OmniSpace: Efficient Geometry Awareness for Autonomous Vehicles MLLMs

Hao Vo[1*], **Phu Loc Nguyen**[1*], **Khoa Vo**[1], **Sieu Tran**[1], **Duc Minh Nguyen**[1], **Ngo Xuan Cuong**[1], **Nghi D. Q. Bui**[2], **Anh Nguyen**[3], **Duy Minh Ho Nguyen**[4,5,6], **Ngan Le**[1]

[1]University of Arkansas [2]Center for AI Research, VinUniversity
[3]University of Liverpool [4]University of Stuttgart [5]DFKI
[6]Max Planck Research School for Intelligent Systems

[*]Equal contribution
Project Page

Abstract

Multimodal Large Language Models (MLLMs) have achieved remarkable performance on 2D visual tasks, yet enhancing their spatial intelligence for real-world applications such as Autonomous Vehicles (AV) remains an open challenge. Existing geometry-aware MLLMs typically rely on auxiliary 3D models at inference time, introducing pipeline complexity and the risk of cascading failures. In this paper, we present **OmniSpace**, a backbone-agnostic framework for geometry-aware spatial reasoning from purely camera-only observations. Motivated by our finding that current MLLMs are bottlenecked by weak cross-view correspondence and depth estimation, **OmniSpace** introduces a Camera Pose Injector, a Multi-view Epipolar Attention module, and a 3D Geometric Distillation objective that jointly address these two limitations by transferring geometric knowledge into the model. Extensive experiments show that **OmniSpace** surpasses existing methods on planning benchmarks (nuScenes, Bench2Drive), risk detection (nuInstruct), language (Omnidrive), and generalization (DriveBench).

1 Introduction

Rapid advancements and impressive performance of Multimodal Large Language Models (MLLMs) [61, 1, 20, 30, 35, 36, 51, 64, 68, 57, 55] have positioned them as promising, generalizable alternatives to traditional Autonomous Vehicles (AV) systems [17, 28, 65, 56]. A broad spectrum of MLLM-based methods has emerged, ranging from multi-stage pipelines [31, 53] to end-to-end frameworks [76, 70], from static-frame methods [60, 10] to dynamic models [62, 19, 76], and from single-view approaches [71, 70, 21, 53] to multi-view paradigms [62, 19].

Despite these advances, many existing methods still formulate AV tasks as a conventional 2D visual autoregressive problem [19, 76, 27] (Figure 1a). While this formulation offers a convenient path to adapting general-purpose MLLMs, it does so at the cost of geometric transparency: depth, scale, camera pose, and cross-view correspondence remain buried within image tokens, rather than being surfaced through geometry-aware visual features. This implicit treatment of geometry is particularly problematic in AV settings, where faithful geometry understanding lies at the heart of every downstream decision – from collision avoidance to lane-level planning. A separate line of work addresses this limitation by introducing additional geometry-aware modules into MLLMs [60, 39, 62] (Figure 1b). Although effective, this paradigm relies on test-time external 3D geometric model, which increases computational cost, complicates deployment, and makes the final prediction dependent on the quality of intermediate geometric estimates. This raises a natural question: *Can MLLMs acquire geometry awareness directly, without sacrificing efficiency or deployment simplicity?*

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

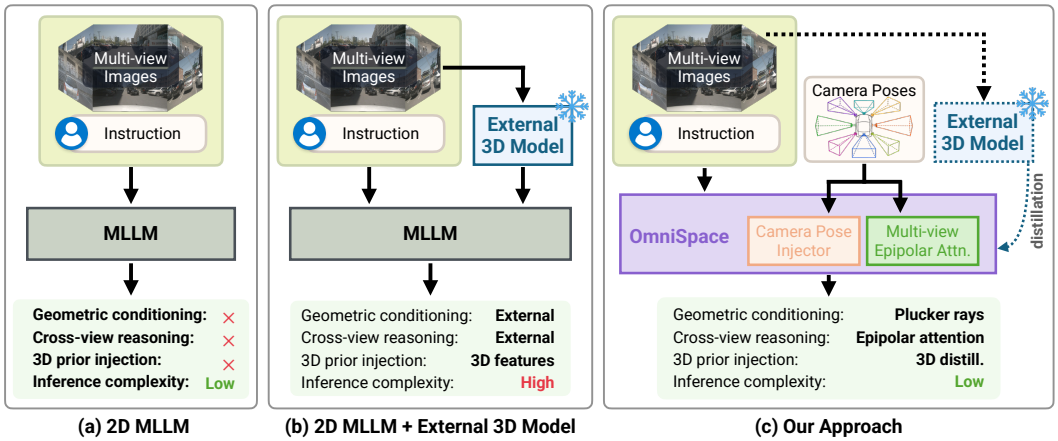


Figure 1: Comparison between existing paradigms and **OmniSpace**.

To answer this question, we conduct a diagnostic investigation of current MLLMs in AV scene understanding. Drawing on principles from multi-view geometry [14], recovering a complete 3D scene requires both estimating per-image depth and reasoning about the geometric relationships among views. We therefore isolate two capabilities most directly tied to 3D scene understanding: (i) *depth estimation*, the ability to infer metric scene structure from camera observations; and (ii) *cross-view correspondence*, the ability to associate the same scene elements across different camera views. In Section 2, we construct a benchmark to evaluate these two capabilities in four representative MLLMs [1, 64, 75, 52] and examine whether stronger performance on these geometric diagnostics translates into stronger performance on AV-oriented tasks [62, 4, 17]. The investigation reveals a consistent trend: *models with stronger depth estimation and cross-view correspondence also achieve better downstream AV tasks*. This finding suggests that these two geometric capabilities are key bottlenecks for MLLM-based AV understanding and motivates us to strengthen them directly within the MLLM, without adding external geometry modules at test time.

In this paper, we propose **OmniSpace** (Figure 1c), a new geometry-aware adaptation framework that strengthens cross-view correspondence and depth perception directly within MLLMs, without introducing additional test-time components. OmniSpace consists of two key designs: (i) To capture cross-view correspondence, we first introduce a *Camera Pose Injector* that encodes each visual feature with its corresponding Plücker ray representation, providing explicit view-aware cues that inform the model of each pixel's 3D origin and viewing direction. Then, we propose *Multi-view Epipolar Attention*, which uses epipolar geometry to restrict cross-view attention to geometrically plausible regions, thereby encouraging the model to associate corresponding scene elements across camera views. (ii) To equip the model with depth understanding, we introduce *3D Geometric Distillation*, using a 3D foundation model [59] as a training-time teacher to distill geometric representations into OmniSpace, improving 3D scene understanding without incurring additional inference cost. Through these designs, OmniSpace directly enhances the geometric representations of MLLMs for AV scene understanding while preserving inference efficiency and deployment simplicity.

Our key contributions are: (i) We identify depth estimation and cross-view correspondence as **two key geometric bottlenecks** for MLLM-based AV scene understanding, and validate their correlation with downstream 3D performance through a diagnostic investigation. (ii) We propose **OmniSpace**, a geometry-aware MLLM adaptation framework that integrates Camera Pose Injection, Multi-view Epipolar Attention, and 3D Geometric Distillation to strengthen spatial reasoning without relying on external 3D models at inference. (iii) Through **extensive experiments** on AV-oriented 3D scene understanding benchmarks, OmniSpace improves planning accuracy, spatial reasoning, and geometric understanding over strong MLLM baselines while maintaining efficient inference.

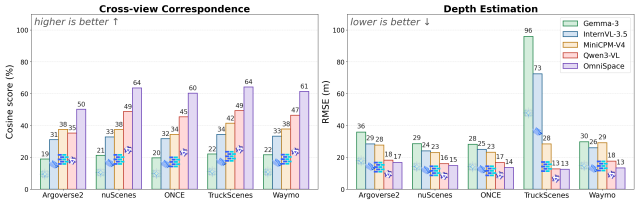
2 Diagnostic Analysis: Geometry Awareness in Autonomous Driving MLLMs

Why current MLLMs lack of geometry awareness. As aforementioned, we hypothesize that the main bottleneck of current MLLMs lies in two specific abilities: depth estimation and cross-view

correspondence. A likely cause is the pretraining paradigm itself. Most recent MLLMs inherit their visual representations from image–text pretraining under a CLIP-style paradigm [41]. Such supervision is highly effective for learning semantic alignment between images and language, but limited explicit supervision for construct a geometric structure, and spatial layout. Consequently, the visual encoder may recognize objects correctly yet still fail to localize them in 3D space or to determine how observations from different cameras correspond to the same physical scene. To validate this observation and quantify its effect on driving performance, we conduct two complementary experiments described below.

Experiment #1 - Figure 2 (left): We first conduct a diagnostic benchmark, called *Geo-Bench*, sampled from five AV datasets: nuScenes [4], Argoverse 2 [66], Waymo [48], ONCE [40], and TruckScenes [8]. Geo-Bench targets two complementary geometric abilities, *cross-view correspondence* and *monocular depth understanding*, evaluated through distinct protocols to isolate whether model failures stem from inaccurate depth perception, weak multi-view association, or both. The QA pairs are designed specifically for depth understanding: each of the 250 multi-view questions asks the model to estimate the distance to a referenced object, and we report root mean squared error (RMSE) between the predicted and ground-truth distances. Cross-view correspondence, in contrast, is not tested through QA but through a representation-level probe: given a patch feature from one camera, we use LiDAR-derived depth and camera calibration to project its 3D location into another view and compute the cosine similarity between the two corresponding patch features, where higher similarity indicates that the model represents the same scene element consistently across views. Figure 2 (left) reports the performance of four recent MLLMs [1, 52, 64, 75] on Geo-Bench under both protocols. Details of the Geo-Bench construction pipeline are provided in Appendix A.

