than 28.0% and 66.6% in terms of average L2 distance error and average collision rate, respectively. These results strongly validate the effectiveness of our novel model design.

5.4 Closed-loop Evaluation

Since our goal is to transfer the model's ability to handle challenging scenarios into real-world settings, it is essential to perform closed-loop evaluations in real-world environments. To this end, we adopted the open-source NAVSIM [50]. For a fair comparison with existing methods on the NAVSIM dataset, we used the standard navtrain and navtest splits for training and validation, respectively, as these splits were also used in prior work. We followed the community's recommended PDMS metric for final evaluation. As shown in Tab. 2, our method achieves notable improvements over the existing Hydra-MDP-W over 3.9% in terms of PDMS. Our method, despite its small parameter size, achieves performance comparable to AutoVLA on most metrics and outperforms AutoVLA (One-shot), demonstrating the effectiveness of sim-to-real adversarial transfer and multi-task end-to-end learning. While AutoVLA (Post-RFT) attains slightly higher PDMS scores due to its 7B backbone and extensive tuning, our approach maintains competitive EP and comfort without large-scale language supervision. Compared to AutoVLA (Best-of-N), the performance gap is mainly due to the significant difference in model size (7B vs. 0.105B).

5.5 Evaluation on Original Split NuScenes Dataset

We provide a comparison of Bridge-AD's performance against existing methods on the original split nuScenes dataset. As shown in Table 3, our method achieves an average L2 distance of 0.52 and an average collision rate of 0.06, which are slightly better than the competing methods by 0.04 and 0.01, respectively. When combined with the main experiments conducted above, the results indicate that while our method excels in identifying and handling hard cases, it shows limited improvement in normal scenarios, which have already been extensively studied during the training process [51, 52].

5.6 Ablation Study

5.6.1 Key Components Design

We validate our components by performing an ablation study to prove the effectiveness of our model design. As shown in Table 4, we start by validating our baseline models: the simulation baseline (only the simulation model) and the real-world baseline (only the real-world model). Specifically, in Index-1, we train our simulation baseline model on the simulation dataset and evaluate it on the real-world validation dataset. Notably, the baseline (Index-1) demonstrates extremely poor performance. In the second row (Index-2), we train our real-world model on the real-world dataset. However, since hard cases were assigned to the validation dataset and easy cases to the training dataset, the model's performance is suboptimal, achieving only 0.83 in L2 distance and 0.32 in collision rate. We also pre-trained our baseline with a simulation dataset, and finetuned with the real-world dataset, achieving 0.82 in L2 distance and 0.33 in collision rate. As shown in Index-4, using only the motion discriminator yields an L2 distance of 0.69 and a collision rate of 0.21. Using the perception discriminator further enhances feature representation, reducing the L2 distance by 0.06 and the collision rate by 0.03.

6 Conclusion

In this study, we proposed an end-to-end autonomous driving method that transfers simulated imitation driving capabilities to real-world deployment. The approach comprises two base models and two discriminators. The simulation model learns uncommon scenarios from synthetic data and transfers this knowledge to the real-world model through adversarial learning with discriminators. Importantly, we designed an efficient adversarial training process, which is critical for the model's successful optimization. Our method effectively addresses hard cases in real-world scenarios. We evaluated our method on the newly split nuScenes dataset, and the experimental results demonstrate that Bridge-AD significantly outperforms existing methods in complex driving scenarios.

References

[1] D. A. Pomerleau, "Alvinn: an autonomous land vehicle in a neural network," in *Proceedings of the 1st International Conference on Neural Information Processing Systems*, ser. NIPS'88. Cambridge, MA, USA: MIT Press, 1988, p. 305–313.

[2] A. Prakash, K. Chitta, and A. Geiger, "Multi-modal fusion transformer for end-to-end autonomous driving," 2021. [Online]. Available: https://arxiv.org/abs/2104.09224

[3] S. Casas, A. Sadat, and R. Urtasun, "Mp3: A unified model to map, perceive, predict and plan," 2021. [Online]. Available: https://arxiv.org/abs/2101.06806

[4] A. Cui, S. Casas, A. Sadat, R. Liao, and R. Urtasun, "Lookout: Diverse multi-future prediction and planning for self-driving," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, Oct. 2021. [Online]. Available: http://dx.doi.org/10.1109/ICCV48922.2021.01580

[5] A. Sadat, S. Casas, M. Ren, X. Wu, P. Dhawan, and R. Urtasun, "Perceive, predict, and plan: Safe motion planning through interpretable semantic representations," 2020. [Online]. Available: https://arxiv.org/abs/2008.05930

[6] S. Hu, L. Chen, P. Wu, H. Li, J. Yan, and D. Tao, "St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning," 2022. [Online]. Available: https://arxiv.org/abs/2207.07601

[7] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li, "Planning-oriented autonomous driving," 2023. [Online]. Available: https://arxiv.org/abs/2212.10156

[8] B. Jiang, S. Chen, Q. Xu, B. Liao, J. Chen, H. Zhou, Q. Zhang, W. Liu, C. Huang, and X. Wang, "Vad: Vectorized scene representation for efficient autonomous driving," 2023. [Online]. Available: https://arxiv.org/abs/2303.12077

[9] T. Ye, W. Jing, C. Hu, S. Huang, L. Gao, F. Li, J. Wang, K. Guo, W. Xiao, W. Mao, H. Zheng, K. Li, J. Chen, and K. Yu, "Fusionad: Multi-modality fusion for prediction and planning tasks of autonomous driving," 2023. [Online]. Available: https://arxiv.org/abs/2308.01006

