NavigScene: Bridging Local Perception and Global Navigation for Beyond-Visual-Range Autonomous Driving

Qucheng Peng
Center for Research in Computer Vision, University of Central Florida
Orlando, FL, USA
qu935149@ucf.edu

Chen Bai
Xpeng Motors
Santa Clara, CA, USA
chenbai@xiaopeng.com

Guoxiang Zhang
Xpeng Motors
Santa Clara, CA, USA
guoxiangz@xiaopeng.com

Bo Xu
Xpeng Motors
Santa Clara, CA, USA

Xiaotong Liu
Xpeng Motors
Santa Clara, CA, USA

Xiaoyin Zheng
Xpeng Motors
Santa Clara, CA, USA

Chen Chen
Center for Research in Computer Vision, University of Central Florida
Orlando, FL, USA
chen.chen@crcv.ucf.edu

Cheng Lu
Xpeng Motors
Santa Clara, CA, USA
luc@xiaopeng.com

Abstract

Autonomous driving systems have made significant advances in Q&A, perception, prediction, and planning based on local visual information, yet they struggle to incorporate broader navigational context that human drivers routinely utilize. We address this critical gap between local sensor data and global navigation information by proposing NavigScene, an auxiliary navigation-guided natural language dataset that simulates a *human-like* driving environment within autonomous driving systems. Moreover, we develop three complementary paradigms to leverage NavigScene: (1) Navigation-guided Reasoning, which enhances vision-language models by incorporating navigation context into the prompting approach; (2) Navigation-guided Preference Optimization, a reinforcement learning method that extends Direct Preference Optimization to improve vision-language model responses by establishing preferences for navigation-relevant summarized information; and (3) Navigation-guided Vision-Language-Action model, which integrates navigation guidance and vision-language models with conventional driving models through feature fusion. Extensive experiments demonstrate that our approaches significantly improve performance across perception, prediction, planning, and question-answering tasks by enabling reasoning capabilities beyond visual range and improving generalization to diverse driving scenarios. This work represents a significant step toward more comprehensive autonomous driving systems capable of navigating complex, unfamiliar environments with greater reliability and safety.

MM '25, Dublin, Ireland
© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3755341

CCS Concepts

• **Computing methodologies → Artificial intelligence**.

Keywords

Vision-language Model, Reinforcement Learning, Autonomous Driving

ACM Reference Format:
Qucheng Peng, Chen Bai, Guoxiang Zhang, Bo Xu, Xiaotong Liu, Xiaoyin Zheng, Chen Chen, and Cheng Lu. 2025. NavigScene: Bridging Local Perception and Global Navigation for Beyond-Visual-Range Autonomous Driving. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3746027.3755341

1 Introduction

Autonomous driving systems [6, 8, 13, 16, 20, 26, 34, 50] have achieved remarkable progress, enabling vehicles to perceive surroundings, predict object movement, and plan actions. These systems fall into two categories: vision-language models (VLMs) [11, 18, 24, 29, 31, 35, 46, 48, 53] for question-answering and end-to-end driving models [7, 17, 38, 52] for perception, prediction, and planning. However, these approaches primarily rely on responses within visual range (typically 100-150 meters), creating a critical gap in incorporating global information for *human-like* long-term planning. This limitation constrains both VLMs and end-to-end models, hindering their ability to reason and generalize to unfamiliar scenes.

In real-world driving, navigation applications like Google Maps [39] provide essential global contextual information for human drivers. These applications communicate the ego vehicle's intended future maneuvers (e.g., turning left or right, proceeding straight) alongside three critical pieces of information: distance to upcoming maneuvers, intersection type, and presence of traffic signals.

[0] This work was done during Qucheng Peng's internship at Xpeng Motors' Silicon Valley office in Santa Clara, CA, USA.

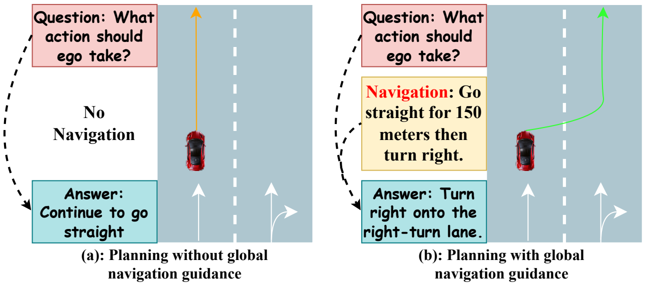

Question: What action should ego take?
No Navigation
Answer: Continue to go straight
(a): Planning without global navigation guidance
Question: What action should ego take?
Navigation: Go straight for 150 meters then turn right.
Answer: Turn right onto the right-turn lane.
(b): Planning with global navigation guidance

Figure 1: Comparison between a) planning without global navigation guidance and b) planning with global navigation guidance. In this example, the vehicle needs to turn right at the next corner. Without beyond-view-range (BVR) knowledge from navigation, the planner makes a conservative decision to continue straight. With global BVR knowledge, it appropriately directs the vehicle to merge into the right-turn lane. Concrete examples from experiments are shown in Fig. 7 and Fig. 8.

*Notably, the distance information typically extends beyond the visual perception capabilities of onboard sensors such as cameras or LiDAR, and is therefore classified as beyond visual range (**BVR**) [9, 25, 47] information.* Despite being crucial for effective planning and decision-making, BVR information remains largely unexplored in autonomous driving research. Current Q&A datasets [11, 35] and models [17, 38] predominantly focus on frame-by-frame perception and prediction, without adequately addressing the navigation context necessary for comprehensive scene understanding and long-term planning.

In Fig. 1, we demonstrate how navigation guidance enhances both question-answering performance and end-to-end planning. In this scenario, navigation provides critical information indicating an intersection 150 meters ahead where the ego vehicle must execute a right turn. However, due to the limited perception range of onboard sensors—typically 100-150 meters—the ego vehicle cannot detect this intersection with sufficient advance notice to initiate the necessary lane change. In contrast, by incorporating global BVR knowledge from navigation tools, the planner proactively directs the vehicle to merge into the right-turn lane well in advance, demonstrating the tangible benefits of navigation-guided planning.

To address this gap, we propose NavigScene, an auxiliary dataset derived from the nuScenes [5] and NAVSIM [10] datasets. Through natural language navigation instructions, we simulate a *human-like* driving environment within autonomous driving systems, effectively imitating navigation tools such as Google Maps that provide BVR knowledge critical for driving decisions and planning. Our dataset bridges the disconnection between local sensor data and global navigation context by providing paired data: multi-view sensor inputs (images or videos) alongside corresponding natural language navigation guidance that captures the global driving environment.

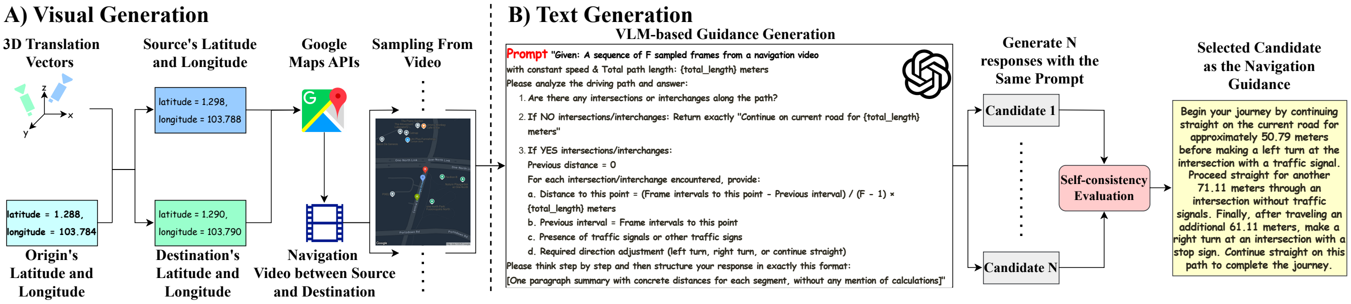
Building upon the *human-mimicking* auxiliary dataset NavigScene, we propose three paradigms to leverage navigation guidance in autonomous driving tasks like Q&A, perception, prediction and planning. First is navigation-guided reasoning, which can be implemented through Navigation-guided Supervised Fine-tuning (**NSFT**) for driving-related Q&A tasks. By incorporating navigation guidance into prompts, we enable comprehensive reasoning that considers both local visual cues and global navigational context, significantly improving the model's ability to answer questions requiring knowledge beyond the immediate visual range. Second is Navigation-guided Preference Optimization (**NPO**), a reinforcement learning method that introduces an auxiliary text summarization task to enhance Direct Preference Optimization (DPO) [33] by establishing preference relationships between summarized answers from vision-language models and navigation guidance, thereby improving BVR reasoning and generalization capabilities. Third is the Navigation-guided Vision-Language-Action (**NVLA**) model, which integrates navigation guidance and vision-language models with conventional end-to-end driving models through feature fusion, creating robust representations for downstream tasks including perception, prediction, and planning. Our contributions can be summarized in three main aspects:

