SpaceDrive: Infusing Spatial Awareness into VLM-based Autonomous Driving

Supplementary Material

The main content of this supplementary material is organized as follows:

- Section A: More implementation details of our method;
- Section B: Additional experiments and ablation studies;
- Section C: Additional visualization comparisons and qualitative analysis.

A. Additional Implementation Details

A.1. SpaceDrive Framework

To ensure seamless adaptability, our method avoids any model-specific customization and fully preserves the original image preprocessing, patchification strategy, text tokenization, and chat template used by each base VLM model. Given the shape of the preprocessed visual patches, the depth map is resized accordingly using min-pooling (*e.g.* patch shapes of $6 \times 24 \times 24$ for LLaVA-1.5-7B [41] and $6 \times 23 \times 23$ for Qwen2.5-VL-7B [2] in our configuration). Newly introduced tokens, such as $\langle \text{IND} \rangle$, are set as learnable and appended to the frozen input embedding layer and output language-model head. The PE decoder for coordinates is implemented as a standard two-layer MLP with the same hidden dimensionality as the base VLM. In all experiments, we set the seed to 888. The LoRA configurations are listed in Tab. A.

A.2. Training Details

Open-loop Planning For open-loop planning, we follow prior works and use 6 future trajectory points sampled at 2 Hz over a 3-second horizon as ground truth supervision. As emphasized in previous studies [38, 74], strong open-loop planning performance can be achieved using only ego-status. To rigorously validate the effectiveness of our framework, the standard SpaceDrive variant intentionally excludes motion dynamics and high-level driving commands (*e.g.* "go straight", "turn right") from its inputs. In this configuration, the model performs trajectory planning exclusively from image observations, enabling a clean evaluation of the spatial reasoning capability brought by our design. The variant SpaceDrive+ includes the current commands and ego dynamics of past 2 frames that are widely used in other works [12, 61].

For VQA training and evaluation, we adopt the dataset provided by OmniDrive [61], which includes scene description, attention, counterfactual reasoning, planning, as well as other general conversations. Consistent with the implementation of OmniDrive, the other VQA tasks are appended subsequent to the trajectory planning task to ensure semantic stability.

Table A. **LoRA configurations for VLM fine-tuning.**

Setting	Rank (r)	Alpha (α)	Dropout	Target Modules
Value	16	16	0.05	q_proj, k_proj, v_proj, o_proj

Table B. **Counterfactual reasoning comparison in the open-loop planning (without ego status).** P and R here stand for Precision and Recall. Results are highlighted in **bold** and underline for the best and the second-best performance.

Method	Safe		Red Light		Collision		Drivable Area	
	P	R	P	R	P	R	P	R
BEV-MLP	70.2	17.3	48.7	53.6	31.1	70.4	32.4	56.6
Omni-L [61]	**72.1**	58.0	59.2	63.3	34.3	71.3	49.1	**59.2**
Omni-Q [61]	70.7	49.0	57.6	58.3	32.3	**72.6**	48.5	58.6
SpaceDrive (**ours**)	65.7	**63.6**	**70.3**	**72.7**	**37.5**	66.4	**55.0**	37.0

Closed-loop Planning Inspired by SimLingo [50], we augment the supervision of 6 trajectory points with 20 additional path waypoints, uniformly spaced at 1-meter intervals. In this setup, the trajectory points serve two purposes: estimating the target speed and identifying the appropriate waypoint for the target direction. This leads to generally more stable steering regardless of whether the ego vehicle is moving or not. Two PID contollers are applied to determine acceleration and steering, respectively. During training, we use a subset of SimLingo routes containing 3600 episodes with PDM-lite as the expert driver.