[10] S. Chen, B. Jiang, H. Gao, B. Liao, Q. Xu, Q. Zhang, C. Huang, W. Liu, and X. Wang, "Vadv2: End-to-end vectorized autonomous driving via probabilistic planning," 2024. [Online]. Available: https://arxiv.org/abs/2402.13243

[11] W. Sun, X. Lin, Y. Shi, C. Zhang, H. Wu, and S. Zheng, "Sparsedrive: End-to-end autonomous driving via sparse scene representation," 2024. [Online]. Available: https://arxiv.org/abs/2405.19620

[12] D. Zhang, D. Chen, P. Zhi, Y. Chen, Z. Yuan, C. Li, Sunjing, R. Zhou, and Q. Zhou, "Mapexpert: Online hd map construction with simple and efficient sparse map element expert," 2024. [Online]. Available: https://arxiv.org/abs/2412.12704

[13] H. Shao, Y. Hu, L. Wang, S. L. Waslander, Y. Liu, and H. Li, "Lmdrive: Closed-loop end-to-end driving with large language models," 2023. [Online]. Available: https://arxiv.org/abs/2312.07488

[14] J. Wang, X. Zhang, Z. Xing, S. Gu, X. Guo, Y. Hu, Z. Song, Q. Zhang, X. Long, and W. Yin, "He-drive: Human-like end-to-end driving with vision language models," 2024. [Online]. Available: https://arxiv.org/abs/2410.05051

[15] Y. Xu, Y. Hu, Z. Zhang, G. P. Meyer, S. K. Mustikovela, S. Srinivasa, E. M. Wolff, and X. Huang, "Vlm-ad: End-to-end autonomous driving through vision-language model supervision," 2024. [Online]. Available: https://arxiv.org/abs/2412.14446

[16] Z. Yuan, C. Qian, J. Tang, R. Chen, Z. Song, L. Sun, X. Chu, Y. Cai, D. Zhang, and S. Li, "Autodrive-r^2 : $Incentivizingreasoningandself-reflectioncapacityforvlamodelinautonomousdriving,''$ *inInternational Conference on Learning Representations*, $C.Vondrick, B.$ $-97\,955.[Online].Available$:

D. Zhang, Z. Yuan, Z. Chen, C.-T. Liao, Y. Chen, F. Shen, Q. Zhou, and T.-S. Chua, "Reasoning-vla: An efficient and spatial-guided general vision-language-action reasoning model for autonomous driving," in *Forty-third International Conference on Machine Learning*.

A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," 2017. [Online]. Available: `https://arxiv.org/abs/1711.03938`

G. Rong, B. H. Shin, H. Tabatabaee, Q. Lu, S. Lemke, M. Možeiko, E. Boise, G. Uhm, M. Gerow, S. Mehta, *et al.*, "Lgsvl simulator: A high fidelity simulator for autonomous driving," *arXiv preprint arXiv:2005.03778*, 2020.

F. Codevilla, E. Santana, A. M. López, and A. Gaidon, "Exploring the limitations of behavior cloning for autonomous driving," 2019. [Online]. Available: https://arxiv.org/abs/1904.08980

D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, "Learning by cheating," 2019. [Online]. Available: https://arxiv.org/abs/1912.12294

K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger, "Transfuser: Imitation with transformer-based sensor fusion for autonomous driving," 2022. [Online]. Available: https://arxiv.org/abs/2205.15997

Z. Liu, H. Tang, A. Amini, X. Yang, H. Mao, D. Rus, and S. Han, "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," 2024. [Online]. Available: https://arxiv.org/abs/2205.13542

D. Zhang, P. Zhi, B. Yong, J.-Q. Wang, Y. Hou, L. Guo, Q. Zhou, and R. Zhou, "Ehss: An efficient hybrid-supervised symmetric stereo matching network," *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1044–1051, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:267661311

K. Renz, K. Chitta, O.-B. Mercea, A. S. Koepke, Z. Akata, and A. Geiger, "Plant: Explainable planning transformers via object-level representations," 2022. [Online]. Available: https://arxiv.org/abs/2210.14222

K. Chitta, A. Prakash, and A. Geiger, "Neat: Neural attention fields for end-to-end autonomous driving," 2021. [Online]. Available: https://arxiv.org/abs/2109.04456

X. Jia, P. Wu, L. Chen, J. Xie, C. He, J. Yan, and H. Li, "Think twice before driving: Towards scalable decoders for end-to-end autonomous driving," 2023. [Online]. Available: https://arxiv.org/abs/2305.06242

Z. Zhang, A. Liniger, D. Dai, F. Yu, and L. V. Gool, "End-to-end urban driving by imitating a reinforcement learning coach," 2021. [Online]. Available: https://arxiv.org/abs/2108.08265

P. Wu, X. Jia, L. Chen, J. Yan, H. Li, and Y. Qiao, "Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline," 2022. [Online]. Available: https://arxiv.org/abs/2206.08129

X. Jia, Y. Gao, L. Chen, J. Yan, P. L. Liu, and H. Li, "Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving," 2023. [Online]. Available: https://arxiv.org/abs/2308.00398

Z. Xu, Y. Zhang, E. Xie, Z. Zhao, Y. Guo, K.-Y. K. Wong, Z. Li, and H. Zhao, "Drivegpt4: Interpretable end-to-end autonomous driving via large language model," 2024. [Online]. Available: https://arxiv.org/abs/2310.01412