- We propose NavigScene, a novel auxiliary dataset that pairs local multi-view sensor inputs with global natural language navigation guidance, addressing the critical gap between local perception and global navigation context in autonomous driving.
- We implement NavigScene across three complementary paradigms: navigation-guided reasoning, navigation-guided preference optimization, and a navigation-guided vision-language-action model, enhancing autonomous driving systems' reasoning and generalization capabilities beyond visual range limitations.
- We conduct comprehensive experiments on both Q&A tasks and end-to-end driving tasks—including perception, prediction, and planning—demonstrating the significant performance improvements achieved by incorporating global navigation knowledge into autonomous driving systems.

2 Related Works

LLMs and VLMs in Autonomous Driving. NuScenes-QA [32] is the first Visual Question Answering benchmark specifically designed for autonomous driving scenarios, establishing a foundation with several baselines that leverage advanced 3D detection [30] and VQA techniques. DriveGPT4 [48] introduces an interpretable end-to-end autonomous driving system powered by Large Language Models, while DriveLM [35] enhances Visual Language Models' reasoning capabilities through graph-based visual question answering. NuInstruct [11] places greater emphasis on crucial multi-view and temporal information in Visual Language Models, which is essential for robust autonomous driving systems. VLP [27, 54] proposes a novel framework that exploits LLMs to bridge the gap between linguistic understanding and autonomous driving.

End-to-end Autonomous Driving. VAD [17] employs an ego query mechanism to predict single-mode trajectories, while VADv2 [7] advances this approach by implementing a probabilistic space based on multiple trajectories. SparseDrive [38] innovates by designing parallel motion and planning modules that reduce computational demands from BEV features. DiffusionDrive [14, 19] introduces a truncated diffusion policy to enhance the trajectory's


A) Visual Generation
3D Translation Vectors
Source's Latitude and Longitude
Google Maps APIs
Sampling From Video
latitude = 1.298, longitude = 103.788
latitude = 1.288, longitude = 103.784
latitude = 1.290, longitude = 103.790
Origin's Latitude and Longitude
Destination's Latitude and Longitude
Navigation Video between Source and Destination
B) Text Generation
VLM-based Guidance Generation
Generate N responses with the Same Prompt
Candidate 1
Candidate N
Self-consistency Evaluation
Selected Candidate as the Navigation Guidance

Prompt "Given: A sequence of F sampled frames from a navigation video with constant speed & Total path length: {total_length} meters
Please analyze the driving path and answer:

1. Are there any intersections or interchanges along the path?
2. If NO intersections/interchanges: Return exactly "Continue on current road for {total_length} meters"
3. If YES intersections/interchanges:
   Previous distance = 0
   For each intersection/interchange encountered, provide:
   a. Distance to this point = (Frame intervals to this point - Previous interval) / (F - 1) × {total_length} meters
   b. Previous interval = Frame intervals to this point
   c. Presence of traffic signals or other traffic signs
   d. Required direction adjustment (left turn, right turn, or continue straight)

Please think step by step and then structure your response in exactly this format:
[One paragraph summary with concrete distances for each segment, without any mention of calculations]"

Begin your journey by continuing straight on the current road for approximately 50.79 meters before making a left turn at the intersection with a traffic signal. Proceed straight for another 71.11 meters through an intersection without traffic signals. Finally, after traveling an additional 61.11 meters, make a right turn at an intersection with a stop sign. Continue straight on this path to complete the journey.

Figure 2: Navigation guidance generation process of one scene. Part A (Visual Generation): Source and destination coordinates are calculated using the origin’s coordinate and 3D translation vectors. A navigation video is constructed via Google Maps APIs, then evenly sampled to extract multiple frames. Part B (Text Generation): The multiple frames are processed by a vision-language model (GPT-4o [1]) with a specialized prompt to generate several candidate responses. Self-consistency evaluation selects the highest-scoring candidate as the final navigation guidance.

probabilistic representation, while MomAD [36] focuses on improving stability and maintaining consistency across consecutive planning decisions.

3 Navigation-based Datasets: NavigScene

Existing autonomous driving datasets predominantly emphasize local-level descriptions, serving perception tasks effectively but inadequately addressing the beyond-visual-range (BVR) knowledge essential for scene understanding and decision making. To simulate *human-like* driving environments and bridge this gap, we propose NavigScene, an auxiliary navigation-guided dataset derived from nuScenes [5] and NAVSIM [10]. Our dataset provides paired multi-view sensor inputs alongside natural language navigation guidance that captures global driving context, enabling autonomous systems to reason with BVR knowledge in complex environments.

3.1 Visual Generation

To establish each scene, we first determine the latitudes and longitudes of both the source and destination. This calculation incorporates the origin’s coordinates and the 3D translation vectors of the source and destination from this origin. For an origin with coordinates (ϕ, λ), where ϕ represents latitude and λ represents longitude (both in decimal degrees), and a translation vector $(\Delta x, \Delta y, \Delta z)$ in meters (where Δx denotes the eastward component, Δy the northward component, and Δz the upward component), the coordinates of the source or destination (ϕ', λ') can be calculated using [37]:

$$\phi' = \phi + \frac{180}{\pi} \cdot \frac{\Delta y}{R}, \qquad \lambda' = \lambda + \frac{180}{\pi} \cdot \frac{\Delta x}{R \cdot \cos(\frac{\pi}{180} \cdot \phi)}, \tag{1}$$

where R represents the Earth’s radius, approximately 6,378,137 m. While Δx and Δy represent translations in the eastern and northern directions in meters, Δz is excluded from latitude and longitude calculations as it does not influence horizontal positioning.

We leverage Google Maps APIs [39] to generate navigation videos using these coordinates. The Direction API provides precise routes, the Static Map API acquires sequential images along routes, and the Distance Matrix API estimates driving distance and duration. Assuming constant velocity, we synthesize realistic navigation videos simulating the driving experience. To facilitate analysis, we evenly sample F frames from these videos for subsequent text generation via VLM in Sec. 3.2.

3.2 Text Generation

While Sec. 3.1 on visual generation (Part A in Fig. 2), integrating complete navigation videos with VLMs or end-to-end architectures poses significant challenges due to alignment difficulties between navigation videos and sensor-based training data. To address this limitation, we transform sequential images into natural language navigation descriptions using VLMs (Part B in Fig. 2). For each sequence of frames, we generate N candidate navigations through a specialized prompt shown in Fig. 2. This prompt first analyzes intersections or interchanges to determine driving directions, then estimates distances based on frame intervals.