B. Additional Experiments and Analyses

B.1. VQA for Counterfactual Reasoning

As aforementioned, to validate the spatial reasoning capabilities of SpaceDrive, we conduct counterfactual reasoning experiments following the setting in OmniDrive [61], as presented in Tab. B. In this evaluation, keywords such as "safety", "collision", "running a red light", and "out of the drivable area" are extracted from the VQA outputs and compared against ground truth keywords to compute Precision and Recall. The results demonstrate that our framework achieves superior performance across the majority of metrics, *e.g.* a Recall of 63.6% in the safety task. It is particularly noteworthy that, without any specific prompt engineering for the dialogue, the mere incorporation of the unified 3D spatial representation enables significantly higher Precision in tasks demanding rigorous spatial understanding, such as Collision (37.5%) and Drivable Area (55.0%). This further confirms our SpaceDrive possesses strong spatial reasoning capabilities.

Table C. **Ablation of depth estimator.**

$f_{dep.}$	Avg. L2 ↓	Avg. Collision ↓	Avg. Intersection ↓
DepthAnythingV2 [68]	1.76	1.95	3.96
UniDepthV2 [49]	1.80	1.88	4.21

Table D. **Ablation of LoRA rank.** Learn. Par. is the abbreviation for the number of LoRA parameters when selecting Qwen2.5-VL-7B as the base VLM.

Rank	Learn. Par.	Avg. L2 ↓	Avg. Collision ↓	Avg. Intersection ↓
16	10.09M	1.80	1.88	4.21
64	40.37M	1.88	2.13	4.08
128	80.74M	1.82	2.25	4.68

Table E. **Ablation of PE frequency.**

Frequency	Avg. L2 ↓	Avg. Collision ↓	Avg. Intersection ↓
$1000^{-2i/d_a}$	1.78	2.01	3.93
$10000^{-2i/d_a}$	1.83	1.83	3.18
$20000^{-2i/d_a}$	1.80	1.88	4.21

Table F. **Ablation of regression loss.**

$\mathcal{L}_{\text{reg.}}$	Avg. L2 ↓	Avg. Collision ↓	Avg. Intersection ↓
MAE	1.86	1.82	5.73
MSE	1.82	2.14	6.22
Huber Loss	1.80	1.88	4.21

B.2. More Ablation Studies

Depth Estimator In Tab. C, we compare the influence of different pre-trained depth estimator on the planning performance. DepthAnythingV2 [68] and UniDepthV2 [49] are selected as representative examples of relative and metric depth estimation models, respectively. We observe that both variants perform similarly on the L2 error metric and Collision rate, which are the most reliable indicator of planning performance. This suggests that the effectiveness of our SpaceDrive is independent of a specific pre-trained depth model, implicitly demonstrating the adaptability of our framework. Notably, LiDAR-based depth ground truth (GT) is inherently sparse and lacks valid depth values in regions such as the sky, necessitating manual definition. Together with factors like camera distortion and projection error, GT-based comparisons are unreliable and thus excluded from the comparison.

LoRA Rank Table D presents a comparison of different LoRA [17] ranks in the VLM during fine-tuning. Benefiting from our universal spatial positional encoding, the coordinate regression process in the language model is simplified. Utilizing only low-rank fine-tuning (rank 16) achieves the optimal overall result (L2 error of 1.80, Collision rate of 1.88%, and Intersection rate of 4.21%). While increasing the rank to 128 substantially raises the number of learnable parameters from 10.09M to 80.74M, it fails to improve the planning accuracy and, instead, leads to a degradation in Collision and Intersection rates. We attribute this to the excessive degrees of training freedom in the high-rank adapter, which hinders the convergence. The above comparison further demonstrates that our method not only offers stronger planning reliability but also maintains parameter efficiency.

PE Frequency Table E investigates the base frequency of the Sin-Cos PE, which impacts both encoding resolution and smoothness. Utilizing a smaller frequency base (corresponding to a higher frequency) introduces a larger phase shift between adjacent positions but leads to positional aliasing at long distances, thereby inhibiting the representation of far-field positions. As shown in the comparison, setting the base to 1000 enhances local resolution and achieves the lowest L2 error of 1.78. However, distant coordinates exhibit near-random phase characteristics, which compromises overall safety (leading to worse Collision and Intersection rates). Conversely, an excessively large base (*e.g.* 20000) generates smoother, more stable encodings over long distances but diminishes local discriminative capability. Compared to the original base of 10000, the resulting L2 error reduction is less pronounced, at only -0.03, but the collision rate increase is negligible. Overall, comparing all variants reveals that the influence of different PE frequencies is relatively limited and non-decisive. We finally adopt 20000 as our PE frequency base.