W. Wang, J. Xie, C. Hu, H. Zou, J. Fan, W. Tong, Y. Wen, S. Wu, H. Deng, Z. Li, H. Tian, L. Lu, X. Zhu, X. Wang, Y. Qiao, and J. Dai, "Drivemlm: Aligning multi-modal large language models with behavioral planning states for autonomous driving," 2023. [Online]. Available: https://arxiv.org/abs/2312.09245

C. Sima, K. Renz, K. Chitta, L. Chen, H. Zhang, C. Xie, J. Beißwenger, P. Luo, A. Geiger, and H. Li, "Drivelm: Driving with graph visual question answering," 2025. [Online]. Available: https://arxiv.org/abs/2312.14150

D. Zhang, F. Shen, R. Zhao, Y. Chen, P. Zhi, C. Li, R. Zhou, and Q. Zhou, "Coc-vla: Delving into adversarial domain transfer for explainable autonomous driving via chain-of-causality visual-language-action model," in *Advances in Neural Information Processing Systems*, D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen, Eds., vol. 38, Main Conference. Curran Associates, Inc., 2025, pp. 70 912–70 939.

[Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2025/file/66d09284cfb6f125fe888f71dc14f35e-Paper-Conference.pdf

C. Cui, Y. Ma, Z. Yang, Y. Zhou, P. Liu, J. Lu, L. Li, Y. Chen, J. H. Panchal, A. Abdelraouf, R. Gupta, K. Han, and Z. Wang, "Large language models for autonomous driving (llm4ad): Concept, benchmark, simulation, and real-vehicle experiment," 2024. [Online]. Available: https://arxiv.org/abs/2410.15281

S. Kornblith, J. Shlens, and Q. V. Le, "Do better imagenet models transfer better?" 2019. [Online]. Available: https://arxiv.org/abs/1805.08974

D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," 2019. [Online]. Available: https://arxiv.org/abs/1901.09960

M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 2200–2207.

G. Wei, C. Lan, W. Zeng, and Z. Chen, "Metaalign: Coordinating domain alignment and classification for unsupervised domain adaptation," 2021. [Online]. Available: https://arxiv.org/abs/2103.13575

Y. Yang, H. Yu, X. Lou, Y. Liu, and C. Choi, "Attribute-based robotic grasping with data-efficient adaptation," *IEEE Transactions on Robotics*, vol. 40, pp. 1566–1579, 2024.

H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," 2020. [Online]. Available: https://arxiv.org/abs/1903.11027

X. Jia, Z. Yang, Q. Li, Z. Zhang, and J. Yan, "Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving," 2024. [Online]. Available: https://arxiv.org/abs/2406.03877

S. Chen, B. Jiang, H. Gao, B. Liao, Q. Xu, Q. Zhang, C. Huang, W. Liu, and X. Wang, "Vadv2: End-to-end vectorized autonomous driving via probabilistic planning," 2024. [Online]. Available: https://arxiv.org/abs/2402.13243

X. Weng, B. Ivanovic, Y. Wang, Y. Wang, and M. Pavone, "Para-drive: Parallelized architecture for real-time autonomous driving," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 15 449–15 458.

Z. Li, K. Li, S. Wang, S. Lan, Z. Yu, Y. Ji, Z. Li, Z. Zhu, J. Kautz, Z. Wu, Y.-G. Jiang, and J. M. Alvarez, "Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation," 2024. [Online]. Available: https://arxiv.org/abs/2406.06978

Z. Zhou, T. Cai, S. Z. Zhao, Y. Zhang, Z. Huang, B. Zhou, and J. Ma, "Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning," 2025. [Online]. Available: https://arxiv.org/abs/2506.13757

Z. Chen, M. Ye, S. Xu, T. Cao, and Q. Chen, "Ppad: Iterative interactions of prediction and planning for end-to-end autonomous driving," 2024. [Online]. Available: https://arxiv.org/abs/2311.08100

Y. Wu, H. Zhang, F. He, R. Wu, C. Qiu, L. Gao, W. Ke, and T. Zhang, "Alignndrive: Aligned lateral-longitudinal planning for end-to-end autonomous driving," 2026. [Online]. Available: https://arxiv.org/abs/2601.01762

Y. Tang, Z. Xu, Z. Meng, and E. Cheng, "Hip-ad: Hierarchical and multi-granularity planning with deformable attention for autonomous driving in a single decoder," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 25 605–25 615.

D. Dauner, M. Hallgarten, T. Li, X. Weng, Z. Huang, Z. Yang, H. Li, I. Gilitschenski, B. Ivanovic, M. Pavone, A. Geiger, and K. Chitta, "Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking," 2024. [Online]. Available: https://arxiv.org/abs/2406.15349

A. Swerdlow, R. Xu, and B. Zhou, "Street-view image generation from a bird's-eye view layout," 2024. [Online]. Available: https://arxiv.org/abs/2301.04634

Y. Wen, Y. Zhao, Y. Liu, F. Jia, Y. Wang, C. Luo, C. Zhang, T. Wang, X. Sun, and X. Zhang, "Panacea: Panoramic and controllable video generation for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 6902–6912.

I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014. [Online]. Available: https://arxiv.org/abs/1406.2661

Z. Yuan, Y. Wang, D. Zhang, K. Cui, R. Chen, J. Tang, L. Sun, H. Yu, C. Qian, X. Chu, *et al.*, "What if agents could imagine? reinforcing open-vocabulary hoi comprehension through generation," *(NeurIPS'26) Advances in Neural Information Processing Systems*, 2026.

Z. Yuan, X. Qu, C. Qian, R. Chen, J. Tang, L. Sun, X. Chu, D. Zhang, Y. Wang, Y. Cai, *et al.*, "Video-star: Reinforcing open-vocabulary action recognition with tools," in *(ICLR'26) The Fourteenth International Conference on Learning Representations*, 2025.