After obtaining N candidate responses, we implement a novel selection strategy to identify the optimal description. We define three similarity metrics $S_{inter}(\cdot,\cdot)$, $S_{dist}(\cdot,\cdot)$, and $S_{word}(\cdot,\cdot)$:

$S_{inter}(\cdot,\cdot)$ represents intersection similarity, emphasizing directional keywords accuracy. For candidate a_i, directional keywords are extracted as $K_{inter}(a_i) = (m^i_1, m^i_2, \ldots)$, then the intersection similarity between a_i and a_j is:

$$S_{inter}(a_i, a_j) = \begin{cases} 1, & \text{if } |K_{inter}(a_i)| = |K_{inter}(a_j)| \\ & \text{and } m^i_d = m^j_d \text{ for all } d \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$S_{dist}(\cdot,\cdot)$ represents distance value similarity. Distance values in a_i are $K_{dist}(a_i) = (n^i_1, n^i_2, \ldots)$, then the distance similarity is:

$$S_{dist}(a_i, a_j) = \begin{cases} \mathbb{E}_{1 \le d \le |K_{dist}(a_i)|}\left[1 - \frac{|n^i_d - n^j_d|}{max(n^i_d, n^j_d)}\right], \text{otherwise} \\ 0, \text{if } |K_{dist}(a_i)| = |K_{dist}(a_j)| \end{cases} \tag{3}$$

$S_{word}(\cdot,\cdot)$ represents lexical similarity, calculated using the Jaccard index:

$$S_{word}(a_i, a_j) = |a_i \cap a_j| / |a_i \cup a_j|. \tag{4}$$

The overall similarity score $S_{over}(\cdot,\cdot)$ between candidates is:

$$S_{over}(a_i, a_j) = \eta_1 S_{inter}(a_i, a_j) + \eta_2 S_{dist}(a_i, a_j) + \eta_3 S_{word}(a_i, a_j) \tag{5}$$

Given that directional accuracy is most critical, followed by distance precision and then lexical similarity, we assign weights such that $\eta_1 > \eta_2 > \eta_3$, then select the optimal answer a^* by identifying the candidate with the highest cumulative similarity:

$$a^* = \arg\max_{a_i \in A} \sum_{j \neq i} S_{over}(a_i, a_j) \tag{6}$$

This approach identifies the best candidate, which serves as the final navigation to simulate *human-like* driving environment.

4 Methodology

4.1 Navigation-guided Reasoning

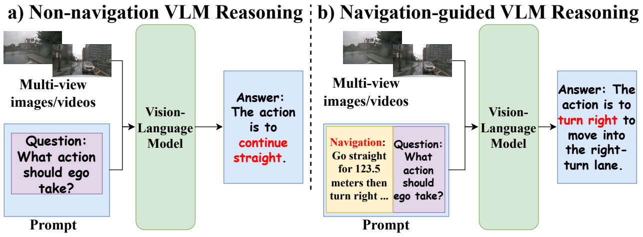

a) Non-navigation VLM Reasoning
b) Navigation-guided VLM Reasoning
Multi-view images/videos
Question: What action should ego take?
Prompt
Vision-Language Model
Answer: The action is to continue straight.
Multi-view images/videos
Navigation: Go straight for 123.5 meters then turn right ...
Question: What action should ego take?
Prompt
Vision-Language Model
Answer: The action is to turn right to move into the right-turn lane.

Figure 3: Comparison between a) non-navigation VLM reasoning and b) navigation-guided VLM reasoning. In our proposed navigation-guided paradigm, both the navigation guidance and question together form the prompt for VLM. (Best viewed when zoomed in.)

In traditional Q&A tasks for autonomous driving [11, 35, 45], multi-view images or videos paired with questions serve as input for VLM, as shown in Fig. 3a. However, this reasoning paradigm is limited to in-scope information and overlooks beyond-visual-range (BVR) information, which is crucial for long-term planning. To address this limitation, we incorporate navigation guidance into our prompting approach (as shown in Fig. 3b), thereby enriching the reasoning process with essential global information.

The navigation-guided reasoning can be represented as:

$$a = M(v, g \oplus q), \tag{7}$$

where M is the vision-language model, a represents natural language outputs as the answer, v denotes multi-view images or videos, g is the navigation guidance, and q is the question. The concatenation of g and q serves as the prompt for M.

4.2 Navigation-guided Preference Optimization

While Section 4.1 introduced NSFT to enhance VLM reasoning capabilities, this approach has limitations in generalizing to unseen scenarios. For VLMs with fewer than 10B parameters, supervised fine-tuning restricts generalization ability [12], and reasoning capacity is constrained by parameter scale [4]. To address these shortcomings, we propose Navigation-guided Preference Optimization (**NPO**), an extension of Direct Preference Pptimization (DPO) [33] applied after NSFT to improve generalization performance on novel navigation scenarios.

NPO builds upon DPO by integrating navigation-related knowledge through an auxiliary text summarization task. In NPO, the VLM still processes multi-view images v and question q as inputs. We establish a preference relationship between detailed answer a and its summarized version s. The summarization s is generated online using a VLM M:

$$s = M(\text{pt} \oplus a), \tag{8}$$

where a is the original answer from supervised fine-tuning, and pt is the prompt *"Summarize this answer to a driving-relevant question to make it simple without losing important information."*

Following DPO methodology, we initialize a learnable reward model M^θ and a frozen reference model M^*. Thus, we can obtain s^θ and s^* from these two VLMs respectively. With navigation guidance g, we quantify the quality of the summarized answer s using the mutual information [40, 43]:

$$\begin{aligned} \text{mi}(s) &= p(a,s)\log\frac{p(a,s)}{p(a)p(s)} - p(s,g)\log\frac{p(s,g)}{p(s)p(g)}, \\ &= -\log p(s) - p(s)p(g|s)\log\frac{p(g|s)}{p(g)}. \end{aligned} \tag{9}$$

Eq. 9 has two goals: to simplify the summarized answer s compared to the original answer a, while enhancing the relevance between the summarized answer s and the navigation guidance g. Based on the implementation in [42, 44], Equation 9 is further simplified to:

$$\text{mi}(s) = -\log p(s) - p(s)\log p(g|s). \tag{10}$$

By incorporating this measurement, we define the reward for summarized answers as:

$$r_s = \log p^\theta(s^\theta|v,q) - \log p^*(s^*|v,q) + \alpha[\text{mi}(s^\theta) - \text{mi}(s^*)], \tag{11}$$

where α is a trade-off hyper parameter. This reward not only measures the difference in summarized answer s between reward model M^θ and reference model M^*, but also the difference in guidance relevance between the two summarized answers.

Similarly, the reward for the original answer a is:

$$r_a = \log p^\theta(a|v,q) - \log p^*(a|v,q). \tag{12}$$

The objective function of NPO is thus formulated as:

$$\mathcal{L}_{\text{NPO}}(\theta) = -\mathbb{E}_{(v,q,a,s^\theta,s^*)\in D}[\log\sigma(r_s - r_a)] \tag{13}$$

where σ is the sigmoid function, and D represents the preference dataset that contains tuples of (multi-view images, question, answer, summary from reward model, summary from reference model).

In our proposed NPO method, we introduce an auxiliary task, navigation-guided text summarization, for both the reward model and the reference model. This strategic addition directs the reward model to focus on guidance-relevant knowledge, significantly enhancing its ability to generate driving-relevant, concise responses while preserving critical information aligned with navigation.

