Supplementary Materials for "DrivePI: Spatial-aware 4D MLLM for Unified Autonomous Driving Understanding, Perception, Prediction and Planning"

Supplementary Material

In this supplementary material, we present more ablation studies, the details of coarse-grained spatial understanding, the details of fine-grained spatial learning, and more visualizations in Sections 1, 2, 3 and 4, respectively.

1. More Ablation Studies

Table 1. Ablation study for the balancing weights in DrivePI.

#	*Occ. Weight*	*Flow Weight*	*3D Occ.* RayIoU↑	*Occ. Flow* mAVE↓	*Planning* L2↓	*Planning* Col.↓	*QA* Acc.↑
I	0.2	0.2	48.1	0.57	0.46	0.19	**61.1**
II	0.5	0.5	**49.3**	0.54	0.49	0.40	60.9
III	1.0	1.0	**49.3**	**0.51**	**0.50**	**0.38**	60.7

Different Balancing Weights. In DrivePI, we adopt multi-task learning to train our model in an end-to-end manner. Here, we conduct experiments to investigate the effect of different loss balancing weights on overall performance. Through preliminary analysis, we observe that the 3D occupancy and occupancy flow losses dominate the total loss function, accounting for over 60% of the combined loss magnitude. To mitigate potential optimization imbalances, we systematically reduce their weights from the default value of 1.0 (*III*) to 0.5 (*II*) and 0.2 (*I*) for both 3D occupancy and occupancy flow tasks. The corresponding results are presented in Table 1. We find that higher occupancy and flow weights yield improved performance on 3D occupancy and occupancy flow, but result in slight degradation in planning accuracy (L2 error) and text understanding in the QA task. Therefore, setting proper balancing weights in multi-task learning remains challenging. For simplicity, we adopt the default weight of 1.0 for both 3D occupancy and flow losses in our final implementation.

Learned Weights of Different Hidden States. For the MLLM (*i.e.*, Qwen2.5-0.5B/3B model), we investigate *what is the importance weight of each hidden state of MLLMs in DrivePI?* To answer this question, we conduct an analysis of the learned importance weights of all hidden states in the MLLM, as shown in Table 2. Specifically, we replace the default setting of using only the last hidden state with a weighted combination $h = \sum_{i=0}^{l} F_i^h \cdot w_i$, where $F_i^h \in \mathbb{R}^{N \times C}$ represents the features of the i-th hidden state, $w_i \in \mathbb{R}^{1 \times 1}$ is the corresponding learnable importance weight, and l denotes the total number of hidden states in the MLLM. Here, the 0.5B (or 3B) MLLM contains 24 (or 36) transformer layers, and when including the input embeddings, we obtain a total of 25 (or 37) hidden states with corresponding learnable weights. As shown in Table 2 and Table 3, we observe an overall trend where

Table 2. The learned importance weights of all hidden states in the MLLM with Qwen-2.5 0.5B model, including the input embedding (indexed as 0). The ***Index*** and **Weight** column indicates the index and the learned importance weight of each hidden state.

Index	**Weight**	***Index***	**Weight**	***Index***	**Weight**
0	0.0328	*10*	0.0375	*20*	0.0463
1	0.0332	*11*	0.0381	*21*	0.0466
2	0.0337	*12*	0.0388	*22*	0.0472
3	0.0341	*13*	0.0397	*23*	0.0477
4	0.0346	*14*	0.0409	*24*	0.0468
5	0.0350	*15*	0.0422		
6	0.0355	*16*	0.0435		
7	0.0360	*17*	0.0449		
8	0.0365	*18*	0.0455		
9	0.0370	*19*	0.0458		

Table 3. The learned importance weights of all hidden states in the MLLM with Qwen-2.5 3B model, including the input embedding (indexed as 0). The ***Index*** and **Weight** column indicates the index and the learned importance weight of each hidden state.