D. Zhang, J. Sun, C. Hu, X. Wu, Z. Yuan, R. Zhou, F. Shen, and Q. Zhou, "Pure vision language action (vla) models: A comprehensive survey," *arXiv preprint arXiv:2509.19012*, 2025.

M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014. [Online]. Available: https://arxiv.org/abs/1411.1784

Z. Yuan, D. Zhang, Z. Li, C. Qian, J. Chen, Y. Chen, K. Chen, T. Mao, Z. Li, H. Jiang, *et al.*, "Dvp-mvs++: Synergize depth-normal-edge and harmonized visibility prior for multi-view stereo," *(TCSVT'25) IEEE Transactions on Circuits and Systems for Video Technology*, 2025.

P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," 2018. [Online]. Available: https://arxiv.org/abs/1611.07004

C. Li, K. Xu, J. Zhu, and B. Zhang, "Triple generative adversarial nets," 2017. [Online]. Available: https://arxiv.org/abs/1703.02291

T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to discover cross-domain relations with generative adversarial networks," 2017. [Online]. Available: https://arxiv.org/abs/1703.05192

F. Luan, S. Paris, E. Shechtman, and K. Bala, "Deep photo style transfer," 2017. [Online]. Available: https://arxiv.org/abs/1703.07511

S. Kim, Q. Song, and F. Liang, "A new paradigm for generative adversarial networks based on randomized decision rules," 2023. [Online]. Available: https://arxiv.org/abs/2306.13641

X. Li, X. Wang, Z. Qi, H. Cao, Z. Zhang, and A. Xiang, "Dtsgan: Learning dynamic textures via spatiotemporal generative adversarial network," 2024. [Online]. Available: https://arxiv.org/abs/2412.16948

W. Ma, O. Karakus, and P. L. Rosin, "Patch-gan transfer learning with reconstructive models for cloud removal," 2025. [Online]. Available: https://arxiv.org/abs/2501.05265

Z. Xie, J. Cao, Y. Zhang, Q. Zhang, and R. Xu, "A dual-agent adversarial framework for robust generalization in deep reinforcement learning," 2025. [Online]. Available: https://arxiv.org/abs/2501.17384

A Appendix: Extension Experiments

A.1 Perception and Prediction Performance

To further evaluate our method, we conducted several sub-task experiments on the newly split nuScenes validation set. Below, we compare our results with state-of-the-art methods.

Detection. We conduct a 3D detection evaluation experiment. As shown in Table 5, Bridge-AD achieves 0.437 mAP, 0.572 mATE, 0.258 mASE, 0.402 mAOE, 0.255 mAVE, 0.170 mAAE and

0.553 NDS, surpassing SparseDrive [11] and UniAD [7] over all metrics. These results demonstrate the effectiveness of our model design.

Multi-object Tracking. Since our method is an end-to-end autonomous driving approach, we generate tracking results to enhance the continuity of temporal and spatial features. The evaluation metrics for tracking are detailed in Table 6. Our method achieves 0.467 AMOTA, 1.143 AMOTP, 0.579 Recall, and 656 IDS, outperforming existing methods by 29.0%, 14.1%, 25.1%, and 8.8%, respectively.

Online mapping. The results of the online mapping are illustrated in Table 7, Bridge-AD delivers the expected performance, achieving a 21.2% improvement in Mean Average Precision (mAP) compared to SparseDrive [11]. Specifically, Bridge-AD significant increase in all map elements AP, yielding 40.1, 43.3 and 47.0 in terms of pedestrian crossings, lane dividers, and road boundaries.

Prediction. Agent trajectory prediction significantly influences the final ego motion planning. Therefore, we evaluate the prediction results to assess our method. Table 8 illustrates that our model outperforms the existing method by 0.09, 0.28, 0.008 and 0.066 in minADE, minFDE, MR and EPA respectively.

Table 5: **Object detection on the new split nuScenes dataset.** Bridge-AD achieves the best performance on detection tasks. All other methods are re-implemented on the new split nuScenes dataset for a fair comparison.

Methods	mAP ↑	mATE ↓	mASE ↓	mAOE ↓	mAVE ↓	mAAE ↓	NDS ↑
UniAD [7]	0.311	0.606	0.277	0.638	0.306	0.184	0.455
SparseDrive [11]	0.396	0.581	0.282	0.505	0.284	0.175	0.515
Ours	**0.437**	**0.572**	**0.258**	**0.402**	**0.255**	**0.170**	**0.553**

Table 6: **Multi-object tracking.** Bridge-AD achieves competitive performance against state-of-the-art methods, Experiments are reproduced on the new split nuScenes.

Methods	AMOTA ↑	AMOTP ↓	Recall ↑	IDS ↓
UniAD[7]	0.300	1.454	0.387	939
SparseDrive[11]	0.362	1.331	0.463	719
Ours	**0.467**	**1.143**	**0.579**	**656**

Table 7: **Online mapping on the new split nuScenes dataset.** Comparison with state-of-the-art method, Bridge-AD outperforms other methods over all metrics.

Methods	AP_{ped} ↑	AP_{div} ↑	AP_{bound} ↑	mAP ↑
VAD[8]	27.5	24.6	40.8	31.0
SparseDrive[11]	32.7	35.2	39.8	35.9
Ours	**40.1**	**43.3**	**47.0**	**43.5**

A.2 Evaluation on Straight Regressor Baseline

We design a new dummy regressor baseline based on our method, termed Straight Regressor Baseline. To produce a straight trajectory, we ignore the y-axis, only predict the x-axis trajectory values, whereas existing methods predict both (x, y) coordinates. We trained and evaluated this Straight Regressor Baseline on new nuScenes split, the results are illustrated in Tab. 9.