4.3 Navigation-guided Vision-Language-Action Model

Beyond question-answering tasks, navigation guidance substantially enhances end-to-end driving system's performance. Conventional end-to-end models [17, 38] that rely solely on sensor data (multi-view images or videos) suffer from limited reasoning capabilities and poor generalization to novel scenarios. To address these

Table 1: Reasoning Results on NuInstruct

VLM	NavigScene	Perception						Prediction		Risk						Planning ↑
		Dis ↓	Spe ↓	Ins ↓	Clo ↑	Sta ↑	SaR ↑	Mot ↓	Sta ↑	App ↑	Lan ↑	Onc ↑	Cro ↑	Ove ↑	Bra ↑	
Llama-Adapter	×	27.9	6.6	6.5	20.4	16.2	24.0	8.7	40.2	13.5	18.7	16.2	18.4	6.5	21.5	25.7
	✓	24.3	3.9	4.0	32.2	19.8	28.7	4.3	44.1	15.8	21.0	19.9	22.8	9.0	26.4	31.2
Llava-7B	×	28.4	6.9	6.6	22.1	16.2	23.5	9.4	38.9	12.3	19.6	16.9	18.7	6.5	21.3	25.3
	✓	24.5	4.3	4.1	27.6	19.5	27.8	6.3	43.6	15.0	22.7	20.2	23.0	9.2	26.8	32.6
Qwen2.5-7B	×	26.9	5.6	5.4	26.5	17.3	26.7	7.8	41.8	15.2	20.3	18.4	20.5	7.7	22.1	26.6
	✓	23.6	3.1	3.2	33.7	20.2	31.9	3.8	45.3	18.6	23.7	21.9	26.2	10.4	27.9	36.4

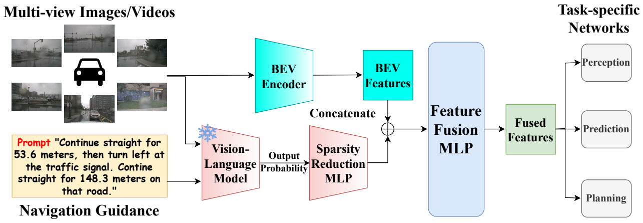


Figure 4: Navigation-guided vision-language-action model for end-to-end driving. BEV features are concatenated with vision-language features generated by the frozen VLM and a learnable sparsity reduction MLP, then processed through a learnable feature fusion MLP to produce fused features for task-specific networks.

Table 2: Reasoning Results on DriveLM-nuScenes

VLM	NavigScene	BLEU-4 ↑	METEOR ↑	CIDEr ↑	ROUGE_L ↑	SPICE ↑	GPT ↑	Comp. ↑
Llama-Adapter	×	50.68	33.75	2.37	64.59	44.20	70.83	29.98
	✓	54.25	37.62	2.81	67.66	48.35	74.08	33.07
Llava-7B	×	49.75	33.21	2.19	63.84	42.56	70.24	29.37
	✓	53.93	36.86	2.75	66.97	46.83	73.77	32.79
Qwen2.5-7B	×	51.65	34.12	2.46	64.97	46.45	71.29	30.31
	✓	55.13	38.20	3.14	67.88	49.89	74.87	34.26

shortcomings, we propose a Navigation-guided Vision-Language-Action (**NVLA**) model that integrates navigation guidance with vision-language models into the end-to-end driving framework.

In conventional end-to-end models, the output of a driving task $j \in \{\text{perception}, \text{prediction}, \text{planning}\}$ can be represented as:

$$o_j^{\text{con}} = H_j(E(v)), \tag{14}$$

where v denotes multi-view images or videos, E is the BEV encoder, H_j is the task-specific network, and o_j^{con} is the output of task j without navigation guidance.

In our navigation-guided VLA model, we incorporate both a VLM post-trained by NSFT and NPO, and navigation guidance. The output probability distribution of modern VLMs typically has a high dimensionality due to their large vocabulary space—for example, LlamaAdapter's [51] output probability dimension is 32,000—making direct alignment or fusion with BEV features (typically 256 dimensions in models like SparseDrive [38]) challenging. To address this mismatch, we introduce a learnable sparsity reduction MLP ϕ^{red} to compress VLM features' dimension, followed by a learnable feature fusion MLP ϕ^{fus}. The complete process is represented as:

$$o_j^{\text{nav}} = H_j(\phi^{\text{fus}}(E(v) \oplus \phi^{\text{red}}(M(v, g)))), \tag{15}$$

where M represents the frozen VLM that has been trained via NSFT as described in Sec. 4.1 then NPO as described in Sec. 4.2, g is the navigation guidance, and o_j^{nav} is the output of task j with navigation guidance. This integration process is illustrated in Fig. 4.

5 Experiments

Datasets. We evaluate NavigScene on two benchmark categories: Q&A datasets for VLM training and end-to-end driving datasets. For Q&A, we use DriveLM-nuScenes [35] (700 scenes with multi-view questions, 200 for testing) and NuInstruct [11] (10,000+ pairs). For end-to-end evaluation, we employ nuScenes [5] (1,000 scenes, 6 cameras/LiDAR/5 radars) with NavigScene-nuScenes, and NAVSIM [10] (120 hours, 8 cameras/5 LiDAR) with NavigScene-NAVSIM.

Implementations. For NavigScene generation, we set $F = 20$, $N = 5$, with weights $\eta_1 = 0.5$, $\eta_2 = 0.3$, $\eta_3 = 0.2$. We evaluate Llama-Adapter-7B [51], Llava-v1.6-Mistral-7B [22], and Qwen2.5-VL-Instruct-7B [49] using LoRA (rank 16, lr=1e-4, $\alpha = 0.6$) with 128-token output limit. For end-to-end models VAD [17] and SparseDrive [38], feature fusion MLPs use lr=2e-4 with AdamW optimizer [23]. NPO training uses 10 epochs, while NSFT and NVLA follow original schedules. All experiments run thrice on Nvidia H800 GPUs.

5.1 Quantitative Results on Q&A Tasks

In Tab. 1, we present NuInstruct [11] results comparing baseline VLMs with NavigScene post-trained models. Results show NavigScene significantly improves performance across all tested VLMs for driving-related Q&A tasks.

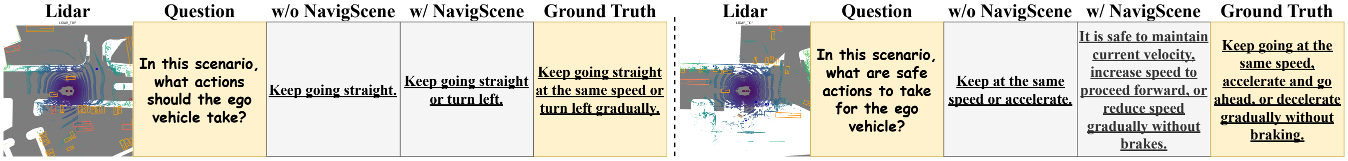
In Tab. 2, we compare VLMs post-trained with NavigScene via NSFT and NPO against DriveLM [35] baselines. Using standard metrics (BLEU-4 [28], METEOR [3], CIDEr [41], ROUGE_L [21], SPICE [2], GPT score [1, 35], and completeness [35]), NavigScene consistently enhances driving-relevant response quality across all VLMs.

5.2 Qualitative Results on Q&A Tasks

In Fig. 5 and Fig. 6, we show examples of VLM responses both with and without NavigScene integration. The BVR knowledge provided by NavigScene significantly enhances the VLM's reasoning capabilities, resulting in more complete and accurate answers.

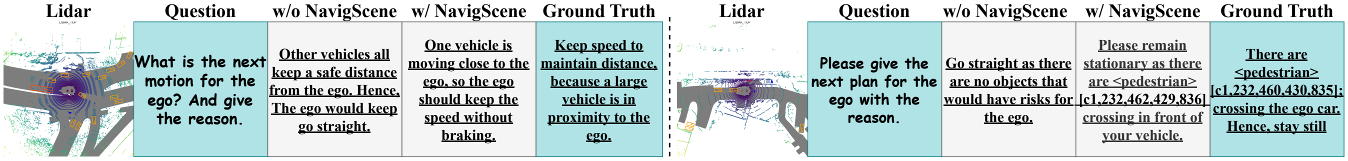
5.3 Quantitative Results on End-to-end Driving

In Tab. 3, we compare end-to-end driving configurations with different VLAs against original models for open-loop and closed-loop planning. For open-loop evaluation using UniAD protocols [15], integrating VLMs with NavigScene substantially enhances performance, with Qwen2.5-7B showing significant improvements in L2 and collision metrics. For closed-loop evaluation using NC, DAC, TTC, Comf., EP, and PDMS metrics, NavigScene integration



Lidar	Question	w/o NavigScene	w/ NavigScene	Ground Truth
	In this scenario, what actions should the ego vehicle take?	Keep going straight.	Keep going straight or turn left.	Keep going straight at the same speed or turn left gradually.
	In this scenario, what are safe actions to take for the ego vehicle?	Keep at the same speed or accelerate.	It is safe to maintain current velocity, increase speed to proceed forward, or reduce speed gradually without brakes.	Keep going at the same speed, accelerate and go ahead, or decelerate gradually without braking.