Experiment #2 - Figure 2 (right): We evaluate the same MLLMs on three AV benchmarks: nuInstruct, nuScenes, and OmniDrive [4, 6, 62]. These benchmarks cover complementary aspects of driving intelligence, including risk perception, planning-related prediction, and driving-scene description. To adapt each MLLM to the target domain, we apply LoRA fine-tuning [15] following the prompting protocol of [19]. We report the official metrics for each benchmark, including BLEU, accuracy, MAP, MAE, L2 error, collision rate, intersection rate, ROUGE, and CIDEr.
We first observe that generic MLLM capability does not always translate into strong AV-domain performance. For example, although InternVL3.5 is a strong general-purpose MLLM, MiniCPM-V4 outperforms it on several AV metrics. More importantly, we observe that *geometry-aware models transfer better to driving tasks*. Models with stronger diagnostic geometry scores generally achieve better downstream AV performance. This supports our hypothesis that geometry awareness is not merely an auxiliary perception ability, but a core capability for autonomous-driving MLLMs. Motivated by this finding, we introduce a geometry-aware training strategy that strengthens monocular depth estimation and cross-view correspondence.



	Metric	Qwen3-VL	InternVL3.5	Gemma-3	MiniCPM-V4
nuInstruct	BLEU ↑	86.71	86.06	84.11	86.39
	Accuracy ↑	59.51	57.50	53.73	57.48
	MAP ↑	35.32	19.06	10.01	30.71
	MAE ↓	2.47	2.86	2.78	2.60
	Average ↑	44.81	39.94	36.29	43.00
nuScenes	L2 ↓	0.36	0.37	0.37	0.37
	Collision ↓	0.36	0.35	0.39	0.35
	Intersection ↓	3.05	3.59	3.59	3.41
	Average ↓	1.26	1.44	1.45	1.38
Omnidrive	BLEU ↑	25.07	24.86	24.70	24.93
	ROUGE ↑	37.22	37.30	36.97	37.26
	CIDEr ↑	94.93	94.01	93.41	94.89
	Average ↑	52.41	52.06	51.69	52.36

Figure 2: Correlation between cross-view correspondence, depth estimation, and downstream AV performance. **Left:** per-model scores on our diagnostic benchmark. **Right:** downstream results on nuInstruct, nuScenes, and OmniDrive after LoRA fine-tuning.

3 OmniSpace

Motivated by the analysis in Section 2, we proposed **OmniSpace**, a plug-in geometry-aware architecture to enhance 3D awareness for AV tasks across a broad range of MLLMs. OmniSpace is designed with three objectives: **(i)** inject spatial awareness from camera views into visual tokens to improve metric depth awareness, **(ii)** explicitly model encourage cross-view correspondence through

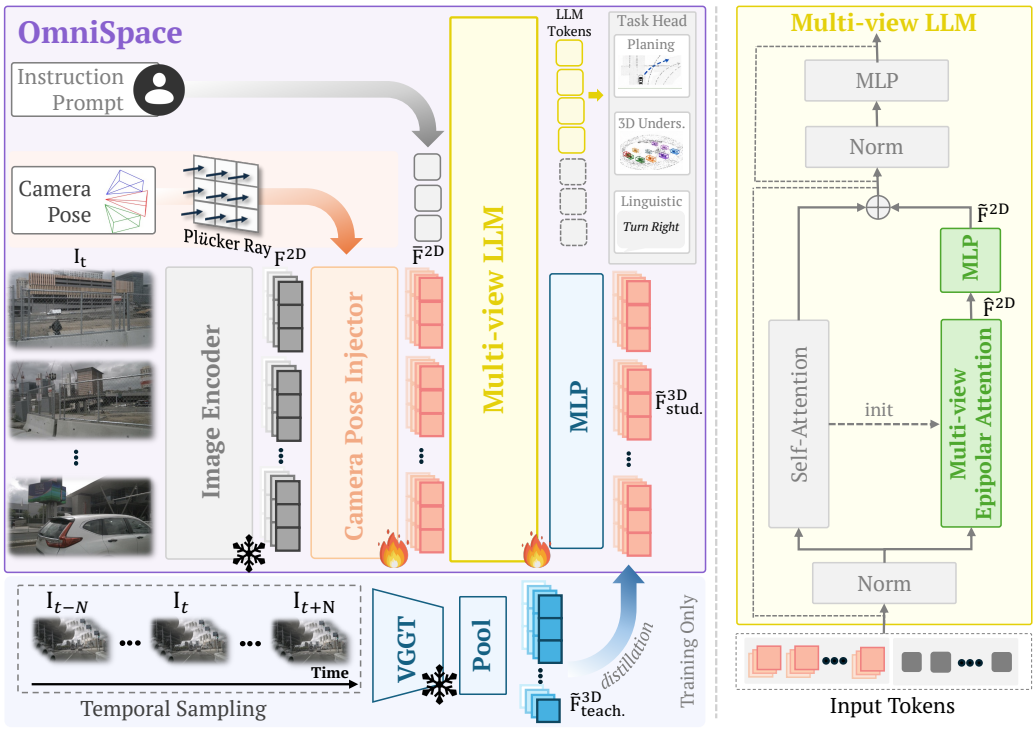


Figure 3: Overall architecture of the proposed **OmniSpace**. Given synchronized multi-view images, OmniSpace encodes 2D visual tokens with camera-pose-aware ray embeddings, refines them through epipolar-constrained cross-view attention, and distills 3D geometric priors during training. The 3D teacher is discarded at inference.

epipolar-constrained interaction and **(iii)** leverage 3D foundation models during training without relying on external geometry models at inference time.

Preliminaries Given multi-view (N) images $\mathcal{I}_t = \{\mathbf{I}_t^1, \ldots, \mathbf{I}_t^N\}$ at timestamp t, a standard MLLM first encodes each image independently using a pretrained 2D visual encoder $\mathcal{E}_{2D}$. This produces visual feature maps $\mathbf{F}_t^n = \mathcal{E}_{2D}(\mathbf{I}_t^n) \in \mathbb{R}^{h \times w \times D}, \quad n = 1, \ldots, N$. Generally, we call $\mathbf{F}^{2D}$.

3.1 Camera Pose Injector

We inject camera-aware spatial priors into each visual token. Each point on the feature map F_{ij}^{2D} corresponds to a camera ray. A ray is defined by a camera center $\mathbf{o} \in \mathbb{R}^3$ and a direction $\mathbf{d}$. As the naïve representation $(\mathbf{o}, \mathbf{d})$ is not ideal, because the same geometric ray can be written using different points along that ray: suppose we have one ray starts at camera center $\mathbf{o}$ and points in direction $\mathbf{d}$. We consider another point lies somewhere along the same ray $\mathbf{o}' = \mathbf{o} + t\mathbf{d}$, where t is any scalar. If we represent the ray naively as ($\mathbf{o}$,$\mathbf{d}$), then $(\mathbf{o}, \mathbf{d})$ and $(\mathbf{o} + t\mathbf{d}, \mathbf{d})$ look different, even though they describe the same geometric line. To avoid this problem, we are motivated by recent advances in Neural Light Fields [46], we adopt the Plücker ray embedding, $P_{ij} = (\mathbf{o} \times \mathbf{d}_{ij}, \mathbf{d}_{ij}) \in \mathbb{R}^6$, where $\mathbf{d}_{ij}$ is the normalized world-space ray direction for token (i, j). This parametrization elegantly maps $(\mathbf{o} \times \mathbf{d})$ and $(\mathbf{o} + t\mathbf{d}) \times \mathbf{d}$ to an identical embedding since any vector crossed with itself is zero:

$$(\mathbf{o} + t\mathbf{d}_{ij}) \times \mathbf{d}_{ij} = \mathbf{o} \times \mathbf{d}_{ij} + t\,(\mathbf{d}_{ij} \times \mathbf{d}_{ij}) = \mathbf{o} \times \mathbf{d}_{ij}. \tag{1}$$

The resulting Plücker ray map $\mathbf{P} \in \mathbb{R}^{(hw) \times 6}$ is passed through a lightweight projection ϕ_p, forming pose-aware feature map: $\bar{F}_{ij}^{2D} = F_{ij}^{2D} + \phi_p(P_{ij})$, where ϕ_p is implemented as a 1×1 convolution. To avoid disrupting the pretrained visual representation at the beginning of fine-tuning, ϕ_p is zero-initialized. Therefore, OmniSpace initially behaves identically to the base MLLM and gradually learns to inject camera geometry as training proceeds.

3.2 Multi-view Epipolar Attention

To encourage view-consistent representation learning, we propose to inject camera-aware geometry into cross-view attention so that a token in one view can only attend to geometrically plausible tokens in another view, rather than attending to every token everywhere. Let $\bar{\mathbf{F}}_i^{2D}$ be the pose-aware feature map from view i, s is one source token location in that feature map, we compute its epipolar lines $\{l_j\}_{j\neq i}$ on all other pose-aware feature maps $\bar{\mathbf{F}}_j^{2D}\}_{j\neq i}$. When computing the attention map across views, we ignore points that do not lie on these epipolar lines, so that the source point s only attends to features along its corresponding camera ray in other views, in addition to all points within its own view:

$$\hat{\mathbf{F}}_{i,s}^{2D} = \mathrm{SoftMax}\left(\frac{Q(\bar{\mathbf{F}}_{i,s}^{2D}) \cdot K([\bar{\mathbf{F}}_i^{2D} \mid \bar{\mathbf{F}}_{j,l_j}^{2D}])^\top}{\sqrt{d}}\right) \cdot V([\bar{\mathbf{F}}_i^{2D} \mid \bar{\mathbf{F}}_{j,l_j}^{2D}]). \tag{2}$$

where $\bar{\mathbf{F}}_{i,s}^{2D}$ is the visual token at position s in view i and $Q\bar{\mathbf{F}}_{i,s}^{2D}$ is the query vector of the source token. $[\mathbf{F}_i^{2D} \mid \mathbf{F}_{j,l_j}^{2D}]$ is set of tokens that the source token is allowed to attend to whereas $\mathbf{F}_i^{2D}$ contains all tokens from the same view and $\bar{\mathbf{F}}_{j,l_j}^{2D}$ is only the tokens in another view j that lie on the epipolar line l_j. The operator $|$ denotes concatenation along the token dimension. Q, K, and V are learnable linear projections that map visual tokens into query, key, and value embeddings, respectively. In this way, each token preserves unrestricted self-attention within its own view while aggregating cross-view evidence only from epipolar-consistent regions. In practice, we dilate each epipolar line with a 3×3 filter to include neighboring target tokens and improve robustness to feature-map discretization and calibration noise. Finally, we inject the epipolar-attended feature through a zero-initialized projection layer: $\tilde{\mathbf{F}}_{i,s}^{2D} = \phi_f \hat{\mathbf{F}}_{i,s}^{2D}$, where ϕ_f is initialized to zero and learned during fine-tuning. This residual design allows the model to gradually acquire geometry-aware cross-view reasoning while preserving the behavior of the original pretrained model at initialization.