Index	**Weight**	***Index***	**Weight**	***Index***	**Weight**
0	0.0254	*13*	0.0259	*26*	0.0277
1	0.0251	*14*	0.0260	*27*	0.0283
2	0.0251	*15*	0.0261	*28*	0.0287
3	0.0252	*16*	0.0262	*29*	0.0294
4	0.0253	*17*	0.0263	*30*	0.0297
5	0.0252	*18*	0.0264	*31*	0.0303
6	0.0253	*19*	0.0265	*32*	0.0304
7	0.0254	*20*	0.0266	*33*	0.0304
8	0.0255	*21*	0.0271	*34*	0.0302
9	0.0255	*22*	0.0265	*35*	0.0303
10	0.0256	*23*	0.0251	*36*	0.0331
11	0.0257	*24*	0.0258		
12	0.0258	*25*	0.0267		

Table 4. 3D occupancy performance on the Occ3D-nuScenes validation set. * indicates that DrivePI is trained exclusively on the 3D occupancy task of Occ3D-nuScenes.

Method	VLM-based	RayIoU↑	RayIoU_{1m}	RayIoU_{2m}	RayIoU_{4m}
RenderOcc [9]		19.5	13.4	19.6	25.5
SimpleOcc [3]		22.5	17.0	22.7	27.9
BEVFormer [5]		32.4	26.1	32.9	38.0
BEVDet-Occ [4]		32.6	26.6	33.1	38.2
FB-Occ [6]		33.5	26.7	34.1	39.7
SparseOcc [8]		36.1	30.2	36.8	41.2
OPUS [12]		41.2	34.7	42.1	46.7
DrivePI* (Ours)	✓	**46.0**	**42.2**	**46.7**	**49.2**

Table 5. An example generated by our multi-stage data engine.

Prompt 1: Front Caption

The front scene depicts a street intersection with several notable elements. On the left side of the front scene, there is a modern building with a white facade and large windows. In front of the building, there is a traffic light displaying a red signal. A parked vehicle is visible near the building, and there is a grassy area with a few trees and a sidewalk. In the center of the front scene...<omitted>

Prompt 2: Back Caption

The back scene depicts a street intersection with various elements captured from different angles. On the left side of the back scene, there is a traffic light and a construction area with a fence and some equipment, including a small vehicle or machinery. Two traffic cones are placed near the fence, indicating a possible work zone or restricted area. The background features a multi-story building with visible structural elements...<omitted>

Prompt 3: Rewriting Caption

The scene depicts a street intersection with various vehicles and elements. There are multiple vehicles, including trucks and cars, some of which are moving and others parked. Traffic cones are present in some parts of the scene, indicating possible construction or restricted areas. The area is surrounded by buildings and greenery, with traffic lights controlling the flow of vehicles. There are no visible pedestrians sitting or lying down. The setting appears to be an urban environment with infrastructure for traffic management.

Prompt 4: 3D Occupancy and Flow prediction

Your task is to predict the 3D occupancy of the scene. Assume you are located at the point $(0, 0, 0)$. The scene area around you (in front, behind, left, and right) is divided into a 200×200 grid, with the bottom-left corner at $(-100, -100)$ and the top-right corner at $(100, 100)$. The height region is divided into 16 bins. We use $< OCC > (x, y, z) < /OCC >$ to represent the point at location (x, y) with a height of z. Assume you are located at the point $< OCC > (100, 100, 0) < /OCC >$. Answer the below question.

Is the position {position} occupied?

What object is occupying position $< OCC > (x, y, z) < /OCC >$? If there is an object, please provide its name and predict the velocity; otherwise, answer 'free'.

Prompt 5: Action Prediction

What is the safe action of the ego car?

Prompt 6: Trajectory Prediction

Predict the future 6-frame trajectory of the ego car in the last.

deeper layers tend to receive larger weights, indicating that higher-level features extracted by deeper transformer layers are more crucial for the effectiveness of our DrivePI.

3D Occupancy on Occ3D. Beyond the unified model DrivePI trained jointly on all tasks, we also train DrivePI exclusively on the 3D occupancy task of Occ3D to enable comprehensive comparisons with existing approaches on the Occ3D benchmark [11]. As shown in Table 4, DrivePI achieves state-of-the-art performance with 46.0% RayIoU, surpassing the previous best method OPUS [12] by a significant margin of 4.8%. Notably, DrivePI is built on a MLLM architecture, which highlights its strong capabilities in fine-grained 3D perception, despite being primarily designed for multi-modal understanding.