Figure 5: Examples of question-answering on the DriveLM dataset.



Lidar	Question	w/o NavigScene	w/ NavigScene	Ground Truth
	What is the next motion for the ego? And give the reason.	Other vehicles all keep a safe distance from the ego. Hence, The ego would keep go straight.	One vehicle is moving close to the ego, so the ego should keep the speed without braking.	Keep speed to maintain distance, because a large vehicle is in proximity to the ego.
	Please give the next plan for the ego with the reason.	Go straight as there are no objects that would have risks for the ego.	Please remain stationary as there are <pedestrian> [c1,232,462,429,836] crossing in front of your vehicle.	There are <pedestrian> [c1,232,460,430,835]; crossing the ego car. Hence, stay still

Figure 6: Examples of question-answering on the NuInstruct dataset.

Table 3: Results for both open-loop and closed-loop planning settings. Left: Open-loop planning performance on nuScenes. Right: Closed-loop planning performance on NAVSIM. All VLMs underwent post-training via NSFT then NPO on DriveLM-nuScenes and NavigScene-nuScenes, followed by prompting with NavigScene-nuScenes and NavigScene-NAVSIM respectively during training.

End-to-end Model	VLM	Open-loop Planning on nuScenes								Closed-loop Planning on NAVSIM					
		L2 (m)↓				Collision Rate (‱) ↓				NC ↑	DAC ↑	TTC ↑	Comf. ↑	EP ↑	PDMS ↑
		1s	2s	3s	Avg.	1s	2s	3s	Avg.						
VAD	None	0.54	1.15	1.98	1.22	9.76	24.20	95.93	43.30	97.0	86.5	89.7	100	75.4	80.6
	Llama-Adapter	0.42	1.02	1.76	1.07	6.28	21.85	81.37	36.50	97.9	88.8	91.6	100	77.6	82.2
	Llava-7B	0.45	1.04	1.79	1.09	6.93	22.09	83.41	37.48	97.7	88.1	91.3	100	76.8	81.8
	Qwen2.5-7B	0.37	0.96	1.65	0.99	4.52	18.10	74.62	32.41	98.3	90.6	92.4	100	79.0	84.0
Sparse Drive	None	0.44	0.92	1.69	1.01	7.38	19.46	70.60	32.48	97.2	91.7	91.4	100	77.9	82.4
	Llama-Adapter	0.33	0.78	1.47	0.86	5.97	15.65	58.41	26.68	98.0	93.5	93.0	100	80.4	83.9
	Llava-7B	0.36	0.80	1.46	0.87	6.26	16.34	62.58	28.39	98.1	93.3	92.8	100	78.6	83.1
	Qwen2.5-7B	0.29	0.64	1.35	0.76	4.38	12.99	44.75	20.71	98.3	96.0	94.1	100	81.7	86.5

Table 4: Object detection, tracking, mapping, and motion forecasting on nuScenes. All VLMs were first post-trained via NSFT then NPO on DriveLM-nuScenes and NavigScene-nuScenes, then prompted using NavigScene-nuScenes during training.

End-to-end Model	VLM	Detection				Tracking				Mapping				Motion Forecasting			
		mAP ↑	mATE ↓	mASE ↓	NDS ↑	AMOTA ↑	AMOTP ↓	Recall ↑	IDS ↓	AP_{ped} ↑	AP_{div} ↑	AP_{bou} ↑	mAP ↑	mADE ↓	mFDE ↓	MR ↓	EPA ↑
VAD	None	0.27	0.70	0.30	0.39	-	-	-	-	40.6	51.5	50.6	47.6	0.78	1.07	0.121	0.598
	Llama-Adapter	0.33	0.58	0.28	0.44	-	-	-	-	43.0	53.6	53.7	50.1	0.73	1.04	0.112	0.605
	Llava-7B	0.30	0.61	0.29	0.41	-	-	-	-	42.7	52.9	53.4	49.7	0.75	1.04	0.116	0.601
	Qwen2.5-7B	0.36	0.56	0.27	0.44	-	-	-	-	47.1	54.2	55.2	52.2	0.69	0.98	0.110	0.609
SparseDrive	None	0.42	0.57	0.28	0.53	0.39	1.25	0.50	886	49.9	57.0	58.4	55.1	0.62	0.99	0.136	0.482
	Llama-Adapter	0.45	0.52	0.25	0.56	0.43	1.22	0.53	864	52.3	58.1	58.9	56.4	0.60	0.94	0.131	0.489
	Llava-7B	0.43	0.52	0.27	0.54	0.42	1.22	0.52	879	51.8	58.0	58.6	56.1	0.61	0.97	0.133	0.487
	Qwen2.5-7B	0.46	0.50	0.24	0.57	0.45	1.20	0.53	857	55.0	58.5	59.3	57.6	0.58	0.92	0.129	0.498

significantly improves performance, particularly in DAC, EP, and PDMS—metrics strongly correlated with human-like driving and navigation interpretation.

In Tab. 4, we compare configurations across detection, tracking, mapping, and motion forecasting tasks. NavigScene enhances performance even in non-planning tasks, achieving detection mAP improvements of 0.09 over VAD and 0.04 over SparseDrive with Qwen2.5-7B.

5.4 Qualitative Results on End-to-end Driving

In Fig. 7, we present two open-loop planning examples comparing performance with and without NavigScene integration. Leveraging beyond-view-range knowledge from NavigScene enables the autonomous driving system to generate more accurate driving commands and route planning. This is particularly evident in the right case, where the vehicle correctly anticipates a right turn by initiating an early lane change with the help of global navigation guidance.

In Fig. 8, we present two closed-loop planning examples demonstrating the impact of NavigScene integration. In the left example, the vehicle intends to switch to the fast track but should continue straight. Without navigation guidance, the driving model incorrectly turns left. Similarly, in the right example, the vehicle needs to turn right at the upcoming intersection. Without NavigScene,

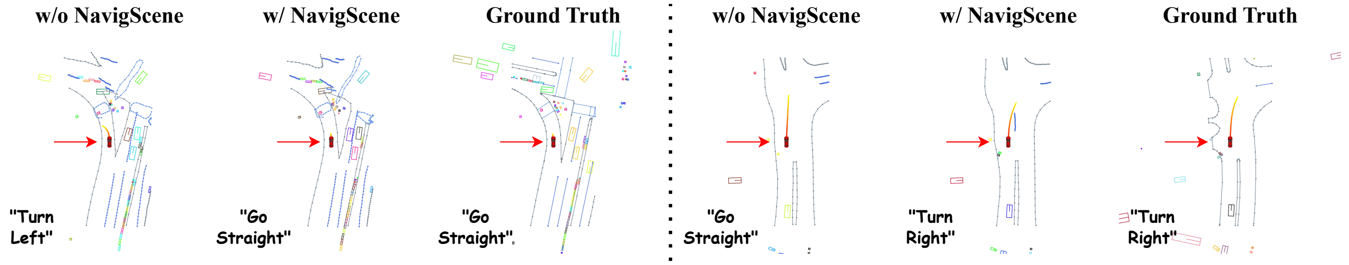


Figure 7: BEV visualization of open-loop planning on nuScenes dataset. "w/o NavigScene" indicates the original SparseDrive, and "w/ NavigScene" is the VLA model integrates Qwen2.5-7B (post-trained on DriveLM and NavigScene via NSFT and NPO) with SparseDrive. Arrows point at ego vehicles, and text in bottom-left displays predicted driving commands, and orange curves represent predicted driving routes. Left: Vehicle proceeds straight and reduces speed to stop. Right: Vehicle anticipates a right turn by initiating an early lane change.