3.3 3D Geometric Distillation

While camera pose injection and multi-view epipolar attention introduce explicit geometric structure into the visual-token pathway, they do not by themselves specify what a 3D-aware representation should encode. A straightforward solution is to use LiDAR or an external 3D foundation model at inference time, but this increases deployment cost and may introduce cascading errors when the external model fails. Instead, we leverage a feed-forward 3D foundation model [59], denoted by $\mathcal{E}_{3D}$, only during training and distill its geometric knowledge into OmniSpace.

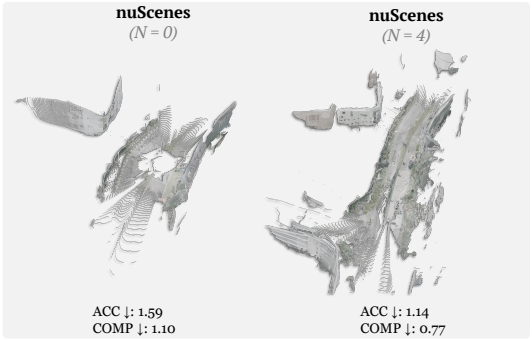


Figure 4: Qualitative and quantitative results of VGGT [59] on nuScenes [4] validation sets under current-frame and temporally aggregated inputs.

Naively following Spatial-MLLM [67] by feeding only synchronized multi-view images at timestamp t, $\mathcal{I}_t$, into $\mathcal{E}_{3D}$ is suboptimal for AV scenes. Existing feed-forward 3D models are typically trained on indoor, object-centric, or reconstruction-oriented datasets [22, 44, 82, 42], where views have substantial overlap and scenes are relatively static. In contrast, AV scenes contain limited inter-view overlap, large outdoor depth ranges, moving objects, and frequent occlusions. As shown in Figure 4, directly applying $\mathcal{E}_{3D}$ to current-frame AV inputs can produce noisy and incomplete geometric supervision.

To ensure reliability, we introduce *temporal geometric aggregation*. At timestep t, we sample a symmetric temporal window of $2K+1$ synchronized multi-view frames: $\mathcal{C}_t = \{\mathcal{I}_{t-K}, \ldots, \mathcal{I}_t, \ldots, \mathcal{I}_{t+K}\}$. We pass the full clip into the 3D teacher $\mathcal{E}_{3D}$ and obtain $\mathcal{E}_{3D}(\mathcal{C}_t) = \{\mathbf{F}_{t-K}^{3D}, \ldots, \mathbf{F}_t^{3D}, \ldots, \mathbf{F}_{t+K}^{3D}\}$, where $\mathbf{F}_\tau^{3D} \in \mathbb{R}^{HW\times\hat{D}}$ denotes the teacher feature map at timestamp τ. We aggregate the teacher features along the temporal dimension to form a stable teacher:

$$\tilde{\mathbf{F}}_{teacher}^{3D} = \frac{1}{2K+1}\sum_{\tau=t-K}^{t+K} \mathbf{F}_\tau^{3D} \in \mathbb{R}^{HW\times\hat{D}}. \tag{3}$$

Table 1: **Open-loop planning results on nuScenes**. The best is **bold** and the runner up is underline.

	Method	LLM	Params	FPS ↑	L2 ↓				Collision (%) ↓				Intersection (%) ↓			
					1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Traditional	UniAD [17]	–	125M	1.8	0.20	0.42	0.75	0.46	0.02	0.25	0.84	0.37	**0.20**	1.33	3.24	1.59
	VAD-Base [28]	–	58M	4.5	0.17	0.34	0.60	0.37	0.04	0.27	0.67	0.33	0.21	2.13	5.06	2.47
	BEV-Planner [34]	–	–	–	0.16	0.32	0.57	0.35	**0.00**	0.29	0.73	0.34	0.35	2.62	6.51	3.16
	ST-P3 [16]	–	11M	–	1.59	2.64	3.73	2.65	0.69	3.62	8.39	4.23	2.53	8.17	14.40	8.37
MLLMs-based	*2D MLLM*															
	Baseline	Qwen2.5-VL 7B	7.75B	0.42	0.15	0.33	0.62	0.37	**0.00**	0.18	0.76	0.31	0.94	3.09	6.76	3.59
	Baseline	InternVL3.5 4B	4.73B	0.47	0.15	0.33	0.62	0.37	**0.00**	0.18	0.88	0.35	0.78	3.03	6.95	3.59
	Baseline	Qwen3-VL 4B	4.4B	0.44	0.15	0.32	0.61	0.36	**0.00**	0.14	0.96	0.36	0.57	2.54	6.04	3.05
	EMMA [21]	–	–	–	0.14	0.29	0.54	0.32	-	-	-	-	-	-	-	-
	DriveVLM [54]	Qwen-VL	9.6B	–	0.18	0.34	0.68	0.40	0.10	0.22	0.45	0.27	-	-	-	-
	2D MLLM + test-time 3D model															
	ORION [9]	Vicuna-7B-v1.5	7.55B	–	0.17	0.31	0.55	0.34	0.05	0.25	0.80	0.37	-	-	-	-
	OmniDrive [62]	LLaVA-7B	7.2B	0.27	0.14	0.29	0.55	0.33	**0.00**	0.13	0.78	0.30	0.56	2.48	5.96	3.00
	VGGDrive [60]	Qwen2.5-VL 7B	8.5B	0.03	0.14	0.28	0.51	0.31	0.02	**0.10**	0.55	0.22	0.63	2.27	4.02	2.31
	SpaceDrive [31]	Qwen2.5-VL 7B	8.7B	0.25	0.15	0.29	0.51	0.32	0.04	0.18	0.49	0.23	0.22	**0.80**	2.79	1.27
	2D MLLM + train-time 3D model															
	OmniSpace	Qwen2.5-VL 7B	7.79B	0.40	**0.12**	**0.25**	0.47	**0.28**	**0.00**	**0.10**	0.57	0.22	0.32	1.22	**2.22**	**1.25**
	OmniSpace	InternVL3.5 4B	4.77B	0.46	0.13	0.26	0.50	0.30	0.01	**0.10**	0.50	0.20	0.31	1.88	2.32	1.50
	OmniSpace	Qwen3-VL 4B	4.44B	0.42	**0.12**	0.26	**0.45**	**0.28**	0.02	0.14	**0.42**	**0.19**	0.25	1.20	2.36	1.27

This temporal aggregation increases effective scene overlap across frames and reduces single-frame reconstruction artifacts caused by sparse camera overlap or transient occlusions.

Given the instruction prompt T and geometry-enhanced visual tokens $\tilde{\mathbf{F}}^{2D}$ from Section 3.2, the MLLM backbone produces visual representations $\bar{\tilde{\mathbf{F}}}^{2D}$. We project these student representations into the teacher feature dimension using a lightweight MLP: $\tilde{\mathbf{F}}^{3D}_{stud.} = \phi_d(\bar{\tilde{\mathbf{F}}}^{2D}) \in \mathbb{R}^{hw \times \hat{D}}$.

Since the teacher and student feature maps may have different spatial resolutions, we apply adaptive pooling to align the teacher feature map to the student resolution. The distillation objective is:

$$\mathcal{L}_{3D} = \left\| \tilde{\mathbf{F}}^{3D}_{stud.} - \mathrm{Pool}(\tilde{\mathbf{F}}^{3D}_{teach.}) \right\|_2^2. \tag{4}$$

This objective encourages the MLLM visual tokens to encode 3D-aware structure learned from the teacher $\mathcal{E}_{3D}$, while preserving the original language-modeling objective for downstream instruction driving reasoning. Because $\mathcal{E}_{3D}$ is used only to generate training targets and is discarded at inference, OmniSpace inherits 3D geometric priors without test-time dependence on external 3D models.

The overall training objective combines the standard downstream cross-entropy loss with the 3D geometric distillation loss defined in Eq. 4.

4 Experiments

4.1 Experiment Setup

Datasets and Metrics. We evaluate OmniSpace on five autonomous-driving benchmarks spanning trajectory planning (nuScenes [4] and Bench2Drive [25]), risk detection (NuInstruct [6]), language understanding (OmniDrive [62]), and generalization (DriveBench [69]). For fair comparison, each model is trained and evaluated following the official protocol and metrics of each benchmark.