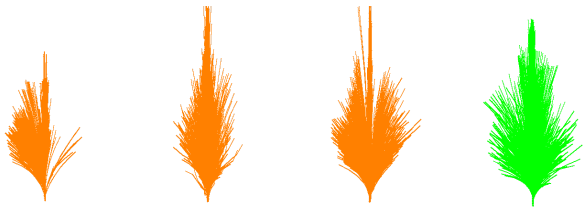
Regression Loss In Tab. F, we compares different regression losses for trajectory prediction. MAE provides robustness to outliers but yields the worst L2 and intersection metrics, suggesting insufficient pressure on medium-scale errors. MSE reduces L2 compared to MAE, but its quadratic growth on large residuals makes optimization more sensitive to outliers, leading to noticeably higher collision and intersection rates. Huber loss strikes a balance between them and achieves the best L2 error together with markedly improved safety metrics. So we adopt Huber loss as our final regression objective.

B.3. Comprehensive Benchmark

Constrained by the limited space in the main paper, we list only the primary relevant works in the benchmark comparisons. Therefore, we provide more comprehensive benchmark comparisons for open-loop and closed-loop planning in Tab. G and Tab. H, respectively. It is worth noting that existing nuScenes [4] open-loop evaluations utilize differing sets of metrics in different studies. While the main paper employs the OmniDrive [61] version commonly used by VLM-based frameworks, Table G provides results derived using the evaluation metrics from ST-P3 [18] and UniAD [19].

Table G. **Open-loop planning results on nuScenes [4].** SpaceDrive+ denotes the adoption of the ego planner input. ‡: The model is trained using only the trajectory prediction task for open-loop planning, without utilizing our generated OmniDrive Q&A data. Methods marked as Hybrid Paradigm here stack traditional and VLM-based approaches, and are thus incomparable. Results are highlighted in **bold** and underline for the best and the second-best performance among VLM-based methods.