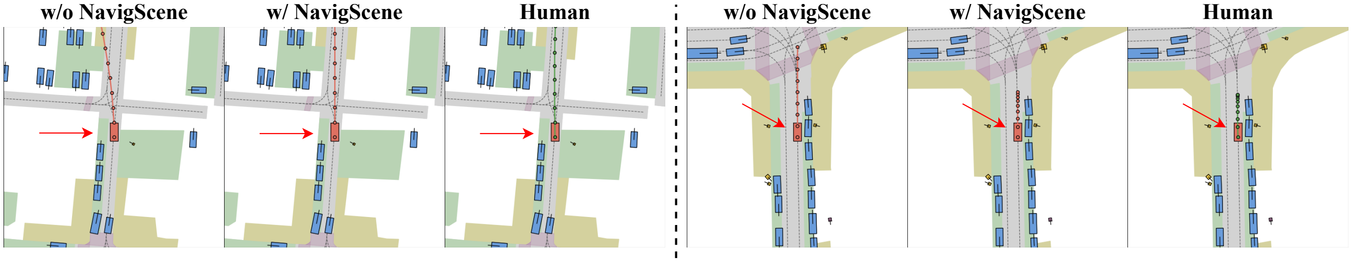


Figure 8: BEV Visualization of closed-loop planning on NAVSIM dataset. w/o NavigScene indicates the original SparseDrive, and w/ NavigScene is the VLA model integrates Qwen2.5-7B (post-trained on DriveLM and NavigScene via NSFT and NPO) with SparseDrive. Arrows point at ego vehicles, with orange routes generated by models and green routes showing human operations. Left: Vehicle anticipates a left turn by initiating an early lane change. Right: Vehicle anticipates a right turn by slowing down and waiting.

Table 5: Ablation study on DriveLM-nuScenes

VLM	NSFT	NPO	BLEU-4 ↑	METEOR ↑	CIDEr ↑	ROUGE_L ↑	SPICE ↑	GPT ↑	Comp. ↑
Llama-Adapter	×	×	50.68	33.75	2.37	64.59	44.20	70.83	29.98
	✓	×	52.20	35.46	2.45	66.53	45.79	72.18	31.74
	×	✓	51.73	34.89	2.53	66.20	46.38	71.46	31.44
	✓	✓	54.25	37.62	2.81	67.66	48.35	74.08	33.07
Llava-7B	×	×	49.75	33.21	2.19	63.84	42.56	70.24	29.37
	✓	×	51.66	34.54	2.41	65.28	44.15	71.59	30.62
	×	✓	50.84	34.07	2.35	64.86	44.01	71.19	30.46
	✓	✓	53.93	36.86	2.75	66.97	46.83	73.77	32.79
Qwen2.5-7B	×	×	51.65	34.12	2.46	64.97	46.45	71.29	30.31
	✓	×	53.47	35.95	2.77	66.23	47.84	73.15	32.08
	×	✓	52.97	36.04	2.82	65.55	47.36	73.08	31.99
	✓	✓	55.13	38.20	3.14	67.88	49.89	74.87	34.26

Table 6: Cross-city generalization results on nuScenes. All VLMs were first conducted NSFT on DriveLM-nuScenes and NavigScene-nuScenes.

End-to-end Model	VLM	NPO	Boston → Singapore		Singapore → Boston	
			Avg. L2 (m) ↓	Avg. Col (‱) ↓	Avg. L2 (m) ↓	Avg. Col (‱) ↓
VAD	None	N/A	0.86	26.83	0.63	20.44
	Llama-Adapter	×	0.94	27.08	0.75	21.19
		✓	0.75	23.29	0.64	19.30
	Qwen2.5-7B	×	0.97	27.51	0.81	21.85
		✓	0.70	22.55	0.61	18.46
SparseDrive	None	N/A	0.97	30.17	0.84	33.62
	Llama-Adapter	×	1.04	27.84	0.87	34.48
		✓	0.88	27.32	0.72	20.99
	Qwen2.5-7B	×	1.11	28.06	0.93	35.64
		✓	0.82	24.70	0.69	19.66

it continues straight for an extended distance, whereas with NavigScene, it appropriately slows down and waits for an opportunity to turn right.

5.5 Cross-city End-to-end Generalization

Table 6 presents cross-city generalization results on nuScenes, examining Boston ⟶ Singapore and Singapore ⟶ Boston transfer tasks following [27]. VLA models with NPO consistently outperform both original end-to-end architectures and VLA models without NPO, demonstrating NPO's effectiveness in enhancing robustness when navigating unfamiliar urban environments with different traffic patterns and infrastructures.

5.6 Ablation Study on Q&A Tasks

In Tables 5 and 7, we present ablation studies conducted on DriveLM-nuScenes and NuInstruct datasets, respectively. We examine four experimental settings: (1) without NSFT and NPO, where the VLM is finetuned without NavigScene; (2) with NSFT only, where the VLM undergoes fine-tuning with NavigScene; (3) with NPO only, where the VLM is first finetuned without NavigScene and then trained using NPO and NavigScene; and (4) with both NSFT and NPO, where the VLM is first finetuned with NavigScene and subsequently trained with NPO and NavigScene.

Both DriveLM-nuScenes and NuInstruct datasets demonstrate consistent performance improvements when applying NSFT and NPO across all three VLMs. The most significant gains occur when both techniques are combined, with Qwen2.5-7B achieving the

Table 7: Ablation study on NuInstruct

VLM	NSFT	NPO	Perception						Prediction		Risk						Planning ↑
			Dis ↓	Spe ↓	Ins ↓	Clo ↑	Sta ↑	SaR ↑	Mot ↓	Sta ↑	App ↑	Lan ↑	Onc ↑	Cro ↑	Ove ↑	Bra ↑	
Llama-Adapter	×	×	27.9	6.6	6.5	20.4	16.2	24.0	8.7	40.2	13.5	18.7	16.2	18.4	6.5	21.5	25.7
	✓	×	25.8	5.3	5.5	24.9	17.4	26.1	7.2	41.8	14.4	20.1	17.9	20.7	7.6	23.9	27.7
	×	✓	26.0	5.5	5.7	24.1	17.0	25.8	7.5	42.2	14.6	19.2	17.3	19.9	7.5	22.8	27.4
	✓	✓	24.3	3.9	4.0	32.2	19.8	28.7	4.3	44.1	15.8	21.0	19.9	22.8	9.0	26.4	31.2
Llava-7B	×	×	28.4	6.9	6.6	22.1	16.2	23.5	9.4	38.9	12.3	19.6	16.9	18.7	6.5	21.3	25.3
	✓	×	26.3	5.8	5.4	24.7	17.5	25.1	8.0	40.4	13.6	20.5	18.0	20.4	7.7	23.0	27.2
	×	✓	26.9	5.5	5.1	24.4	17.1	24.7	8.6	39.9	13.8	20.7	18.5	20.0	7.3	23.2	27.0
	✓	✓	24.5	4.3	4.1	27.6	19.5	27.8	6.3	43.6	15.0	22.7	20.2	23.0	9.2	26.8	32.6
Qwen2.5-7B	×	×	26.9	5.6	5.4	26.5	17.3	26.7	7.8	41.8	15.2	20.3	18.4	20.5	7.7	22.1	26.6
	✓	×	25.2	4.4	4.1	29.8	18.9	28.0	6.5	43.7	17.1	21.9	20.3	23.2	8.4	24.5	29.9
	×	✓	25.5	4.8	4.5	29.3	18.4	28.3	6.9	43.2	16.8	21.5	20.4	23.4	8.1	24.2	29.7
	✓	✓	23.6	3.1	3.2	33.7	20.2	31.9	3.8	45.3	18.6	23.7	21.9	26.2	10.4	27.9	36.4

Table 8: Ablation study on both open-loop and closed-loop planning settings. Left: Open-loop planning performance on nuScenes. Right: Closed-loop planning performance on NAVSIM. All VLMs' NSFT and NPO are based on DriveLM-nuScenes and NavigScene-nuScenes.