Baselines and Implementation Details. We evaluate OmniSpace with several MLLM backbones, including Qwen3-VL-4B [1], Qwen2.5-VL-7B [2], and InternVL3.5-4B [64], using Qwen3-VL-4B as the default backbone. All backbones are fine-tuned with LoRA [15] using rank 16, while the original vision encoder and vision-language projector frozen. We use VGGT [59] as the 3D teacher $\mathcal{E}_{3D}$ and set the temporal sampling window parameter to $N = 4$. Models are trained for 2 epochs with batch size 8, learning rate 1×10^{-4}, and cosine annealing. For fair comparison, we follow [60, 62, 31] and incorporate ego-vehicle states into the instruction prompt for planning tasks. All experiments are conducted on 8 NVIDIA A100 GPUs 40GB.

4.2 State-of-the-art Comparison

Evaluation on Open-loop Planning on nuScenes. Table 1 shows that OmniSpace consistently improves over its corresponding fine-tuned baselines across Qwen2.5-VL, InternVL3.5, and Qwen3-VL backbones, indicating that the gains come from the proposed spatial modeling rather than a specific

Table 2: **Closed-loop planning results on Bench2Drive.** The best is **bold** and the runner up is underline. Methods marked with † use additional training data beyond the standard benchmark.

Method	Params	DS ↑	SR(%) ↑
Traditional			
AD-MLP [77]	285K	18.05	0.00
UniAD-Base [17]	125M	45.81	16.36
VAD-Base [28]	58M	42.35	15.00
MomAD [47]	87M	44.54	16.71
GenAD [81]	113M	44.81	15.90
SparseDrive [49]	86M	47.38	17.72
UAD [11]	–	49.22	20.45
WoTE [32]	64M	61.71	31.36
ThinkTwice [24]	129M	62.44	37.17
DriveTransformer-L [26]	520M	63.46	38.60
DriveAdapter [23]	161M	64.22	42.08
HiP-AD [50]	98M	86.77	69.09

Method	LLM	Params	DS ↑	SR(%) ↑
2D MLLM				
ReAL-AD [38]	MiniCPMLlama3-2.5V	–	41.17	11.36
Dual-AEB [79]	Qwen-0.5B	–	45.23	10.00
X-Driver [37]	LLaVA	–	51.70	18.10
GEMINUS [58]	–	–	65.39	37.73
VDRive [12]	InternVL3-8B	–	66.25	50.51
StuckSolver [3]	GPT-4o	–	70.89	50.01
DriveMoE [74]	Paligemma-3B	–	74.22	48.64
ETA [13]	LLaVA 1.6 7B Vicuna	–	74.33	48.33
VLR-Drive [29]	LLaVA-NeXT Video 7B	–	75.01	50.00
SimLingo [43]†	InternVL2-1B	1.36B	85.07	67.27
2D MLLM + test-time 3D model				
ORION [9]	Vicuna-7B-v1.5	7.55B	77.74	54.62
SpaceDrive [31]	Qwen2.5-VL 7B	8.7B	78.02	55.11
2D MLLM + train-time 3D model (Ours)				
OmniSpace	Qwen2.5-VL 7B	7.79B	79.65	60.15
OmniSpace	Qwen3-VL 4B	4.44B	**81.40**	**65.52**

backbone. Compared with MLLM-based methods that rely on test-time 3D modules, OmniSpace achieves competitive or better planning accuracy while maintaining substantially higher throughput. These results suggest that training-time geometric supervision and explicit camera-pose modeling provide an effective accuracy–efficiency trade-off for open-loop planning.

Evaluation on Closed-loop Planning on Bench2Drive. In addition to the open-loop setting, we further conduct closed-loop evaluation to establish a more comprehensive and reliable assessment of planning performance. As shown in Table 2, by introducing explicit geometric constraints, OmniSpace achieves a Driving Score of 79.65 and a Success Rate of 60.15% with the Qwen2.5-VL backbone, surpassing SpaceDrive [31] which relies on an external depth estimation model. A consistent trend is observed with the Qwen3-VL 4B variant, which ranks as the best-performing MLLM-based method on the benchmark, despite using a substantially smaller backbone than most competing approaches. Notably, although SimLingo [43] reports a higher score, it benefits from extensive data augmentation through Action Dreaming, whereas OmniSpace attains comparable performance under the standard benchmark setting. These results collectively demonstrate the generalizability and effectiveness of our spatial-aware design across different backbones and evaluation regimes.

Evaluation on Risk Detection on NuInstruct. To evaluate the performance of OmniSpace on risk scenario perception, we conduct experiments on the NuInstruct benchmark. As shown in Table 3 (Left), most MLLM-based models struggle with the risk detection task (MAP), revealing a fundamental limitation in their spatial reasoning capabilities. While InMLLM and VGGDrive improve performance by incorporating BEVFormer [33] and VGGT [59] into their inference pipelines, this introduces substantial computational overhead at test time. In contrast, our approach achieves superior performance across all metrics—including the lowest MAE (2.01), highest Accuracy (64.81), highest MAP (48.45), and highest BLEU (90.66)—without relying on any auxiliary 3D models at inference. These results demonstrate that distilling geometric priors during training enables OmniSpace to develop strong risk perception capabilities while preserving deployment efficiency.

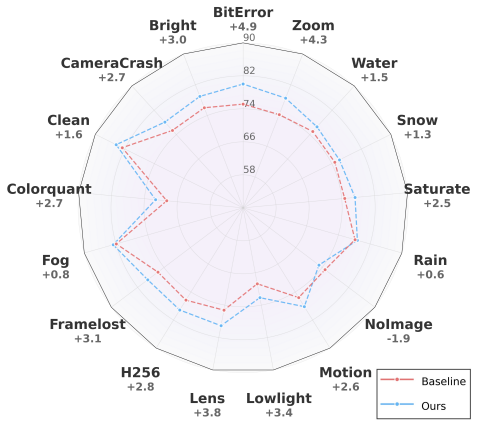


Figure 5: Generalization comparison on DriveBench [69], evaluated using GPT-Score.

Evaluation on Language Caption on OmniDrive. To verify that geometric constraints do not degrade the linguistic capability of the base MLLM we evaluate OmniSpace on the OmniDrive benchmark, which focuses on caption-related tasks where base VLMs typically excel. As shown in Table 3 (Right), OmniSpace achieves the best overall performance among fine-tuned methods, with notable gains on CIDEr (96.70) and ROUGE (40.44), and a competitive BLEU score (37.88), yielding the highest average of 58.34. Compared to OmniDrive [62], HERMES [83], and VGGDrive [60], our method consistently delivers superior captioning quality, indicating that geometric distillation not only enhances spatial reasoning but also enriches the model's ability to describe complex driving

Table 3: **Left: Risk detection results on NuInstruct**. Avg.$^* = \max\left(\frac{\text{Accuracy+MAP+BLEU}-\text{MAE}}{4}, 0\right)$. **Right: Language caption on OmniDrive**. ZS and FT denote Zero-shot and Fine-tuned, respectively. ‡ shows results take from [60].

	Model	MAE↓	Acc.↑	MAP↑	BLEU↑	Avg.* ↑
ZS	GPT-4o [20]‡	9.93	10.64	0	7.08	1.95
	LLAVA-OV [30]‡	87.04	3.75	0	8.55	0
	RoboTron [19]‡	19.36	2.57	0	8.06	0
	Qwen2.5-VL‡	24.10	0.63	0	5.56	0
FT	Baseline	2.47	59.51	35.32	86.71	44.81
	InMLLM [6]‡	9.08	32.48	21.93	35.20	20.13
	VGGDrive [60]‡	3.08	56.37	37.49	81.13	42.98
	OmniSpace	**2.01**	**64.81**	**40.36**	**90.66**	**48.45**

	Model	BLEU↑	CIDEr↑	ROUGE↑	Avg.↑
ZS	GPT-4o [20]‡	10.91	24.42	22.34	19.22
	LLAVA-OV [30]‡	16.14	28.41	22.14	22.23
	RoboTron [19]‡	20.30	34.33	23.67	26.10
FT	Baseline	25.07	94.93	37.22	52.41
	OmniDrive [62]‡	38.00	68.60	32.60	46.40
	HERMES [83]‡	–	74.10	32.70	–
	VGGDrive [60]‡	37.58	86.57	34.40	52.85
	OmniSpace	**37.88**	**96.70**	**40.44**	**58.34**

scenes. This confirms that the spatial-aware design of OmniSpace complements rather than conflicts with the language generation capability inherited from the base MLLM.

Evaluation on Generalization. To further assess the robustness of our approach, we evaluate the generalization ability of OmniSpace under challenging scenarios by finetuning the baseline model (Qwen3VL-4B) on the DriveLM dataset [45] and testing on DriveBench [69] (Figure 5). The results show that distilling knowledge from a powerful teacher not only improves the baseline's spatial intelligence but also enhances its generalization, with OmniSpace consistently achieving higher performance than the baseline across all scenarios. See Appendix D and the supplementary material for additional qualitative results.

4.3 Ablation Study

Table 4: Effectiveness of each module on nuScenes.

Exp.	Camera Pose Injector	Multi-view EpipolarAttention	3D Geometry Distillation	L2 ↓	Collision (%) ↓	Intersection (%) ↓
#1	✗	✗	✗	0.36	0.36	3.05
#2	✓	✗	✗	0.32	0.24	2.80
#3	✗	✗	✓	0.30	0.23	2.92
#4	✗	✓	✗	0.33	0.23	2.84
#5	✓	✗	✓	0.29	0.21	2.52
#6	✓	✓	✓	**0.28**	**0.19**	**1.27**

Table 5: Temporal Sampling (N) on nuScenes. Time of feature extraction.