This result demonstrates that the new split of the nuScenes dataset, which includes challenging scenarios, effectively shows that our strategy successfully transfers challenge handling capabilities to the real-world model.

We also evaluated the Straight Regressor Baseline on the original nuScenes split. The results in Tab. 10, indicate that the new split strategy is significantly more challenging than the original nuScenes split.

Table 8: **Prediction.** Bridge-AD achieves competitive performance against state-of-the-art perception-oriented methods on the new split nuScenes dataset.

Methods	minADE(m)↓	minFDE(m)↓	MR↓	EPA↑
UniAD[7]	0.78	1.17	0.142	0.462
SparseDrive[11]	0.76	1.35	0.145	0.445
Ours	**0.67**	**1.07**	**0.137**	**0.511**

Table 9: **Performance of Straight Regressor Baseline on new nuScenes split.**

Methods	L2 (m) ↓				Collision Rate (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Straight Regressor Baseline	0.82	1.10	2.07	1.33	0.74	1.18	1.65	1.19
Ours	**0.31**	**0.60**	**0.86**	**0.59**	**0.06**	**0.10**	**0.18**	**0.11**

A.3 Granular Studies

In our main method, the ratio of simulation data vs real-world data is 0.6 (600:1000), we added an experiment of 0.3, we have achieved the results in Tab. 11. As seen, the performance drop with low simulation dataset ratio.

A.4 Qualitative Analysis

We conducted additional qualitative comparisons to systematically evaluate the performance of our Bridge-AD method. Figure 2 and 3 illustrate a continuous bypass scenario. In Figure 2, our method bypasses the vehicles on the left and prepares to return to its lane. However, when another vehicle appears ahead in the path, as shown in Figure 3, the ego vehicle aborts the lane return and executes a new bypass maneuver. Figure 4 depicts a complex intersection scenario, where crosswalk markings could mislead trajectory planning. However, our method successfully generates an optimal path, as observed in Figure 4. Figure 5 illustrates a turning scenario surrounded by vehicles. The qualitative results indicate that our method generates an effective trajectory plan.

Bridge-AD's ability to transfer rare and challenging scenarios knowledge enables it to handle complex trajectory predictions. As shown in Figure 6, at a road junction, the vehicles behind (captured in the rear and rear-right camera views) are positioned at a critical point for lane switching. Our method generates complex predicted trajectories, which is one of the major limitations.

B Appendix: Supplements

B.1 Limitations of Real-World Data

In real-world autonomous driving application scenarios, researchers collect data from vehicles, these vehicles are driven by skilled people. This leads to three key challenges: (1) Autonomous driving is hindered by data collection and data labeling in the physical world, both of which require substantial human and financial resources; (2) Rare scenarios, such as traffic accidents, driving violations, non-standard roads, pedestrian trespassing, and extreme weather conditions, are challenging to capture. (3) There is a lack of high-order interactions and efficient decision-making between vehicles in real-world environments. As imitation learning-based approaches, current autonomous driving methods cannot guarantee vehicle safety and reliability across all driving scenarios unless these issues are addressed.

Table 10: **Performance of Straight Regressor Baseline on origin nuScenes split.**

Methods	L2 (m) ↓				Collision Rate (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Straight Regressor Baseline	0.58	0.77	1.31	0.89	0.66	0.69	0.75	0.70
Ours	**0.23**	**0.50**	**0.82**	**0.52**	0.02	0.05	**0.12**	**0.06**

Table 11: **Granular Studies over simulation data vs real-world.**

Methods	L2	Collision Rate
Ours (with dataset ratio = 0.3)	0.63	0.13

B.2 Limitations of Simulation Data

Recently, simulations have become widely used in autonomous driving research. Autonomous driving simulators support flexible configurations of sensor suites, kinetic models, interactive static and dynamic scenes, environmental conditions (such as weather and lighting), and even detailed collision data. These simulations effectively address some of the real-world challenges mentioned above. Simulations facilitate the collection of large-scale labeled data, which can be utilized to validate the effectiveness of data-driven models. Furthermore, since certain rare and challenging scenarios are too dangerous to stage in real-world settings, simulators can generate them freely. Additionally, autonomous driving models can undergo closed-loop evaluations in simulators, making them more representative of real-world scenarios. Researchers are increasingly focused on developing more realistic simulation environments. However, a significant gap remains between simulations and real-world driving. In summary, simulations introduce a new challenge: how to transfer models trained on virtual datasets to real-world applications.

B.3 Methods Shift

Autonomous driving methods face the domain shift problem, where models perform well in simulated environments during training but struggle in real-world driving scenarios. To address these challenges, we propose a novel method that simulates non-standard driving scenarios and integrates them into our model to enhance reliability. We train our model using a novel adversarial strategy to transfer simulated scenarios into real-world applications.

B.4 Loss

Different from existing methods, our approach employs distinct loss functions at different training stages.

Pre-training Loss. Firstly, during the pre-training step, we adopt a multi-task loss similar to [11] for both the simulation model and real-world model, which serves to pre-train our base model. The multi-task loss is defined as follows:

$$l_{pre} = l_{det} + l_{map} + l_{mot} + l_{pred} + l_{plan} \tag{32}$$

where l_{det} and l_{map} denotes the detection loss and online map construction loss, l_{mot} is the tracking loss, l_{pred} and l_{plan} correspond to the prediction loss and planning loss, respectively.

