EVODRIVEVLA: EVOLVING DRIVING VLA MODELS VIA COLLABORATIVE PERCEPTION-PLANNING DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language-Action models have shown great promise for autonomous driving, yet they suffer from degraded perception after unfreezing the visual encoder and struggle with accumulated instability in long-term planning. To address these challenges, we propose **EvoDriveVLA**—a novel collaborative perception-planning distillation framework that integrates self-anchored perceptual constraints and future-informed trajectory optimization. Specifically, self-anchored visual distillation leverages self-anchor teacher to deliver visual anchoring constraints, regularizing student representations via trajectory-guided key-region awareness. In parallel, future-informed trajectory distillation employs a future-aware oracle teacher with coarse-to-fine trajectory refinement and Monte Carlo dropout sampling to synthesize reasoning trajectories that model future evolutions, enabling the student model to internalize the future-aware insights of the teacher. EvoDriveVLA achieves SOTA performance in nuScenes open-loop evaluation and significantly enhances performance in NAVSIM. Code is provided in the supplementary material.

1 INTRODUCTION

With the rapid advances of Vision-Language Models (VLMs) Liu et al. (2023); Bai et al. (2025); Zhang et al. (2025b), increasing attention has been directed towards leveraging VLMs for autonomous driving, giving rise to driving Vision-Language-Action (VLA) models that can directly output driving actions and trajectories. Compared to traditional end-to-end approaches Hu et al. (2023); Jiang et al. (2023); Gao et al. (2025), VLA models are capable of not only predicting trajectories, but also understanding navigation instructions Peng et al. (2025), performing scene-based question answering Sima et al. (2024), and utilizing chain-of-thought reasoning Li et al. (2025c). Their superior generalization and reasoning potential make them the mainstay in autonomous driving. However, during practical training, VLA models suffer from degraded perceptual capabilities after unfreezing the visual encoder, as well as trajectory instability in long-term planning.

As a pivotal technique for boosting the performance of autonomous driving systems, knowledge distillation Hinton et al. (2015) has gained significant traction in recent research. As illustrated in Fig. 1, existing distillation methods can be categorized into single-trajectory distillation and multi-trajectory distillation. Single-trajectory approaches, exemplified by DiMA Hegde et al. (2025), directly supervise the student using trajectories predicted by a teacher model. In contrast, multi-trajectory methods, such as DistillDrive Yu et al. (2025), encourage the teacher to produce diverse trajectory outputs by constructing a planning vocabulary, aiming to enrich planning knowledge in distillation through structured trajectory candidates and alleviate the limited expressiveness and poor scenario adaptability of single-trajectory approaches.

However, existing methods have not considered sufficiently the principled design of knowledge distillation for autonomous driving: (1) The visual encoder, which serves as the core component of scene perception, has not been adequately emphasized or effectively handled during the distillation process in existing training pipelines. (2) When the teacher and student models are trained under identical settings, the teacher offers no substantial advantage in planning capability, and therefore fails to provide more accurate or informative knowledge for distillation. (3) Although existing multi-trajectory distillation methods increase the diversity of teacher-generated trajectories, such

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

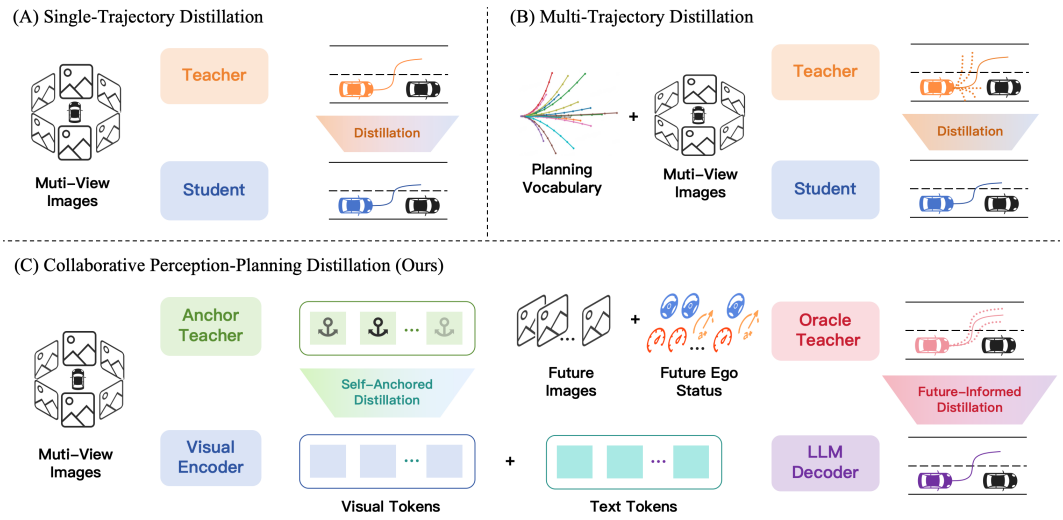


Figure 1: Comparison of distillation methods for autonomous driving. (a) Single-Trajectory Distillation; (b) Multi-Trajectory Distillation; (c) Collaborative Perception-Planning Distillation (Ours).

diversity is largely constrained by predefined planning vocabularies, limiting their ability to truly adapt to the dynamic and context-dependent nature of real-world driving scenarios.