N	L2 ↓	Collision (%) ↓	Intersection (%) ↓	Time (h) ↓
0	0.32	0.26	2.18	10
2	0.30	0.22	1.67	27
4	**0.28**	**0.19**	**1.27**	**82**
6	0.28	0.20	1.23	110

Component-wise Analysis. Table 4 summarizes the contribution of each proposed module. Compared with the vanilla fine-tuned baseline (Exp.#1), each component improves trajectory error and/or safety-related metrics. The Camera Pose Injector (Exp.#2) and 3D Geometry Distillation (Exp.#3) provide strong individual gains, while Epipolar Feature Alignment (Exp.#4) also improves all reported metrics by enforcing geometry-aware cross-view interaction. Combining pose conditioning with geometric distillation (Exp.#5) further improves performance, and adding Epipolar Feature Alignment yields the best overall results. Together (Exp.#6), these trends suggest that the three components address complementary aspects of spatial reasoning, including camera-pose grounding, 3D geometric supervision, and cross-view consistency.

Table 6: Effectiveness of different ray representation on nuScenes open-loop planning.

Ray	L2 ↓	Collision (%) ↓	Intersection (%) ↓
-	0.29	0.22	2.66
Naïve Ray	0.29	0.24	2.50
Plücker Ray	**0.28**	**0.19**	**1.27**

Table 7: Effectiveness of different dual attention mechanism on nuScenes open-loop planning.

Attention	L2 ↓	Collision (%) ↓	Intersection (%) ↓
-	0.29	0.21	2.52
Vanilla Attention	0.29	0.24	1.96
Epipolar Attention	**0.28**	**0.19**	**1.27**

Different Temporal Sampling Configuration. Table 5 studies the temporal sampling window N used for 3D geometric distillation. The time column reports offline teacher feature-extraction time measured on a single NVIDIA A100 40GB GPU. Increasing N improves trajectory error and safety-related metrics over the single-frame setting, indicating that additional frames help the teacher recover more reliable 3D structure through multi-view consistency while increasing feature-extraction time. The gains saturate at $N = 4$, while increasing to $N = 6$ brings only marginal improvement at a higher distillation cost. We therefore adopt $N = 4$ as the default setting, which provides sufficient temporal context for geometric supervision while avoiding unnecessary overhead.

Different Pose Embedding Analysis. Table 6 compares different positional embedding schemes within the Camera Pose Injector. Naïve Ray encoding, which uses only ray directions, provides limited improvement over the variant without explicit ray encoding, suggesting that direction alone is insufficient to disambiguate spatial relations across views. In contrast, Plücker Rays jointly encode ray direction and origin and achieve the best results across all metrics, with a particularly clear improvement in Intersection rate. This indicates that explicitly incorporating both viewing orientation and spatial offset provides a richer geometric prior for cross-view correspondence and scene-layout-aware trajectory prediction.

Different Attention Mechanism Analysis. Table 7 compares attention designs for cross-view interaction. Vanilla Attention improves Intersection rate but does not consistently improve all metrics, suggesting that unconstrained global attention may introduce noisy cross-view correspondences. In contrast, Epipolar Attention restricts feature interaction to geometrically valid epipolar regions and achieves the best results across all metrics. This confirms that geometry-constrained attention provides more reliable cross-view alignment for spatially consistent trajectory prediction.

5 Related Works

Geometry-Aware in MLLMs A growing body of work observes that MLLMs trained under image–text supervision lack the *Spatial Intelligence* required to understand 3D scenes [72, 73, 78], as they struggle to construct internal representations of scene geometry. To address this, recent methods adopt straightforward recipes built on 3D foundation models [63, 59]: Spatial-MLLM [67] and VG-LLM [80] simply fuse 2D features with 3D features at the input stage, while 3DRS [18] and 3DThinker [5] merely apply a feature-distillation loss at the MLLM's last layer. This lightweight paradigm has been validated almost exclusively on indoor, static, or object-centric scenes, where short baselines and high inter-view overlap let off-the-shelf 3D teachers produce clean supervision that a single alignment loss can transfer. Driving scenes invert these properties: surround views with limited overlap, large depth ranges, and pervasive motion cause the same teachers to yield noisy, incomplete signals (Figure 4). Existing realizations thus do not transfer to driving without rethinking how supervision is generated and consumed.

MLLMs for Autonomous Vehicles Recent advances in MLLMs [1, 64, 52, 75] have driven a wave of driving systems, spanning language-conditioned models such as DriveGPT4 [71], DriveLM [45], and DriveVLM [54], as well as unified frameworks including EMMA [21], ORION [9], OmniDrive [62], and RoboTron-Drive [19]. As our diagnostic analysis in Section 2 shows, these models treat AV tasks as a 2D autoregressive problem and consequently struggle with depth and cross-view correspondence. The closest works to ours mitigate this by coupling the VLM with an external 3D model at inference: VGGDrive [60] routes VGGT [59] features through hierarchical cross-attention, and SpaceDrive [31] lifts visual patches into a common geometric space via a frozen depth estimator. Both depend on auxiliary 3D modules at *inference time*, which as reported in Table 1, drops throughput to 0.03 and 0.25 FPS respectively and exposes predictions to cascading failures of the upstream estimator. OmniSpace departs from this paradigm by confining 3D supervision to training, internalizing geometric awareness within the MLLM itself and avoiding any test-time 3D dependency.

6 Conclusion & Discussion

We presented **OmniSpace**, a geometry-aware adaptation framework that strengthens depth estimation and cross-view correspondence directly within MLLMs, without requiring an external 3D model at inference. Through Camera Pose Injection, Multi-view Epipolar Attention, and 3D Geometric Distillation with temporal aggregation, OmniSpace internalizes geometric priors at training time and consistently improves planning, risk detection, language, and generalization performance across five autonomous-driving benchmarks while preserving the inference efficiency of pure 2D MLLMs.

Discussion. Several aspects of OmniSpace leave room for further refinement. First, our temporal aggregation averages VGGT features over a symmetric window and could be made more precise by accounting for ego-motion when fusing neighboring frames. Second, the Plücker-ray and epipolar modules assume reasonably accurate camera calibration, which holds in standard AV datasets but may need additional robustness handling under sensor drift or recalibration. Finally, we adopt a single fixed teacher (VGGT); exploring stronger or AV-pretrained 3D foundation models is a straightforward direction we leave for future work.

7 Acknowledgement

This material is based upon work supported by the National Science Foundation under Award No. #2443877 (NSF CAREER), #2223793 (NSF EFRI BRAID), #2445877 (NSF E-RISE), #R01CA277739-04 (NIH). We also acknowledge the Arkansas High-Performance Computing Center for providing GPUs. Duy Minh Ho Nguyen contributed to this research under support from the Lower Saxony Ministry of Science and Culture (MWK) in the zukunft.niedersachsen program, the Endowed Chair of AAI at the University of Oldenburg, and the Federal Ministry of Research, Technology and Space (BMFTR) under grant numbers 16IW24006 (NoIDLEChatGPT) and 16IW26002 (AMENABLE).

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[2] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[3] Zhipeng Bao and Qianwen Li. Large language model-assisted autonomous vehicle recovery from immobilization. *arXiv preprint arXiv:2510.26023*, 2025.

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

[5] Zhangquan Chen, Manyuan Zhang, Xinlei Yu, Xufang Luo, Mingze Sun, Zihao Pan, Yan Feng, Peng Pei, Xunliang Cai, and Ruqi Huang. Think with 3d: Geometric imagination grounded spatial reasoning from limited views. *arXiv preprint arXiv:2510.18632*, 2025.

[6] Xinpeng Ding, Jianhua Han, Hang Xu, Xiaodan Liang, Wei Zhang, and Xiaomeng Li. 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*, pages 13668–13677, 2024.

[7] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, 2017.

[8] Felix Fent, Fabian Kuttenreich, Florian Ruch, Farija Rizwin, Stefan Juergens, Lorenz Lechermann, Christian Nissler, Andrea Perl, Ulrich Voll, Min Yan, et al. Man truckscenes: A multimodal dataset for autonomous trucking in diverse conditions. *Advances in Neural Information Processing Systems*, 37:62062–62082, 2024.

[9] Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. *arXiv preprint arXiv:2503.19755*, 2025.

[10] Mohsen Gholami, Ahmad Rezaei, Zhou Weimin, Sitong Mao, Shunbo Zhou, Yong Zhang, and Mohammad Akbari. Spatial reasoning with vision-language models in ego-centric multi-view scenes. *arXiv preprint arXiv:2509.06266*, 2025.

[11] Mingzhe Guo, Zhipeng Zhang, Yuan He, Ke Wang, Liping Jing, and Haibin Ling. End-to-end autonomous driving without costly modularization and 3d manual annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[12] Ziang Guo and Zufeng Zhang. Vdrive: Leveraging reinforced vla and diffusion policy for end-to-end autonomous driving. *arXiv preprint arXiv:2510.15446*, 2025.

[13] Shadi Hamdan, Chonghao Sima, Zetong Yang, Hongyang Li, and Fatma Guney. Eta: Efficiency through thinking ahead, a dual approach to self-driving with large models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[14] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[15] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 2022.

[16] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, 2022.

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

[18] Xiaohu Huang, Jingjing Wu, Qunyi Xie, and Kai Han. 3drs: Mllms need 3d-aware representation supervision for scene understanding. *arXiv preprint arXiv:2506.01946*, 2025.

[19] Zhijian Huang, Chengjian Feng, Feng Yan, Baihui Xiao, Zequn Jie, Yujie Zhong, Xiaodan Liang, and Lin Ma. Robotron-drive: All-in-one large multimodal model for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8011–8021, 2025.

[20] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[21] Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, et al. Emma: End-to-end multimodal model for autonomous driving. *arXiv preprint arXiv:2410.23262*, 2024.

[22] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014.

[23] Xiaosong Jia, Yulu Gao, Li Chen, Junchi Yan, Patrick Langechuan Liu, and Hongyang Li. Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

[24] Xiaosong Jia, Penghao Wu, Li Chen, Jiangwei Xie, Conghui He, Junchi Yan, and Hongyang Li. Think twice before driving: Towards scalable decoders for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[25] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. *Advances in Neural Information Processing Systems*, 2024.