2. Details of Coarse-grained Spatial Understanding

Table 5 presents a variety of prompts generated by our multi-stage data engine. The data engine first produces individual captions for both front and back views, which are subsequently merged and refined into a coherent final caption. The corresponding prompts are shown as "**Prompt 1-3**" in Table 5. To enhance DrivePI's comprehensive spatial

understanding capabilities, we design instruction question-answering (QA) pairs covering four core tasks: 3D occupancy prediction, flow prediction, action prediction, and trajectory prediction. For 3D occupancy and flow prediction tasks, we generate questions about the occupancy status, category, and velocity of given 3D locations using occupancy and flow ground truth. The corresponding prompts are as shown in "**Prompt 4: Occupancy and Flow Prediction**" in Table 5. For action prediction, we design prompts to predict subsequent driving actions, as demonstrated in "**Prompt 5: Action Prediction**" in Table 5. For trajectory prediction, we create prompts that guide the model to predict future ego-vehicle trajectories, as shown in "**Prompt 6: Trajectory Prediction**" in Table 5.

3. Details of Fine-grained Spatial Learning

We provide details of task-specific heads, including 3D occupancy, flow, and action diffusion heads for fine-grained spatial learning.

Details of 3D Occupancy Head. For the 3D occupancy head, we mainly follow the previous superior method FlashOcc [13]. Specifically, given the spatial feature map $F_{out} \in \mathbb{R}^{H \times W \times C}$ (refer to the main paper), we perform a reshape operation along the channel dimension to transform F_{out} from a shape of $H \times W \times C$ to $H \times W \times Z \times C'$, where C' and Z represent the channel dimension and depth dimension of the features, respectively. Then, we use an MLP to predict the category of 3D occupancy. For loss functions, we adopt the same losses as in [13], including the focal loss $\mathcal{L}_{focal}$, geometric loss $\mathcal{L}_{geo}$, semantic loss $\mathcal{L}_{sem}$, and Lovász loss $\mathcal{L}_{lovasz}$.

Details of Occupancy Flow Head. We use an additional MLP to predict the velocity of 3D occupancy based on the occupancy head. To address the imbalance in the distribution of static and moving objects, we employ L1 loss and apply distinct loss weights to the flow prediction task for static and dynamic occupancy, respectively. Specifically, we assign a weight of 1.0 to each dynamic occupancy flow for computing the flow loss, while applying a smaller weight of 0.01 to static occupancy flow during training.

Details of Action Diffusion Head. We refer to previous action diffusion methods [2, 7] to implement a simple action diffusion head that predicts trajectories based on the denoising procedure. On nuScenes [1], DiffusionDrive [7] applies SparseDrive [10] to achieve plan query initialization for better trajectory planning results. In contrast, our approach directly generates trajectories without requiring SparseDrive for plan query initialization. For the loss of action diffusion head, we employ L1 loss for the action diffusion head during training.

4. More Visualization

In this section, we provide additional visualizations to demonstrate the coarse-grained and fine-grained understanding capabilities of DrivePI. As shown in Table 6, DrivePI generates accurate 3D occupancy and flow predictions in complex congested environments, and successfully implements appropriate actions for stationary waiting scenarios. Furthermore, we visualize a straight-ahead driving scene in Table 7, where DrivePI demonstrates high consistency between coarse-grained and fine-grained understanding. We also present a challenging nighttime turning scenario in Table 8, which is typically difficult for VLA models. In this scene, leveraging multi-modal information, DrivePI accurately describes the environment despite the very dark imagery. Therefore, our DrivePI delivers accurate and reasonable results in this challenging low-light condition, demonstrating its robustness and effectiveness.