Method	Ego Status		ST-P3 Metrics								UniAD Metrics							
			L2 (m) ↓				Collision (%) ↓				L2 (m) ↓				Collision (%) ↓			
	BEV	Planner	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Traditional Modular Paradigm																		
ST-P3 [18]	-	-	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71	1.72	3.26	4.86	3.28	0.44	1.08	3.01	1.51
UniAD [19]	-	-	0.44	0.67	0.96	0.69	0.04	0.08	0.23	0.12	0.48	0.96	1.65	1.03	0.05	0.17	0.71	0.31
VAD-Base [27]	-	-	0.41	0.70	1.05	0.72	0.07	0.17	0.41	0.22	0.54	1.15	1.98	1.22	0.10	0.24	0.96	0.43
UAD [14]	-	-	0.28	0.41	0.65	0.45	0.01	0.03	0.14	0.06	0.39	0.80	1.50	0.90	0.01	0.12	0.43	0.19
MomAD [55]	-	-	0.28	0.49	0.78	0.52	0.08	0.14	0.34	0.19	0.36	0.83	1.56	0.91	0.06	0.23	1.00	0.43
GenAD [80]	-	✓	0.31	0.57	0.91	0.60	0.01	0.05	0.22	0.09	0.43	0.88	1.62	0.98	0.06	0.16	0.68	0.30
Drive-WM [63]	✓	✓	0.43	0.77	1.20	0.80	0.10	0.21	0.48	0.26	-	-	-	-	-	-	-	-
SparseDrive [57]	-	✓	0.29	0.55	0.91	0.58	0.01	0.02	0.13	0.06	0.44	0.92	1.69	1.01	0.07	0.19	0.71	0.32
DiffusionDrive [40]	-	✓	0.27	0.54	0.90	0.57	0.03	0.05	0.16	0.08	-	-	-	-	-	-	-	-
VLM-based Paradigm																		
EMMA [22]	-	-	**0.14**	**0.29**	0.54	**0.32**	-	-	-	-	-	-	-	-	-	-	-	-
RDA-Driver [21]	✓	✓	0.17	0.37	0.69	0.40	**0.01**	**0.05**	0.26	**0.10**	0.23	0.73	1.54	0.80	**0.00**	**0.13**	0.83	**0.32**
DriveVLM [59]	-	✓	0.18	0.34	0.68	0.40	-	-	-	-	-	-	-	-	-	-	-	-
ORION [12]	✓	-	0.17	0.31	0.55	0.34	-	-	-	-	-	-	-	-	-	-	-	-
OmniDrive-Q [61]	-	-	1.15	1.96	2.84	1.98	-	-	-	-	-	-	-	-	-	-	-	-
OmniDrive-Q++ [61]	✓	✓	**0.14**	**0.29**	0.55	0.33	-	-	-	-	-	-	-	-	-	-	-	-
OmniDrive-L‡ [61]	-	-	1.47	2.43	3.38	2.43	-	-	-	-	-	-	-	-	-	-	-	-
OmniDrive-L++‡ [61]	-	✓	0.31	0.62	1.06	0.66	-	-	-	-	-	-	-	-	-	-	-	-
OmniDrive-L [61]	-	-	1.43	2.34	3.24	2.34	-	-	-	-	-	-	-	-	-	-	-	-
OmniDrive-L++ [61]	-	✓	0.15	0.36	0.70	0.40	-	-	-	-	-	-	-	-	-	-	-	-
SpaceDrive (**ours**)	-	-	1.06	1.79	2.55	1.80	0.35	0.61	1.31	0.76	1.41	2.88	4.51	2.93	0.59	1.72	4.53	2.28
SpaceDrive+ (**ours**)	-	✓	0.15	**0.29**	**0.51**	**0.32**	0.05	0.08	**0.16**	**0.10**	**0.20**	**0.53**	**1.13**	**0.62**	0.10	0.31	**0.80**	0.40
Hybrid Paradigm																		
VLP [48]	✓	-	0.30	0.53	0.84	0.55	0.01	0.07	0.38	0.15	0.36	0.68	1.19	0.74	0.03	0.12	0.32	0.16
ReAL-AD [46]	✓	-	0.30	0.48	0.67	0.48	0.07	0.10	0.28	0.15	0.40	0.71	1.14	0.77	0.02	0.12	0.37	0.17
DriveVLM-Dual [59]	✓	-	0.15	0.29	0.48	0.31	0.05	0.08	0.17	0.10	-	-	-	-	-	-	-	-
SOLVE-VLM [7]	✓	-	0.13	0.25	0.47	0.28	-	-	-	-	-	-	-	-	-	-	-	-
Senna [28]	✓	-	0.11	0.21	0.35	0.22	0.04	0.08	0.13	0.08	-	-	-	-	-	-	-	-



(a) OmniDrive-L (b) SpaceDrive (c) SpaceDrive-CoT (d) Ground Truth

Figure A. **Trajectory distribution of open-loop planning for different frameworks and ground truth.**

C. Additional Visualization

C.1. More Adaptability Analysis

Figure A illustrates the distribution of planned trajectories across all scenarios under the open-loop setting of nuScenes [4]. We first analyze the output trajectory distribution of OmniDrive-L [61], a typical scheme utilizing textual digit tokens for waypoint coordinates, shown in Fig. A.a. Due to the VLM's limitations in numerical processing, as discussed in Section 1, OmniDrive-L exhibits clear mode collapse for right-turn cases. In sharp contrast, our SpaceDrive, which is based on the universal 3D PE representation, significantly mitigates this issue, as shown in Fig. A.b. Furthermore, when adopting inference techniques such as Chain-of-Thought during inference, the output trajectory planning demonstrates enhanced robustness (Fig. A.c) and closer alignment with the ground truth distribution (Fig. A.d). This result further supports the strong adaptability of our method to language model inference techniques.