End-to-end Model	VLM	NSFT	NPO	Open-loop Planning on nuScenes								Closed-loop Planning on NAVSIM					
				L2 (m)↓				Collision Rate (‱) ↓				NC ↑	DAC ↑	TTC ↑	Comf. ↑	EP ↑	PDMS ↑
				1s	2s	3s	Avg.	1s	2s	3s	Avg.						
VAD	None	N/A	N/A	0.54	1.15	1.98	1.22	9.76	24.20	95.93	43.30	97.0	86.5	89.7	100	75.4	80.6
	Llama-Adapter	×	×	0.69	1.48	2.64	1.60	11.35	28.64	104.57	48.19	95.6	84.2	87.1	99.8	72.5	78.2
		✓	×	0.50	1.07	1.88	1.15	8.53	22.84	89.77	40.38	97.3	87.7	90.2	100	76.6	81.3
		×	✓	0.52	1.12	1.90	1.18	8.79	23.01	91.24	41.01	97.2	87.4	90.0	100	75.9	81.0
		✓	✓	0.42	1.02	1.76	1.07	6.28	21.85	81.37	36.50	97.9	88.8	91.6	100	77.6	82.2
	Qwen2.5-7B	×	×	0.66	1.50	2.58	1.58	12.48	29.96	107.25	49.90	95.3	84.4	86.6	99.5	72.8	77.8
		✓	×	0.45	1.03	1.82	1.10	7.24	22.38	85.69	38.44	97.5	88.0	90.4	100	76.9	81.5
		×	✓	0.49	1.06	1.85	1.12	7.85	22.46	87.33	39.21	97.6	88.4	91.1	100	77.2	81.9
		✓	✓	0.37	0.96	1.65	0.99	4.52	18.10	74.62	32.41	98.3	90.6	92.4	100	79.0	84.0
SparseDrive	None	N/A	N/A	0.44	0.92	1.69	1.01	7.38	19.46	70.60	32.48	97.2	91.7	91.4	100	77.9	82.4
	Llama-Adapter	×	×	0.62	1.38	2.40	1.47	10.53	25.57	98.19	44.76	96.1	85.9	87.6	100	73.3	78.7
		✓	×	0.39	0.87	1.62	0.96	7.10	18.35	65.99	30.48	97.6	92.2	91.7	100	78.2	82.7
		×	✓	0.42	0.89	1.66	0.99	7.25	18.67	68.23	31.38	97.6	92.6	92.3	100	78.6	82.9
		✓	✓	0.33	0.78	1.47	0.86	5.97	15.65	58.41	26.68	98.0	93.5	93.0	100	80.4	83.9
	Qwen2.5-7B	×	×	0.58	1.27	2.35	1.40	10.06	24.94	97.83	44.28	96.4	86.6	88.1	99.8	73.4	79.0
		✓	×	0.38	0.87	1.60	0.95	6.97	17.84	64.41	29.74	97.8	92.3	92.0	100	79.0	83.6
		×	✓	0.40	0.88	1.66	0.98	7.03	18.06	67.69	30.93	97.8	92.0	91.8	100	78.9	83.5
		✓	✓	0.29	0.64	1.35	0.76	4.38	12.99	44.75	20.71	98.3	96.0	94.1	100	81.7	86.5

highest overall scores across metrics in both datasets. For DriveLM-nuScenes, the NSFT+NPO combination substantially improved BLEU-4, METEOR, and CIDEr scores, with Qwen2.5-7B reaching the best CIDEr score of 3.14. In NuInstruct, this combined approach led to strong reductions in Dis and Spe metrics. These results suggest that NSFT and NPO collaboration enhances VLMs' ability to understand complex driving scenarios and generate appropriate responses.

5.7 Ablation Study on Open-loop and Closed-loop Planning

In Tab. 8, we present ablation studies on nuScenes (open-loop) and NAVSIM (closed-loop) examining five settings: (1) without VLM; (2) without NSFT/NPO (VLM finetuned without NavigScene); (3) NSFT only; (4) NPO only; and (5) both NSFT and NPO. All VLMs are frozen and connected with end-to-end models to construct VLA models.

Table 8 shows clear performance improvements when incorporating NSFT and NPO in both open-loop and closed-loop planning across VAD and SparseDrive models. SparseDrive generally outperforms VAD, while Qwen2.5-7B consistently outperforms Llama-Adapter. VLM integration with NSFT+NPO provides substantial gains over baselines, particularly reducing collision rates and improving trajectory accuracy, demonstrating that navigation-specific fine-tuning followed by preference optimization creates the most effective autonomous driving systems.

6 Conclusion

In this paper, we address a critical limitation in current autonomous driving systems: the disconnection between local sensor data and global navigation context. First we introduced NavigScene, an auxiliary navigation-guided natural language dataset that bridges this gap by simulating *human-like* driving environments. Besides, through three complementary paradigms based on NavigScene: Navigation-guided Reasoning, Navigation-guided Preference Optimization, and Navigation-guided Vision-Language-Action model, we achieve significant improvements in driving-related tasks across Q&A, perception, prediction, and planning. We enable reasoning capability beyond visual range and enhance generalization ability to diverse driving scenarios. In a word, this work brings autonomous driving systems closer to *human-like* ability to navigate complex, unfamiliar environments with improved reliability and safety.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[2] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V 14*. Springer, 382–398.

[3] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. 65–72.

[4] Jing Bi, Yuting Wu, Weiwei Xing, and Zhenjie Wei. 2025. Enhancing the Reasoning Capabilities of Small Language Models via Solution Guidance Fine-Tuning. In *Proceedings of the 31st International Conference on Computational Linguistics*. 9074–9084.

[5] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11621–11631.

[6] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. 2024. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[7] Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. 2024. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243* (2024).

[8] Pranav Singh Chib and Pravendra Singh. 2023. Recent advancements in end-to-end autonomous driving using deep learning: A survey. *IEEE Transactions on Intelligent Vehicles* 9, 1 (2023), 103–118.

[9] Joao PA Dantas, Marcos ROA Maximo, and Takashi Yoneyama. 2023. Autonomous agent for beyond visual range air combat: A deep reinforcement learning approach. In *Proceedings of the 2023 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation*. 48–49.

[10] Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, et al. 2024. Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. *Advances in Neural Information Processing Systems* 37 (2024), 28706–28719.

[11] Xinpeng Ding, Jianhua Han, Hang Xu, Xiaodan Liang, Wei Zhang, and Xiaomeng Li. 2024. Holistic autonomous driving understanding by bird's-eye-view injected multi-modal large models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13668–13677.

[12] Guanting Dong, Hongyi Yuan, Keming Lu, Chengpeng Li, Mingfeng Xue, Dayiheng Liu, Wei Wang, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2024. How Abilities in Large Language Models are Affected by Supervised Fine-tuning Data Composition. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 177–198.

[13] Yulin He, Siqi Wang, Wei Chen, Tianci Xun, and Yusong Tan. 2024. Sniffing Threatening Open-World Objects in Autonomous Driving by Open-Vocabulary Models. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 9067–9076.

[14] Yangfan He, Xinyan Wang, and Tianyu Shi. 2024. Ddpm-moco: Advancing industrial surface defect generation and detection with generative and contrastive learning. In *International Joint Conference on Artificial Intelligence*. Springer, 34–49.