References

[1] 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 *CVPR*, 2020. 3

[2] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44 (10-11):1684–1704, 2025. 3

[3] Wanshui Gan, Ningkai Mo, Hongbin Xu, and Naoto Yokoya. A simple framework for 3d occupancy estimation in autonomous driving. *arXiv preprint arXiv:2303.10076*, 2023. 1

[4] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 1

[5] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *ECCV*, 2022. 1

[6] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023. 1

[7] Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, and Xinggang Wang. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *CVPR*, 2025. 3

[8] Haisong Liu, Yang Chen, Haiguang Wang, Zetong Yang, Tianyu Li, Jia Zeng, Li Chen, Hongyang Li, and Limin Wang. Fully sparse 3d occupancy prediction. In *ECCV*, 2024. 1

[9] Mingjie Pan, Jiaming Liu, Renrui Zhang, Peixiang Huang, Xiaoqi Li, Hongwei Xie, Bing Wang, Li Liu, and Shanghang

Table 6. The visualization of remaining stationary with coarse-grained and fine-grained results.

Coarse-grained Perception, Prediction, and Planning

Task 1: Caption

The scene depicts a quiet residential street lined with houses and lush greenery. The street is mostly empty, with a few parked vehicles visible in some parts. There are trees providing shade along the sidewalks, and the houses have gated entrances and fences. A bicycle is parked on the sidewalk near one of the houses. The overall atmosphere is calm and suburban, with no pedestrians or moving vehicles present.

Task 2: Occupancy and Flow prediction

Question: What can be seen at position $< OCC > (83, 99, 5) < /OCC >$? If there is an object, please provide its name and predict the velocity; otherwise, answer 'free'

Answer: {label: car}, {vx: 0.1, vy: 0.06}

Task 3: Action decision

Question: What is the safe action of the ego car?

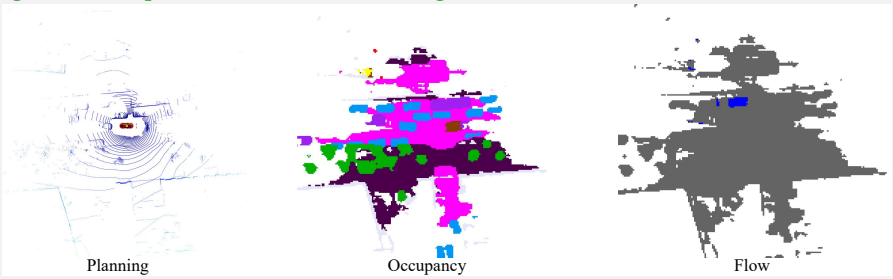
Answer: Go straight.

Task 4: Trajectory prediction

Question: Predict the future 6-frame trajectory of the ego car in the last.

Answer: Future 6-frame trajectory: [(0.12, 0.00), (0.05, 0.00), (-0.01, 0.00), (0.00, 0.00), (0.00, 0.00), (0.00, 0.00)].

Fine-grained Perception, Prediction, and Planning



Zhang. Renderocc: Vision-centric 3d occupancy prediction with 2d rendering supervision. In *ICRA*, 2024. 1

[10] Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. In *ICRA*, 2025. 3

[11] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. In *NeurIPS*, 2023. 2

[12] Jiabao Wang, Zhaojiang Liu, Qiang Meng, Liujiang Yan, Ke Wang, Jie Yang, Wei Liu, Qibin Hou, and Ming-Ming Cheng. Opus: occupancy prediction using a sparse set. In *NeurIPS*, 2024. 1, 2

[13] Zichen Yu, Changyong Shu, Jiajun Deng, Kangjie Lu, Liu. Zongdai, Jiangyong Yu, Dawei Yang, Hui Li, and Yan Chen. Flashocc: Fast and memory-efficient occupancy prediction via channel-to-height plugin. *arXiv preprint arXiv:2311.12058*, 2023. 3

Table 7. The visualization of going straight with coarse-grained and fine-grained results.

Coarse-grained Perception, Prediction, and Planning

Task 1: Caption

The scene depicts a busy urban street with several vehicles, both parked and moving. There are multiple cars, some stopped at intersections and others in motion. Pedestrians are visible walking along the sidewalks. Construction cranes and buildings under construction are present in the background, indicating ongoing development. Traffic cones are placed on the road, likely for directing traffic or marking construction areas. The environment appears to be a bustling city area with a mix of modern architecture and active traffic.