Adversarial Training Loss. Secondly, in step 1 of adversarial training, we incorporate two discriminator losses: the perception discriminator loss and the motion discriminator loss. Both are inverted-label cross-entropy loss. These losses enable the discriminators to learn the differences between the two data distributions, and they are defined as follows:

$$l_{step1} = l_{pd} + l_{md} \tag{33}$$

Here, l_{pd} and l_{md} are perception discriminator loss and motion discriminator loss, respectively. In step 2 of adversarial training, we use both the discriminator loss and the real-world multi-task loss, as shown below:

$$l_{step2} = l_{det} + l_{map} + l_{mot} + l_{pred} + l_{plan} + l_{pd} + l_{md} \tag{34}$$

However, as described in the Model Training section, two discriminators only pass the gradient parameters, and do not update their model parameters.

B.5 Model Training

Our training process consists of two stages: pre-training and adversarial training.

B.5.1 Pre-training

In this training process, we train our simulation model with simulation datasets. Since the simulation datasets contain a large number of hard cases, such as accidents ahead, vehicle driving violations, non-standard roads, and pedestrian trespassing, it is designed to handle complex scenarios effectively. Next, we train our real-world model using both simulation and real-world datasets, to establish a foundation model for rapid convergence during adversarial training.

B.5.2 Adversarial Training with Multi-Step Back-Propagation

The adversarial training process is a crucial component of our model, as detailed in fig. 1. We load the pre-trained model for our simulation and real-world model firstly. The adversarial training process involves multiple steps with our unique back-propagation strategies.

Step 1:

Forward Propagation: In this forward propagation, the simulation and real-world models are frozen, and only the discriminators are trained. We feed simulation data and real-world data into simulation model and real-world model separately, then the features extracted from both models are subsequently fed into the discriminators.

Back Propagation: When our discriminators outputs the final logits, we used discriminator losses to perform the backward propagation. Since only the discriminators are trainable at this stage, we back-propagate and update exclusively the discriminator parameters, while the simulation and real-world models remain unchanged.

Step 2:

Forward Propagation: In this process, only the real-world model and discriminators are involved. Real-world data are fed into the real-world model, and the resulting features are passed through the discriminators to obtain logits.

Back Propagation: The backward propagation in this step is our unique contribution. The backward propagation process is different from others. Firstly, we only back-propagate the discriminator losses pass through the two discriminators separately without updating parameters of two discriminators. Secondly, the back-propagated gradients from discriminators and the multi-task supervised losses are used to back-propagate and update the real-world model. This strategy mitigates convergence instability and accelerates adversarial training.

B.6 Dataset Analysis

Simulation Dataset

We have included a statistical distribution of the challenging simulation scenarios, as shown in Tab. 12. A new supplementary table below provides detailed counts for each scenario category. We hope these additions enhance the transparency of our dataset and support a more rigorous interpretation of the model's performance in handling rare but critical driving events.

Real-World Dataset

To demonstrate our model's ability to handle uncommon cases, we used the challenging subset of the nuScenes dataset as the val set (a total of 122 scenes), while the remaining easier cases were used for training. The test set includes scenarios such as: Scene-0026_379: The ego vehicle is intercepted by a construction worker yielding to a construction truck approaching from the left. Scenes 0046_568, 0094_948, 0131_1153, and 0162_1556: Instances of pedestrian intrusion. Scene-0150_1358: A construction worker blocks the road using traffic cones or water barriers in front of the ego vehicle. Scene-0201_1978: The ego vehicle is blocked by a car attempting to park in a designated area. As shown in the last two rows of Table 1, our proposed method outperforms existing standard methods on these challenging scenarios.

B.7 Dataset Splitting

Most end-to-end autonomous driving methods evaluate their performance using the nuScenes dataset, which follows the original 700/150/150 split for training, validation, and testing scenes. However,

Table 12: CARLA Scenarios Statistical Distribution.

Scenarios Category	**Proportion(%)**	**Clips Count(total 600)**
Normal (Straight, Left, Right Turn)	61.6	370
Pedestrian Intrusion	4.7	28
Fog	6.7	40
Rain	7.8	47
NearMiss Vehicle Interactions	2.7	16
Traffic Jam	4.5	27
Traffic Accident	1.8	11
Vehicle Cut In	2.2	13
Opposite Vehicle Intrusion	0.8	5
Vehicle U-Turning	0.5	3
Construction Obstacle	4.0	24
Bicycle Intrusion	0.5	3
Lane Merge	1.2	7
No Traffic Light Intersection	0.7	4
Turn Left and Merge In	0.3	2

this original split was designed for normal case scenarios, not hard cases. In the original split, the training and validation sets overlap by over 84% in terms of static environments, with both easy and hard scenarios proportionally distributed across both sets. This overlap means that a model could easily memorize the static environments from the training set and perform exceptionally well on the validation set, which can lead to complete failure when generalizing to new, unseen scenes. To better assess the model's ability to generalize to unfamiliar environments and adapt to challenging cases, we propose a new dataset split, where hard cases are used for validation and easy cases for training. This approach focuses on testing the model's ability to transfer simulated hard cases to unseen real-world scenarios.

To provide a clearer understanding, we visualize both the original and the new dataset splits. As shown in Figure 7 and Figure 8, the green line represents the training set, the blue line indicates the validation set, and the red line denotes the test set. We ensure that there is no overlap between the training and validation datasets across all two cities, maintaining a balanced distribution. As depicted in these figures, our new split strategy assigns the hard cases to the validation set, thereby preventing the model from memorizing scenarios during the training process.