[15] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. 2023. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 17853–17862.

[16] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. [n. d.]. Bench2Drive: Towards Multi-Ability Benchmarking of Closed-Loop End-To-End Autonomous Driving. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[17] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. 2023. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8340–8350.

[18] Xiang Li, Yangfan He, Shuaishuai Zu, Zhengyang Li, Tianyu Shi, Yiting Xie, and Kevin Zhang. 2025. Multi-Modal Large Language Model with RAG Strategies in Soccer Commentary Generation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 6197–6206.

[19] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, and Xinggang Wang. 2025. DiffusionDrive: Truncated Diffusion Model for End-to-End Autonomous Driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[20] Wenlong Liao, Sunyuan Qiang, Xianfei Li, Xiaolei Chen, Haoyu Wang, Yanyan Liang, Junchi Yan, Tao He, and Pai Peng. 2024. CalibRBEV: Multi-Camera Calibration via Reversed Bird's-eye-view Representations for Autonomous Driving. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 9145–9154.

[21] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*. 74–81.

[22] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems* 36 (2023), 34892–34916.

[23] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[24] Ana-Maria Marcu, Long Chen, Jan Hünermann, Alice Karnsund, Benoit Hanotte, Prajwal Chidananda, Saurabh Nair, Vijay Badrinarayanan, Alex Kendall, Jamie Shotton, et al. 2024. LingoQA: Visual question answering for autonomous driving. In *European Conference on Computer Vision*. Springer, 252–269.

[25] Torsten Merz and Farid Kendoul. 2011. Beyond visual range obstacle avoidance and infrastructure inspection by an autonomous helicopter. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 4953–4960.

[26] Ming Nie, Renyuan Peng, Chunwei Wang, Xinyue Cai, Jianhua Han, Hang Xu, and Li Zhang. 2024. Reason2drive: Towards interpretable and chain-based reasoning for autonomous driving. In *European Conference on Computer Vision*. Springer, 292–308.

[27] Chenbin Pan, Burhaneddin Yaman, Tommaso Nesti, Abhirup Mallik, Alessandro G Allievi, Senem Velipasalar, and Liu Ren. 2024. Vlp: Vision language planning for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14760–14769.

[28] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*. 311–318.

[29] Qucheng Peng, Benjamin Planche, Zhongpai Gao, Meng Zheng, Anwesa Choudhuri, Terrence Chen, Chen Chen, and Ziyan Wu. 2025. 3D Vision-Language Gaussian Splatting. In *The Thirteenth International Conference on Learning Representations*.

[30] Qucheng Peng, Ce Zheng, and Chen Chen. 2024. A dual-augmentor framework for domain generalization in 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2240–2249.

[31] Qucheng Peng, Ce Zheng, Zhengming Ding, Pu Wang, and Chen Chen. 2025. Exploiting Aggregation and Segregation of Representations for Domain Adaptive Human Pose Estimation. In *2025 IEEE 19th International Conference on Automatic Face and Gesture Recognition (FG)*. 1–10. doi:10.1109/FG61629.2025.11099339

[32] Tianwen Qian, Jingjing Chen, Linhai Zhuo, Yang Jiao, and Yu-Gang Jiang. 2024. Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 4542–4550.

[33] Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems* 36 (2023), 53728–53741.

[34] Fu Rong, Wenjin Peng, Meng Lan, Qian Zhang, and Lefei Zhang. 2024. Driving Scene Understanding with Traffic Scene-Assisted Topology Graph Transformer. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 10075–10084.

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

[36] Ziying Song, Caiyan Jia, Lin Liu, Hongyu Pan, Yongchang Zhang, Junming Wang, Xingyu Zhang, Shaoqing Xu, Lei Yang, and Yadan Luo. 2025. Don't Shake the Wheel: Momentum-Aware Planning in End-to-End Autonomous Driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

[37] Gilbert Strang and Kai Borre. 1997. *Linear algebra, geodesy, and GPS*. Siam.

[38] Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. 2024. Sparsedrive: End-to-end autonomous driving via sparse scene representation. *arXiv preprint arXiv:2405.19620* (2024).

[39] Gabriel Svennerberg. 2010. *Beginning google maps API 3*. Apress.

[40] Naftali Tishby and Noga Zaslavsky. 2015. Deep learning and the information bottleneck principle. In *2015 ieee information theory workshop (itw)*. Ieee, 1–5.

[41] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4566–4575.

[42] Peter West, Ari Holtzman, Jan Buys, and Yejin Choi. 2019. BottleSum: Unsupervised and Self-supervised Sentence Summarization using the Information Bottleneck Principle. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 3752–3761.

[43] Yi Xin, Junlong Du, Qiang Wang, Zhiwen Lin, and Ke Yan. 2024. VMT-Adapter: Parameter-Efficient Transfer Learning for Multi-Task Dense Scene Understanding.

In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 16085–16093.

[44] Yi Xin, Siqi Luo, Xuyang Liu, Haodi Zhou, Xinyu Cheng, Christina E Lee, Junlong Du, Haozhe Wang, MingCai Chen, Ting Liu, et al. 2024. V-petl bench: A unified visual parameter-efficient transfer learning benchmark. *Advances in Neural Information Processing Systems* 37 (2024), 80522–80535.

[45] Yi Xin, Siqi Luo, Haodi Zhou, Junlong Du, Xiaohong Liu, Yue Fan, Qing Li, and Yuntao Du. 2024. Parameter-efficient fine-tuning for pre-trained vision models: A survey. *arXiv preprint arXiv:2402.02242* (2024).

[46] Yi Xin, Juncheng Yan, Qi Qin, Zhen Li, Dongyang Liu, Shicheng Li, Victor Shea-Jay Huang, Yupeng Zhou, Renrui Zhang, Le Zhuo, et al. 2025. Lumina-mGPT 2.0: Stand-Alone AutoRegressive Image Modeling. *arXiv preprint arXiv:2507.17801* (2025).

[47] Yi Xin, Le Zhuo, Qi Qin, Siqi Luo, Yuewen Cao, Bin Fu, Yangfan He, Hongsheng Li, Guangtao Zhai, Xiaohong Liu, et al. 2025. Resurrect Mask AutoRegressive Modeling for Efficient and Scalable Image Generation. *arXiv preprint arXiv:2507.13032* (2025).

[48] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. 2024. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters* (2024).

[49] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115* (2024).

[50] Chen Yang, Yangfan He, Aaron Xuxiang Tian, Dong Chen, Jianhui Wang, Tianyu Shi, Arsalan Heydarian, and Pei Liu. 2024. Wcdt: World-centric diffusion transformer for traffic scene generation. *arXiv preprint arXiv:2404.02082* (2024).

[51] Renrui Zhang, Jiaming Han, Chris Liu, Aojun Zhou, Pan Lu, Yu Qiao, Hongsheng Li, and Peng Gao. 2024. LLaMA-Adapter: Efficient Fine-tuning of Large Language Models with Zero-initialized Attention. In *The Twelfth International Conference on Learning Representations*.

[52] Wenzhao Zheng, Ruiqi Song, Xianda Guo, Chenming Zhang, and Long Chen. 2024. Genad: Generative end-to-end autonomous driving. In *European Conference on Computer Vision*. Springer, 87–104.

[53] Yiyang Zhou, Yangfan He, Yaofeng Su, Siwei Han, Joel Jang, Gedas Bertasius, Mohit Bansal, and Huaxiu Yao. 2025. ReAgent-V: A Reward-Driven Multi-Agent Framework for Video Understanding. *arXiv preprint arXiv:2506.01300* (2025).

[54] Ziqi Zhou, Jingyue Zhang, Jingyuan Zhang, Yangfan He, Boyue Wang, Tianyu Shi, and Alaa Khamis. 2024. Human-centric Reward Optimization for Reinforcement Learning-based Automated Driving using Large Language Models. *arXiv preprint arXiv:2405.04135* (2024).