[26] Xiaosong Jia, Junqi You, Zhiyuan Zhang, and Junchi Yan. Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. In *International Conference on Learning Representations (ICLR)*, 2025.

[27] Bo Jiang, Shaoyu Chen, Bencheng Liao, Xingyu Zhang, Wei Yin, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Senna: Bridging large vision-language models and end-to-end autonomous driving. *arXiv preprint arXiv:2410.22313*, 2024.

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

[29] Fanjie Kong, Yitong Li, Weihuang Chen, Chen Min, Yizhe Li, Zhiqiang Gao, Haoyang Li, Zhongyu Guo, and Hongbin Sun. Vlr-driver: Large vision-language-reasoning models for embodied autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[30] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[31] Peizheng Li, Zhenghao Zhang, David Holtz, Hang Yu, Yutong Yang, Yuzhi Lai, Rui Song, Andreas Geiger, and Andreas Zell. Spacedrive: Infusing spatial awareness into vlm-based autonomous driving. *arXiv preprint arXiv:2512.10719*, 2, 2025.

[32] Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via bev world model. *arXiv preprint arXiv:2504.01941*, 2025.

[33] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[34] Zhiqi Li, Zhiding Yu, Shiyi Lan, Jiahan Li, Jan Kautz, Tong Lu, and Jose M Alvarez. Is ego status all you need for open-loop end-to-end autonomous driving? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14864–14873, 2024.

[35] Ji Lin, Hongxu Yin, Wei Ping, Pavlo Molchanov, Mohammad Shoeybi, and Song Han. Vila: On pretraining for visual language models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26689–26699, 2024.

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

[37] Wei Liu, Jiyuan Zhang, Binxiong Zheng, Yufeng Hu, Yingzhan Lin, and Zengfeng Zeng. X-driver: Explainable autonomous driving with vision-language models. *arXiv preprint arXiv:2505.05098*, 2025.

[38] Yuhang Lu, Jiadong Tu, Yuexin Ma, and Xinge Zhu. Real-ad: Towards human-like reasoning in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[39] Yuechen Luo, Fang Li, Shaoqing Xu, Yang Ji, Zehan Zhang, Bing Wang, Yuannan Shen, Jianwei Cui, Long Chen, Guang Chen, et al. Last-vla: Thinking in latent spatio-temporal space for vision-language-action in autonomous driving. *arXiv preprint arXiv:2603.01928*, 2026.

[40] Jiageng Mao, Minzhe Niu, Chenhan Jiang, Hanxue Liang, Jingheng Chen, Xiaodan Liang, Yamin Li, Chaoqiang Ye, Wei Zhang, Zhenguo Li, et al. One million scenes for autonomous driving: Once dataset. *arXiv preprint arXiv:2106.11037*, 2021.

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, 2021.

[42] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10901–10911, 2021.

[43] Katrin Renz, Long Chen, Elahe Arani, and Oleg Sinavski. Simlingo: Vision-only closed-loop autonomous driving with language-action alignment. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

[44] Thomas Schops, Johannes L Schonberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3260–3269, 2017.

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

[46] Vincent Sitzmann, Semon Rezchikov, Bill Freeman, Josh Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. *Advances in Neural Information Processing Systems*, 34:19313–19325, 2021.

[47] Ziying Song, Caiyan Jia, Lin Liu, Hongyu Pan, Yongchang Zhang, Junming Wang, Xingyu Zhang, Shaoqing Xu, Lei Yang, and Yadan Luo. Don't shake the wheel: Momentum-aware planning in end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025.

[48] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2446–2454, 2020.

[49] Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[50] Yingqi Tang, Zhuoran Xu, Zhaotie Meng, and Erkang Cheng. Hip-ad: Hierarchical and multi-granularity planning with deformable attention for autonomous driving in a single decoder. *arXiv preprint arXiv:2503.08612*, 2025.

[51] Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

[52] Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*, 2024.

[53] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*, 2024.

[54] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, XianPeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. In *Conference on Robot Learning*, 2025.

[55] Thanh-Dat Truong, Huu-Thien Tran, Tran Thai Son, Bhiksha Raj, and Khoa Luu. Directed-tokens: A robust multi-modality alignment approach to large language-vision models. *arXiv preprint arXiv:2508.14264*, 2025.

[56] Hao Vo, Khoa Vo, Thinh Phan, Ngo Xuan Cuong, Gianfranco Doretto, Hien Nguyen, Anh Nguyen, and Ngan Le. Semlt3d: Semantic-guided expert distillation for camera-only long-tailed 3d object detection. *arXiv preprint arXiv:2604.18476*, 2026.

[57] Khoa Vo, Thinh Phan, Kashu Yamazaki, Minh Tran, and Ngan Le. Henasy: Learning to assemble scene-entities for interpretable egocentric video-language model. *Advances in Neural Information Processing Systems*, 37:86483–86499, 2024.

[58] Chi Wan, Yixin Cui, Jiatong Du, Shuo Yang, Yulong Bai, Peng Yi, Nan Li, and Yanjun Huang. Geminus: Dual-aware global and scene-adaptive mixture-of-experts for end-to-end autonomous driving. *arXiv preprint arXiv:2507.14456*, 2025.

[59] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5294–5306, 2025.

[60] Jie Wang, Guang Li, Zhijian Huang, Chenxu Dang, Hangjun Ye, Yahong Han, and Long Chen. Vggdrive: Empowering vision-language models with cross-view geometric grounding for autonomous driving. *arXiv preprint arXiv:2602.20794*, 2026.

[61] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[62] Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. Omnidrive: A holistic vision-language dataset for autonomous driving with counterfactual reasoning. In *Proceedings of the computer vision and pattern recognition conference*, pages 22442–22452, 2025.

[63] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20697–20709, 2024.

[64] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

[65] Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. Para-drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15449–15458, 2024.

[66] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. *arXiv preprint arXiv:2301.00493*, 2023.

[67] Diankun Wu, Fangfu Liu, Yi-Hsin Hung, and Yueqi Duan. Spatial-mllm: Boosting mllm capabilities in visual-based spatial intelligence. *arXiv preprint arXiv:2505.23747*, 2025.

[68] Zhiyu Wu, Xiaokang Chen, Zizheng Pan, Xingchao Liu, Wen Liu, Damai Dai, Huazuo Gao, Yiyang Ma, Chengyue Wu, Bingxuan Wang, et al. Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding. *arXiv preprint arXiv:2412.10302*, 2024.

[69] Shaoyuan Xie, Lingdong Kong, Yuhao Dong, Chonghao Sima, Wenwei Zhang, Qi Alfred Chen, Ziwei Liu, and Liang Pan. Are vlms ready for autonomous driving? an empirical study from the reliability, data and metric perspectives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6585–6597, 2025.

[70] Yi Xu, Yuxin Hu, Zaiwei Zhang, Gregory P Meyer, Siva Karthik Mustikovela, Siddhartha Srinivasa, Eric M Wolff, and Xin Huang. Vlm-ad: End-to-end autonomous driving through vision-language model supervision. *arXiv preprint arXiv:2412.14446*, 2024.

[71] Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K Wong, Zhenguo Li, and Hengshuang Zhao. Drivegpt4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*, 9(10):8186–8193, 2024.

[72] Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10632–10643, 2025.

[73] Shusheng Yang, Jihan Yang, Pinzhi Huang, Ellis L Brown II, Zihao Yang, Yue Yu, Shengbang Tong, Zihan Zheng, Yifan Xu, Muhan Wang, et al. Cambrian-s: Towards spatial supersensing in video. In *The Fourteenth International Conference on Learning Representations*, 2025.

[74] Zhenjie Yang, Yilin Chai, Xiaosong Jia, Qifeng Li, Yuqian Shao, Xuekai Zhu, Haisheng Su, and Junchi Yan. Drivemoe: Mixture-of-experts for vision-language-action model in end-to-end autonomous driving. *arXiv preprint arXiv:2505.16278*, 2025.

[75] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024.

[76] Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, Xing Wei, and Ning Guo. Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving. *arXiv preprint arXiv:2505.17685*, 2025.

[77] Jiang-Tian Zhai, Ze Feng, Jinhao Du, Yongqiang Mao, Jiang-Jiang Liu, Zichang Tan, Yifu Zhang, Xiaoqing Ye, and Jingdong Wang. Rethinking the open-loop evaluation of end-to-end autonomous driving in nuscenes. *arXiv preprint arXiv:2305.10430*, 2023.

[78] Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, et al. Theory of space: Can foundation models construct spatial beliefs through active exploration? *arXiv preprint arXiv:2602.07055*, 2026.

[79] Wei Zhang, Pengfei Li, Junli Wang, Bingchuan Sun, Qihao Jin, Guangjun Bao, Shibo Rui, Yang Yu, Wenchao Ding, Peng Li, et al. Dual-aeb: Synergizing rule-based and multimodal large language models for effective emergency braking. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, 2025.

[80] Duo Zheng, Shijia Huang, Yanyang Li, and Liwei Wang. Learning from videos for 3d world: Enhancing mllms with 3d vision geometry priors. *arXiv preprint arXiv:2505.24625*, 2025.

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

[82] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

[83] Xin Zhou, Dingkang Liang, Sifan Tu, Xiwu Chen, Yikang Ding, Dingyuan Zhang, Feiyang Tan, Hengshuang Zhao, and Xiang Bai. Hermes: A unified self-driving world model for simultaneous 3d scene understanding and generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 27817–27827, 2025.

Appendix

Table of Contents

A Geo-Bench Data Construction **0**

A.1 Data Preparation and Coordinate Unification 0

A.2 QA Construction 0

B Deatiled of Datasets and Metrics **1**

C Further Analysis **2**

D Additional Qualitative Results **2**

The supplementary material provides additional details on data construction, evaluation protocols, additional analysis and qualitative results to further support our main findings:

A. Geo-Bench Data Construction: We describe how Geo-Bench, the diagnostic benchmark used in Sec. 2, is constructed from five autonomous-driving datasets, including the coordinate unification procedure and the QA generation pipeline for evaluating depth understanding and cross-view correspondence.