B.8 Metric Acronyms

We have add a subsection below to explain the metric acronyms. The short summary is below:

L2(m): L2 distance, unit is meter.

Avg.: Average of 1 second, 2 second and 3 second.

mAP: Mean Average Precision.

mATE: Mean Average Translation Error.

mASE: Mean Average Scale Error.

mAOE: Mean Average Orientation Error.

mAVE: Mean Average Velocity Error.

mAAE: Mean Average Attribute Error.

NDS: nuScenes Detection Score.

AMOTA: Average Multi Object Tracking Accuracy.

AMOTP: Average Multi Object Tracking Precision.

APped,APdiv,APbound: Average Precision of pedestrian crossings, lane dividers, and road boundaries.

minADE: Minimum Average Displacement Error.

minFDE: Minimum Final Displacement Error.

MR: Miss Rate.

B.9 Domain Adversarial Transfer

Furthermore, a new strategy for style transfer has emerged. GAN [53] pioneers the adversarial generative model, which simultaneously trains a generative model to capture the data distribution and a discriminative model to estimate the probability of samples. This method learns deep representations without requiring extensively annotated training data [54–56]. Another approach directly controls the data generation process by incorporating additional information, such as class labels or multimodal data [57, 58]. Instead of learning a generic generative model, Pix2Pix [59] learns a conditional generative model, which is widely used for image-to-image translation tasks, this simple framework is sufficient to fit the expected distribution using a simple loss function. Subsequent works successfully transfer style between domains using various techniques, such as stacking and additional branches [60–62]. Recently, Kim et al. [63] investigates randomized decision rules, and introduces an empirical Bayes-like method in the training process to improve the diversity of outputs. Furthermore, DTSGAN [64] introduces a spatio-temporal generative adversarial network that captures motion and content distribution in video sequences. Patch-GAN [65] proposes a reconstructive transfer learning approach built on a generative adversarial network (GAN) to explore the potential of the novel masked auto-encoder (MAE) image reconstruction model. This approach achieves a significant improvement compared to other GAN-based methods. Additionally, some researchers incorporate GANs into reinforcement learning tasks to enhance robustness. Among them, Xie et al. [66] proposes a dual-agent adversarial policy learning model, enabling agents to spontaneously learn semantics without additional human prior knowledge. Experiments demonstrate that this adversarial framework significantly improves performance.

B.10 Limitations

While our model significantly improves upon existing methods, it still has certain limitations. The complexity of our training process makes optimization challenging, particularly during the initial epochs. If minor deviations occur in the initial epochs, the entire training process may diverge. Moreover, maintaining two baseline models results in high training costs.



Figure 2: Qualitative results on continuity bypass case. Part 1.



Figure 3: Qualitative results on continuity bypass case. Part 2.



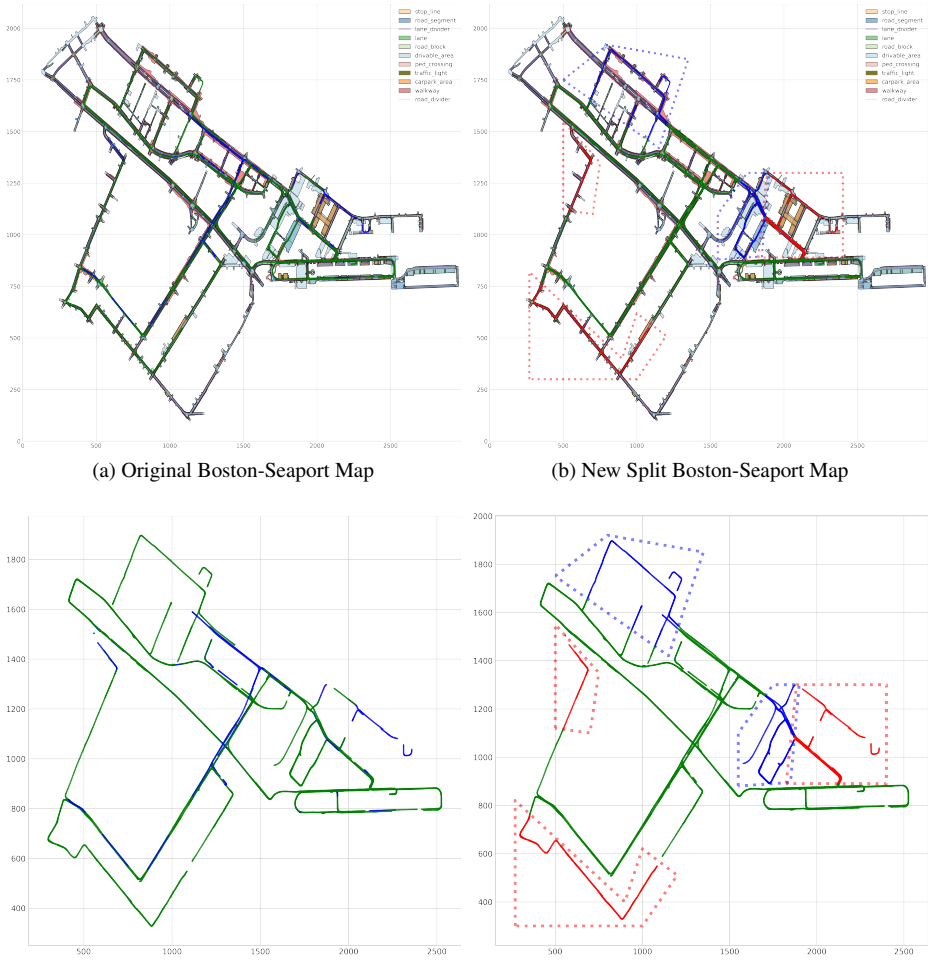
Figure 4: Qualitative results on confused road intersection scenario (multiple crossline at intersection).