Task 2: Occupancy and Flow prediction

Question: What can be seen at position $< OCC > (159, 78, 4) < /OCC >$? If there is an object, please provide its name and predict the velocity; otherwise, answer 'free'

Answer: {label: car}, {vx: 0.1, vy: -0.07}

Task 3: Action decision

Question: What is the safe action of the ego car?

Answer: Go straight.

Task 4: Trajectory prediction

Question: Predict the future 6-frame trajectory of the ego car in the last.

Answer: Future 6-frame trajectory: [(3.42, -0.03), (3.52, -0.02), (3.59, -0.04), (3.72, -0.07), (3.87, -0.10), (3.88, -0.09)].

Fine-grained Perception, Prediction, and Planning

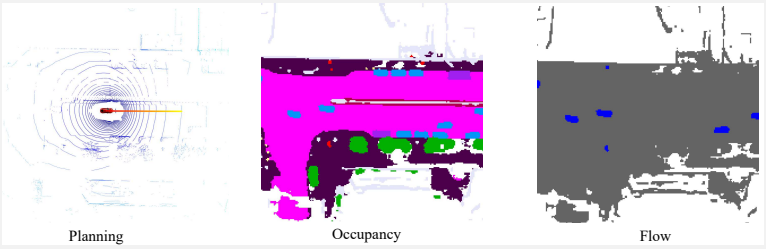
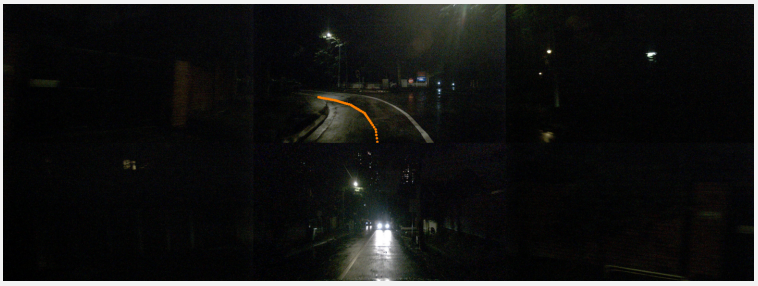


Table 8. The visualization of turning with coarse-grained and fine-grained results.



Coarse-grained Perception, Prediction, and Planning

Task 1: Caption

The scene depicts a quiet residential street lined with houses and lush greenery. The road is paved and has a few parked vehicles, including a white van and a car. There are trees and plants along the sidewalks, and some trash bins are visible near the houses. The area appears calm and suburban, with no pedestrians or moving vehicles in sight.

Task 2: Occupancy and Flow prediction

Question: What can be seen at position $< OCC > (173, 102, 8) < /OCC >$? If there is an object, please provide its name and predict the velocity; otherwise, answer 'free'

Answer: {label: vegetation}, {vx: 0.0, vy: 0.0}

Task 3: Action decision

Question: What is the safe action of the ego car?

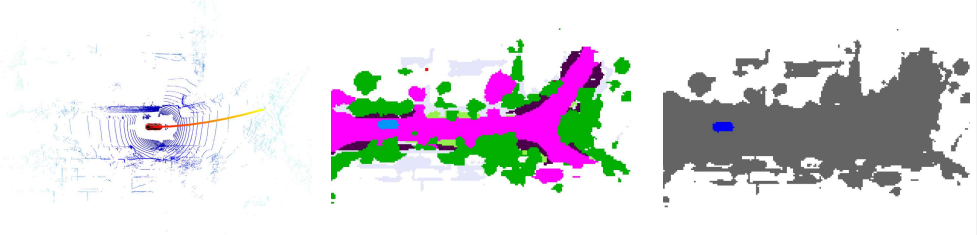
Answer: Turn left.

Task 4: Trajectory prediction

Question: Predict the future 6-frame trajectory of the ego car in the last.

Answer: Future 6-frame trajectory: [(3.55, -0.02), (3.34, -0.25), (3.12, -0.68), (2.98, -1.17), (2.69, -1.74), (2.47, -2.18)].

Fine-grained Perception, Prediction, and Planning



Planning Occupancy Flow