B. Detailed Datasets and Metrics: We provide an in-depth description of the benchmarks used to evaluate OmniSpace, covering nuScenes, Bench2Drive, NuInstruct, OmniDrive, and DriveBench, along with their task setups and evaluation metrics.

C. Further Analysis: We further report the performance of OmniSpace under calibration noise and analyze geometry awareness through and comparisons with other Geometry-Aware AV MLLMs on Geo-Bench.

D. Additional Qualitative Results: We present further qualitative examples across NuInstruct, OmniDrive, nuScenes open-loop planning, and Bench2Drive closed-loop planning, with additional video results provided in the supplementary material.

A Geo-Bench Data Construction

This appendix describes how Geo-Bench, the diagnostic benchmark used in Section 2, is constructed. Geo-Bench is built from the validation splits of five autonomous-driving datasets (nuScenes [4], Argoverse 2 [66], Waymo [48], ONCE [40], and TruckScenes [8]) and is designed to isolate two complementary geometric abilities of MLLMs: *depth understanding*, evaluated through QA pairs, and *cross-view correspondence*, evaluated through a representation-level probe. We describe the data preparation, the QA generation, and the probing protocol below.

A.1 Data Preparation and Coordinate Unification

The five source datasets do not share a common annotation format. They differ in two key aspects: (i) the reference frame in which 3D bounding boxes are expressed, and (ii) the convention used for the yaw angle. To enable a consistent multi-view geometric evaluation, we unify all annotations into the nuScenes coordinate convention. For each non-nuScenes dataset, we apply a planar rotation R around the z-axis with offset α to every bounding-box center $\mathbf{c}_i$ and yaw θ_i:

$$\mathrm{R} = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \theta_i' = \theta_i + \alpha, \quad \mathbf{c}_i' = \mathrm{R}\,\mathbf{c}_i. \tag{5}$$

The angular offset α is set to $\pi/2$, $\pi/2$, π, and $3\pi/4$ for Argoverse 2, Waymo, ONCE, and TruckScenes, respectively. We rotate the camera extrinsic and ego-pose matrices accordingly so that all subsequent cross-view projections remain consistent. After unification, Geo-Bench draws from a shared pool of multi-view frames with consistent calibration, ego pose, and 3D box annotations.

A.2 QA Construction

The depth-understanding subset contains 250 multi-view distance-estimation questions covering two complementary settings: *ego-to-object*, where the model estimates the distance from the ego vehicle to a single referenced object, and *object-to-object*, where the model estimates the distance between two referenced objects observed across multi-view images. Each question shows the model the surround-view images of a frame, with referenced objects highlighted by colored 2D bounding-box overlays drawn directly on the camera images: a **blue** box labeled `(1)` for the first (or only) referenced object and a **magenta** box labeled `(2)` for the second referenced object. Ground-truth distances are computed directly from the unified 3D bounding boxes as $d_i^{\text{ego}} = \|\mathbf{c}_i^{\text{ego}}\|$ for the ego-to-object setting and $d_{ij} = \|\mathbf{c}_i - \mathbf{c}_j\|$ for the object-to-object setting.

Object sampling. For each scene, we restrict candidate objects to those (i) annotated as a vehicle, pedestrian, or cyclist, (ii) lying within $[3, 60]$ m of the ego vehicle, and (iii) projected onto a single

camera with at least 60% of their 2D bounding box visible. For object-to-object questions, we additionally require the two objects to appear in different camera views so that the answer cannot be inferred from a single image, and we require the inter-object distance to lie in $[5, 50]$ m. Objects are sampled to cover a broad range of distances rather than concentrating in any single regime, and we sample 50 questions per source dataset, balanced between the two question types.

Prompt templates. The prompts are identical across all evaluated MLLMs. Multi-view inputs are presented as an ordered sequence of camera-name tags interleaved with their corresponding images. The number of views varies across datasets (e.g., six for nuScenes and Argoverse 2, five for Waymo, seven for TruckScenes); we show one representative six-camera example below.

[System] You are a helpful assistant for autonomous driving. You will be given a question and a list of images, and you need to answer the question based on the images. Please first do your thinking process in `<think> </think>` and then provide the final answer in `<answer> </answer>` tags.

[User]
`<CAM_NAME_1>: <image> <CAM_NAME_2>: <image> ... <CAM_NAME_K>: <image>`

(Ego-to-Object) Given the current driving images, estimate the approximate distance from the marked object (highlighted with a blue bounding box) to our vehicle (ego vehicle / camera origin).

(Object-to-Object) Given the current driving images, estimate the approximate distance between Object-1 (marked with blue bounding box and label "(1)") and Object-2 (marked with magenta bounding box and label "(2)").

Provide your answer as a single numerical value in meters (e.g., 15.5).

Example camera tags for six-view datasets:
`CAM_FRONT_LEFT`, `CAM_FRONT`, `CAM_FRONT_RIGHT`, `CAM_BACK_LEFT`, `CAM_BACK`, `CAM_BACK_RIGHT`.

B Deatiled of Datasets and Metrics

To demonstrate the effectiveness of OmniSpace in injecting geometric awareness into MLLMs, we evaluate OmniSpace across various attributes within the autonomous driving domain. These benchmarks cover task-specific scenarios, including open-loop planning, closed-loop planning, scene understanding, perception, cross-view risk detection, and generalization.

nuScenes [4]. The nuScenes dataset comprises 1,000 urban driving scenes with full-stack 360° sensing (6 cameras, 1 LiDAR, and 5 radars). For open-loop planning, we predict 6 waypoints within a 3 s horizon and evaluate: (i) *waypoint displacement* (L2 error), (ii) *collision rate* (the fraction of future timestamps overlapping with any dynamic agent), and (iii) *intersection rate* (the fraction of timestamps intruding into non-drivable map regions).

Bench2Drive [25]. Bench2Drive is a closed-loop planning benchmark emphasizing interactive scenarios (e.g., merging, overtaking, yielding, and emergency negotiation) in a deterministic CARLA V2 [7] simulator. Our closed-loop evaluation adopts the official protocol of 220 short routes, covering 44 interactive scenarios, with 5 distinct routes defined for each scenario. Closed-loop metrics include *Driving Score* (route progress penalized by safety infractions) and *Success Rate* (the percentage of scenarios completed without terminal violation). All reported results adopt identical horizon, temporal sampling, footprint inflation, and map definitions for fair comparison.

NuInstruct [6]. NuInstruct samples a total of 11,850 keyframes from 850 videos within the nuScenes dataset. It includes a wide range of challenging samples, such as cross-view risk perception, target distance estimation, agent and ego state prediction, target motion prediction, and reasoning tasks. The dataset effectively reflects a model's ability to perform cross-view understanding and analyze 3D geometric scene information. NuInstruct employs a variety of metrics to evaluate different tasks, including *Mean Absolute Error* (MAE) for regression tasks, *accuracy* for classification tasks, *mean Average Precision* (mAP) for detection tasks, and a rule-based *BLEU* metric for captioning tasks.

OmniDrive [62]. OmniDrive is also built upon the nuScenes dataset and primarily focuses on scene description, attention-based question answering, counterfactual reasoning question answering, decision-planning question answering, and general dialogue. Due to limitations in the availability of evaluation data and metrics, we follow [60] select scene description and general dialogue from this dataset to train and evaluate the model's ability to understand autonomous driving scene captions.

The dataset primarily includes three language evaluation metrics: *BLEU*, *CIDEr*, and *ROUGE*, with the "average" representing the mean of the three metrics.

DriveBench [69]. DriveBench is a benchmark dataset designed to assess VLM reliability across 17 settings, including clean, corrupted, and text-only inputs. It comprises 19,200 frames, 20,498 question–answer pairs, three question types, and four mainstream driving tasks (*perception*, *prediction*, *planning*, and *behavior explanation*), and evaluates 12 popular VLMs. Built upon DriveLM [45], DriveBench employs GPT-3.5-turbo as the judge model to score open-ended responses, and we follow the same protocol when evaluating our model.

C Further Analysis

Table S-I: Effect of calibration noise on performance.

Calibration Noise	L2 ↓	Collision (%) ↓	Intersection (%) ↓
$\pm 10°$; ± 10 cm	0.30	0.22	2.12
$\pm 5°$; ± 5 cm	0.29	0.21	1.28
$\pm 0°$; ± 0 cm	**0.28**	**0.19**	**1.27**

Table S-II: Geometry-awareness comparison on Geo-Bench.

Model	Cross-View Correspondence (%) ↑	Depth Estimation (m) ↓
VGGDrive	48.65	43.36
SpaceDrive	27.25	54.36
OmniSpace (ours)	**59.8**	**14.4**

Robustness of under calibration noise

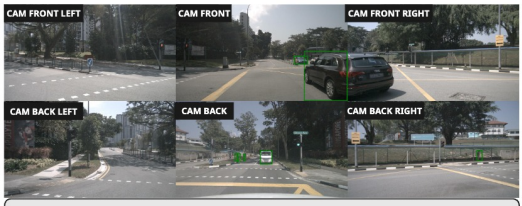
Since OmniSpace relies on calibrated camera geometry for Multi-view Epipolar Attention, we further evaluate its sensitivity to calibration errors by perturbing both the rotational and translational components of the camera calibration. As shown in Table S-I, performance degrades only moderately as the perturbation magnitude increases. Under $\pm 5°$ rotation and ± 5 cm translation noise, the L2 error increases from 0.28 to 0.29, while the collision and intersection rates change only slightly. Even under the stronger $\pm 10°$ and ± 10 cm perturbation, the model maintains stable performance. This robustness is partly attributed to the dilated epipolar region used in our attention mechanism, which helps absorb small calibration inaccuracies. These results indicate that OmniSpace is reasonably robust to moderate calibration noise and sensor drift of similar magnitude.