C.2. Failure Analysis of Textual Coordinate Output

A further quantitative analysis is conducted to assess the driving capability of conventional VLM-based models that output trajectory coordinates as textual digit tokens in closed-loop simulation, as shown in Fig. B. We the exact same scenario as in Fig. 3 and employ OmniDrive-L [61], a framework structurally analogous to SpaceDrive, utilizing the same closed-loop training configuration as in Sec. A.2. This figure clearly illustrates that in the closed-loop setting, the

Figure B. **Qualitative results of OmniDrive-L [61] in closed-loop setting on Bench2Drive [25].** Green and pink dots represent path and speed waypoints, respectively. Parameters such as speed and steering wheel angle can be found in the figures.

planned trajectories generated by OmniDrive-L collapse into an approximately straight line, and the directional control exhibits random oscillation. This phenomenon aligns with the mode collapse previously observed during open-loop evaluation (See Sec. C.1). Critically, this oscillation is amplified over time, leading to vehicle instability and ultimately making the vehicle veer off the road and collide with the guardrail. This result provides strong empirical support for our analysis in Sec. 4.2: purely text-based trajectory coordinate output from VLMs is inadequate for reliable closed-loop driving.

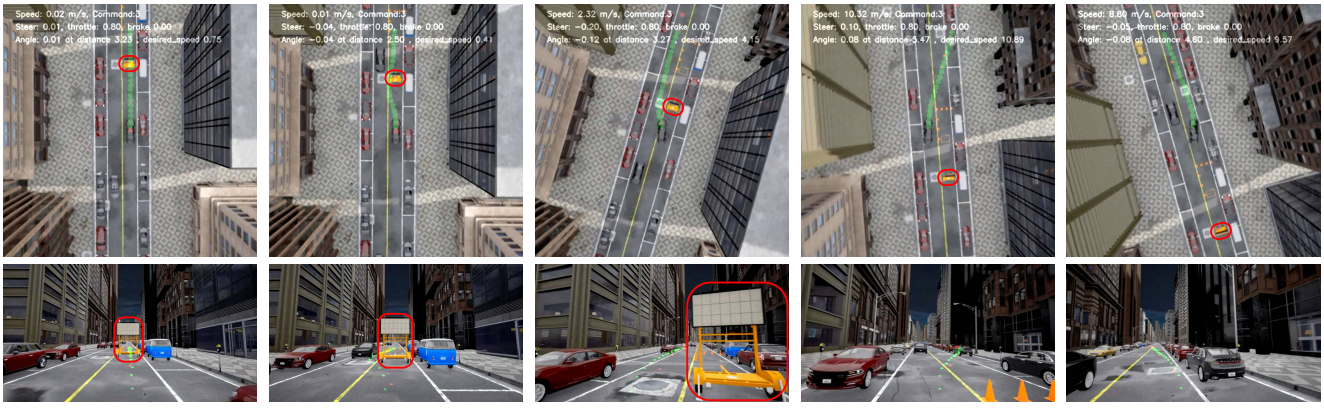
C.3. More Qualitative Closed-Loop Results

We present additional closed-loop simulation visualizations for SpaceDrive in Fig. C, covering 3 representative safety-critical scenarios: (a) navigating around a construction zone requiring a brief excursion into the oncoming lane; (b) decelerating and yielding due to a sudden pedestrian crossing during normal driving; and (c) performing an emergency stop and yielding to an ambulance rapidly approaching from the rear. All these scenarios demand the model to quickly establish a deep understanding of the 3D spatial context and generate a sound trajectory in a minimal timeframe. The visualizations clearly indicate that our proposed framework, by leveraging its unified 3D representation, effectively manages these critical, unforeseen situations. This further substantiates the efficacy of our proposed SpaceDrive framework.

Table H. **Closed-loop planning results on Bench2Drive [25].** Results are highlighted in **bold** and underline for the best and the second-best performance among VLM-based methods.