Figure 5: Qualitative results on turning scenario that surrounded by vehicles.



Figure 6: Failure case: Qualitative results of complex prediction trajectories.



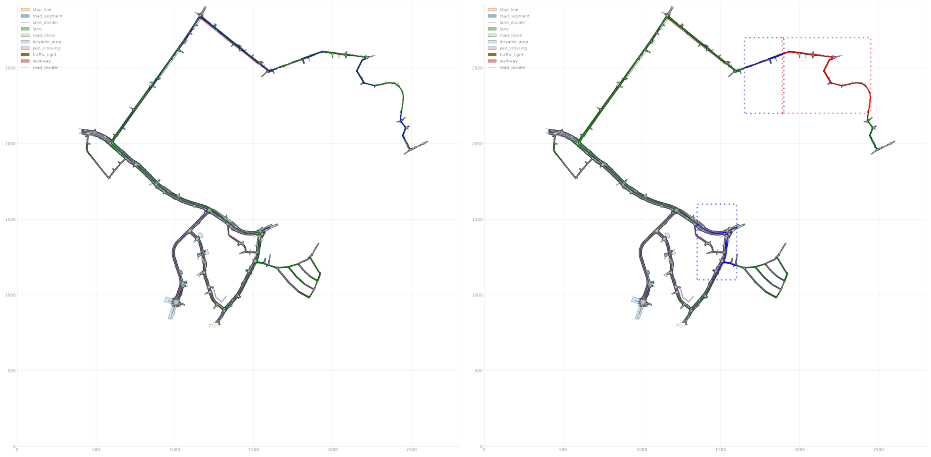
(a) Original Boston-Seaport Map

(b) New Split Boston-Seaport Map

(c) Original Boston-Seaport Data distribution

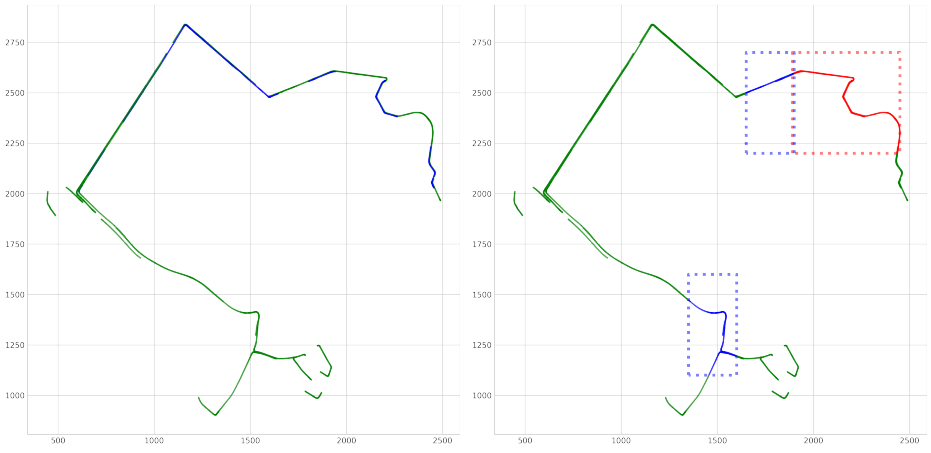
(d) New Split Boston-Seaport Data distribution

Figure 7: Comparison of original and new split datasets on the Boston Seaport map in nuScenes dataset. (a) and (c) are the original split visualization, (b) and (d) are the new split visualization. To ensure diversity in zone types within each set, regions from various parts of the city are included. Green: Train, Blue: Validation, Red: Test.



(a) Original Boston-Seaport Map

(b) New Split Boston-Seaport Map



(c) Original Boston-Seaport Data distribution

(d) New Split Boston-Seaport Data distribution

Figure 8: Comparison of original and new split datasets on the map in nuScenes dataset. (a) and (c) are the original split visualization, (b) and (d) are the new split visualization. To ensure diversity in zone types within each set, regions from various parts of the city are included. Green: Train, Blue: Validation, Red: Test.

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: The abstract and last paragraphs accurately reflect our paper's contributions and scope.

   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: The limitation is presented in the "Supplements" subsection of section "Appendix".

   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: [Yes]

Justification: The theoretical assumptions and proof are presented in section "Method".

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: The results of this paper can be reproduced with the code in supplement 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: [Yes]

Justification: We have attached code in supplement materials.

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, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: Due to the page limitation, we only put several important settings and parameters in the paper, the whole settings are included in our code.

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: [Yes]

Justification: We put the related information in the section "Experiments", "Appendix:Extension Experiments" and details are represented in our code.

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: The computer resources is identified in subsection " 5.2 Implementation Details".

   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: We have review the NeurIPS Code of Ethics. Our paper meets this requirement.

   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: [Yes]

    Justification: We discussed in "Social Impacts" section after Appendix.

    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: [N/A]

    Justification: [N/A]

    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: [Yes]

    Justification: Our paper does not have such risks. nuScenes and CARLA datasets were properly credited and their licenses respected.

    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: [Yes]

    Justification: We will open-source all components of this paper and ensure that only assets with appropriate usage permissions are cited and utilized.

    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: [N/A]

    Justification: [N/A]

    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: [N/A]

    Justification: [N/A]

    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: [N/A]

    Justification: We only use for checking grammar.

    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/2026/LLM`) for what should or should not be described.