Comparison with Geometry-Aware AV MLLMs

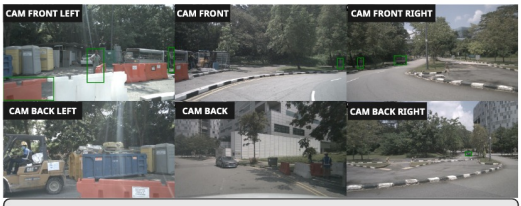
We further compare OmniSpace with existing geometry-aware AV MLLMs on Geo-Bench by reproducing VGGDrive [60] and SpaceDrive [31] under the same evaluation protocol used for OmniSpace. Since the training code for VGGDrive is not publicly available and SpaceDrive is originally designed only for trajectory estimation, we adapt both methods to Geo-Bench and fine-tune them on DriveLM for 800 steps using the same setup as OmniSpace. As shown in Table S-II, OmniSpace achieves 59.8% cross-view correspondence accuracy and a depth estimation error of 14.4 m, outperforming VGGDrive (48.65%, 43.36 m) and SpaceDrive (27.25%, 54.36 m). These results provide direct evidence that OmniSpace improves the two geometric capabilities targeted by our design, namely cross-view correspondence and depth estimation, which is consistent with the downstream AV performance observed in our experiments.

D Additional Qualitative Results

Additional qualitative results are shown in Figure S-I for NuInstruct, Figures S-II and S-III for OmniDrive, Figure S-IV for nuScenes Open-Loop Planning, and Figure S-V for Bench2Drive. Additional qualitative video results for planning tasks are also provided in the supplementary material.



Q: What is the distance between <car>[CAM_FRONT,1098,481,1230,561] and the ego car?
GT: (29.58, -7.39) **Pred: (30.5, -7.41)**
Q: What is the distance between <pedestrian>[CAM_BACK,569,486,604,566] and the ego car?
GT: (-18.73, -5.55) **Pred: (-19.2, -6.0)**
Q: What is the distance between <pedestrian>[CAM_BACK_RIGHT,1199,456,1255,557] and the ego car?
GT: (-13.43, -18.01) **Pred: (-11.7, -15.85)**
Q: What is the status for the ego car?
GT: the ego no turn and keep speed
Pred: the ego no turn and keep speed
Q: What is the status for <pedestrian>[CAM_BACK,546,483,585,559]?
GT: stay still **Pred: stay still**
Q: What is the next motion for the ego? And give the reason.
GT: There are no objects that would have risks for the ego car. Hence, The ego would keep speed and go straightforward.
Pred: There are no objects that would have risks for the ego car. Hence, The ego would keep speed and go straightforward.



Q: What is the closest object near to the ego car?
GT: <barrier>[CAM_FRONT_LEFT,0.0,0.71,0.28,1.0].
Pred: <barrier>[CAM_FRONT_LEFT,0.0,0.75,0.3,1.0]
Q: What is the closest object in the back of the ego car?
GT: <car>[CAM_BACK_RIGHT,0.7,0.53,0.72,0.57].
Pred: <car>[CAM_BACK_RIGHT,0.69,0.52,0.73,0.57]
Q: What is the distance between <pedestrian>[CAM_FRONT_RIGHT,100,497,161,598] and the ego car?
GT: (22.89, -11.73) **Pred: (25.5, -13.4)**
Q: What is the status for <truck>[CAM_FRONT_RIGHT,441,477,561,566]?
GT: parked with no immediate intent to move. **Pred: stay still**
Q: What is the status for the ego car?
GT: the ego no turn and keep speed
Pred: the ego no turn and keep speed
Q: What is the next motion for the ego? And give the reason.
GT: There are no objects that would have risks for the ego car. Hence, The ego would keep speed and go straightforward.
Pred: There are no objects that would have risks for the ego car. Hence, The ego would keep speed and go straightforward.

Figure S-I: OmniSpace's results on NuInstruct compared to ground truths



Q: Can you provide a summary of the current driving scenario based on the input images?"
Pred: The vehicle is situated in a bustling urban environment, surrounded by the dynamic energy of city life. To the left, the skyline is punctuated by construction cranes, signaling ongoing development, while the road is lined with parked cars and a few pedestrians, hinting at the area's commercial activity. Directly ahead, the road is clear, inviting a steady drive, flanked by more parked cars and a pedestrian crossing, suggesting a need for vigilance for any crossing individuals. To the right, ...
Q: Considering the driving conditions, what precautions should we take?
Pred: We should maintain a moderate speed, stay alert for construction-related changes, keep a safe distance from the vehicle ahead, and be prepared for potential weather changes.
Q: What additional considerations should a driver have in this scenario?
Pred: The driver should be prepared for sudden stops due to traffic, watch for construction workers or equipment entering the roadway, and be ready to follow any temporary traffic signs or signals.
Q: If we wanted to change lanes right now, what should we be aware of?
Pred: If we wanted to change lanes, we should be aware of the traffic cones and the 'ONLY' lane markings, ensuring we are in the correct lane for our intended direction. We should also check for any vehicles in the blind spot and signal our intention to change lanes well in advance.

Figure S-II: An illustration of counterfactual reasoning ability of our method on OmniDrive dataset



Q: If there's an emergency vehicle approaching, what should be the driver's immediate action?
A: **If an emergency vehicle approaches, the driver should pull over to the right side of the road, ensuring a clear path for the emergency vehicle to pass safely**
Q: Given the clear path ahead, what is the best practice for merging into the underpass?
A: **The best practice for merging into the underpass would be to signal early, check mirrors and blind spots for any vehicles or cyclists, and smoothly integrate into the flow of traffic while maintaining a safe speed.**

Figure S-III: Another examples of counterfactual reasoning of our proposed solution

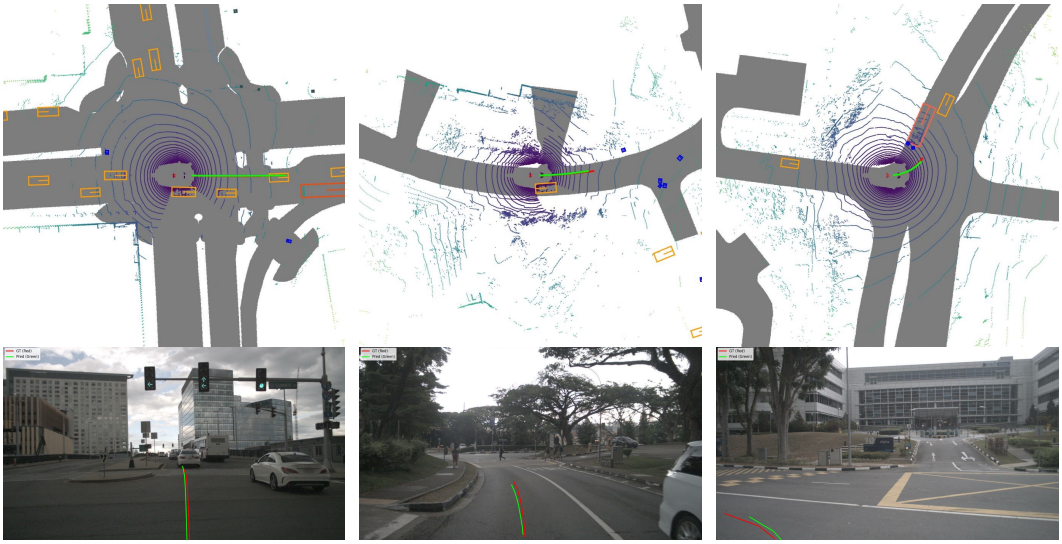


Figure S-IV: Qualitative results of our OmniSpace on planning task in nuScenes dataset



Figure S-V: Qualitative results of our OmniSpace on planning tasks in the Bench2Drive dataset.

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: We provided diagnostic motivation and experiments to support our claims.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: We included it at the end of the paper.

   Guidelines:

   - The answer [N/A] 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: [N/A].

Justification: All assumptions are already stated.

Guidelines:

- The answer [N/A] 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: We provided experiment descriptions in the main paper and appendix. Furthermore, we will release our implementation on GitHub upon acceptance.

   Guidelines:

   - The answer [N/A] 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 will provide open access to the data and code as well as complete instructions to faithfully reproduce the results upon acceptance. Also, we can provide code and data through an anonymous link if some reviewers ask for it.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes].

   Justification: We provided details for experiments in the paper and appendix.

   Guidelines:

   - The answer [N/A] 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: [N/A].

   Justification: We follow the *de facto* protocols used by previous works for fair comparisons, which do not include information about statistical significance of the experiments.

   Guidelines:

   - The answer [N/A] 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 paper reports the computational cost for all methods and the resource requirement for implementation.

   Guidelines:

   - The answer [N/A] 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: All authors have reviewed the paper and NeurIPS Code of Ethics to ensure that the paper conforms with the NeurIPS Code of Ethics in every respect.

   Guidelines:

   - The answer [N/A] 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: [N/A].

    Justification: Our work helps to reduce carbon footprint while injecting the geometric information without adding additional computational cost.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A].

    Justification: Our work does not pose such risks.

    Guidelines:

    - The answer [N/A] 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: We have properly cited papers and resources used in our experiments.

    Guidelines:

    - The answer [N/A] 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: [N/A].

    Justification: Our paper does not release new assets.

    Guidelines:

    - The answer [N/A] 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: The experiments were performed on publicly available datasets. The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] 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: The experiments were performed on publicly available datasets. The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

    Answer: [N/A].

    Justification: The paper used LLMs to support development for generating data, which was then verified by humans.

    Guidelines:

    - The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.