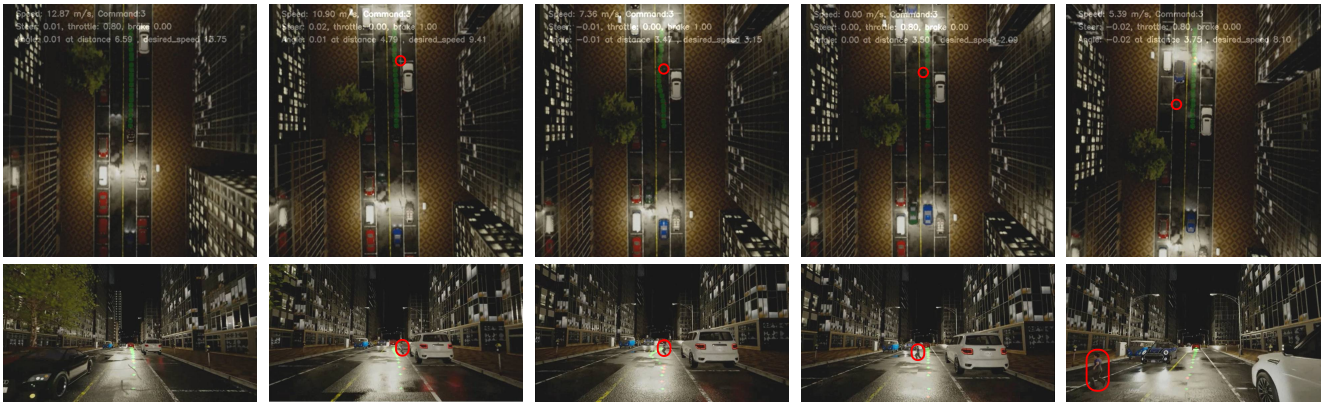
Method	Closed-loop Metric	
	Driving Score ↑	Success Rate(%) ↑
Traditional Modular Paradigm		
AD-MLP [74]	18.05	0.00
UniAD-Base [19]	45.81	16.36
VAD-Base [27]	42.35	15.00
MomAD [55]	44.54	16.71
GenAD [80]	44.81	15.90
SparseDrive [57]	47.38	17.72
UAD [14]	49.22	20.45
SeerDrive [75]	58.32	30.17
WoTE [35]	61.71	31.36
DriveDPO [51]	62.02	30.62
ThinkTwice [24]	62.44	37.17
DriveTransformer-L [26]	63.46	38.60
DriveAdapter [23]	64.22	42.08
Raw2Drive [70]	71.36	50.24
Hydra-NeXt [39]	73.86	53.22
DiffusionDrive [40]	77.68	52.72
PGS [20]	78.08	48.64
GaussianFusion [43]	79.10	54.40
TF++ [84]	84.21	64.39
R2SE [42]	86.28	67.76
HiP-AD [58]	86.77	69.09
VLM-based Paradigm		
ReAL-AD [46]	41.17	11.36
Dual-AEB [77]	45.23	10.00
X-Driver [44]	51.70	18.10
GEMINUS [60]	65.39	37.73
VDRive [15]	66.25	50.51
StuckSolver [3]	70.89	50.01
DriveMoE [69]	74.22	48.64
ETA [16]	74.33	48.33
VLR-Drive [30]	75.01	50.00
ORION [12]	77.74	54.62
SimLingo [50]	**85.07**	**67.27**
SpaceDrive+ (**ours**)	<u>78.02</u>	<u>55.11</u>

- (a) Navigate around the construction roadblock



Slow down before construction. | Wait for oncoming traffic. | Quickly change lane past. | Return to lane promptly. | Resume straight-line travel.

- (b) Yield to the sudden appearance and road crossing



Normal straight-line driving. | See crossing, quickly decelerate. | Attempt to avoid collision. | Brake to full stop and wait. | Accelerate after confirming safety.

- (c) Yield to the ambulance coming from behind

Normal straight-line driving. | See ambulance, try to pull over. | Decelerate and pull aside. | Yield to the ambulance. | Resume normal driving.

Figure C. **More qualitative results of closed-loop evaluation on Bench2Drive [25].** We include 3 scenarios here to demonstrate the closed-loop planning capability of SpaceDrive: (a) urban road construction; (b) a sudden pedestrian crossing; (c) yielding to an ambulance. Green and pink dots represent path and speed waypoints, respectively. Red circles indicate objects requiring attention in the scenario. Parameters such as speed and steering wheel angle can be found in the figures.