To address the limitations of existing knowledge distillation methods for VLA-based autonomous driving, we propose EvoDriveVLA, a novel collaborative perception-planning distillation framework incorporating self-anchored and future-informed distillation for autonomous driving. As illustrated in Fig. 2, the proposed framework consists of "self-anchored visual distillation" and "future-informed trajectory distillation" to synergistically enhance visual representation and trajectory prediction. Specifically, at the perceptual distillation level, we introduce a self-anchor teacher to provide visual anchoring constraints, preventing the visual encoder from losing its pre-trained representation capabilities after being unfrozen. Simultaneously, trajectory-guided attention is integrated to impose stronger anchoring constraints specifically on critical perceptual regions. At the planning distillation level, we construct a future-aware oracle teacher by incorporating privileged information, including future scene images and ego status, thereby endowing the teacher with superior trajectory prediction accuracy. We further employ a coarse-to-fine trajectory refinement strategy combined with Monte Carlo dropout (MC-Dropout) sampling to synthesize reasoning trajectories that model future evolutions. Subsequently, the optimal trajectory is selected as a soft target for distillation. This enables the student to internalize the oracle's future-predictive capabilities for more robust spatial reasoning and motion prediction. Experimental results demonstrate consistent improvements over matched base models in both open-loop nuScenes Caesar et al. (2020) and closed-loop NAVSIM Dauner et al. (2024) evaluations. Our contributions are summarized as follows:

- We propose EvoDriveVLA, a novel collaborative perception-planning distillation framework with self-anchored and future-informed distillation for driving.
- We introduce self-anchored visual distillation, imposing visual anchoring constraints on trajectory-guided key regions to enhance perceptual capabilities.
- We propose future-informed trajectory distillation, leveraging an oracle teacher to generate high-quality candidates via trajectory refinement and MC-Dropout, enabling the student to internalize the teacher's future-aware insights.
- Our proposed method achieves SOTA performance in open-loop evaluation and significantly enhances performance in closed-loop evaluation.

2 Related Work

2.1 End-to-End Autonomous Driving

Represented by works such as UniAD Hu et al. (2023), end-to-end methods establish a unified mapping framework from perception to planning, significantly improving the overall adaptability and performance of the system. Furthermore, VAD Jiang et al. (2023) and VADv2 Chen et al. (2024) enhance the generalization capability of perception and planning through large-scale visual pre-training, while RAD Gao et al. (2025) builds high-fidelity simulation environments based on 3D Gaussian Splatting Kerbl et al. (2023); Wei et al. (2025); Huang et al. (2024); Wei et al. (2026) for closed-loop optimization. With the rise of generative models, approaches such as DiffusionDrive Liao et al. (2025a) leverage diffusion models to generate high-quality multimodal trajectories, and DiffusionDrivev2 Zou et al. (2025) further integrates reinforcement learning to improve prediction diversity and safety in complex scenarios. Benefiting from the formidable and emergent capabilities of large visual language models (VLMs), Vision–Language–Action (VLA) models Chi et al. (2025); Cao et al. (2025a) have increasingly emerged as a promising paradigm in end-to-end autonomous driving. Early works Sima et al. (2024); Tian et al. (2024) pioneered the use of VLMs for scene understanding via question answering and trajectory planning. Subsequently, OmniDrive Wang et al. (2025) and OpenDriveVLA Zhou et al. (2025) further incorporated reasoning tasks and 3D perception modules to improve the accuracy of trajectory prediction and the completeness of environmental modeling.

2.2 Distilling Knowledge for Autonomous Driving

The success of knowledge distillation in VLMs Cai et al. (2025); Cao et al. (2025b) is gradually being extended to autonomous driving. Early works Li et al. (2024a; 2025a) focused on traditional end-to-end models, distilling prior knowledge including traffic rules and safety constraints to improve the accuracy and safety of trajectory prediction. DSDrive Liu et al. (2025) distills the trajectory outputs of VLMs teacher as soft targets to improve prediction quality, while DistillDrive Yu et al. (2025) further introduces a planning vocabulary to diversify teacher-generated trajectories. Furthermore, some studies Khanzada & Kwon (2025) distill dense-reward dynamics from teacher world models into reward policies for efficient reinforcement learning, while others enhance robustness via cross-modal probabilistic distillation Liao et al. (2025b) or bridge semantic planning and scene understanding through fine-grained feature distillation in diffusion-based planners Zhang et al. (2025a).

3 Methodology

3.1 Preliminary

In autonomous driving, Vision-Language-Action (VLA) models formulate trajectory planning as the prediction of future waypoint sequences conditioned on multi-modal observations. At each time step t, the model receives a set of multi-view camera images $\mathcal{I}_t = \{I_t^{(v)}\}_{v=1}^{V}$, a textual instruction prompt P_t, and the ego-vehicle state $S_t = (x_t, y_t, v_t, a_t, \delta_t)$, which includes vehicle position, velocity, acceleration, and steering angle. The model outputs a sequence of future waypoints $W_t = \{w_{t+\tau}\}_{\tau=1}^{T}$, where each $w_{t+\tau} = (x_{t+\tau}, y_{t+\tau})$ represents the position at future step $t + \tau$.

We treat waypoints W_t as actions and observations as $\mathcal{O}_t = (\mathcal{I}_t, P_t, S_t)$. Our goal is to model the conditional distribution of future actions given these observations. Specifically, we aim to learn a policy p_θ that captures the joint dependencies among the vision, language, and action modalities. During training, we optimize the negative log-likelihood of the predicted distribution with respect to the ground-truth waypoint sequence $W_t^* = \{w_{t+\tau}^*\}_{\tau=1}^{T}$, which is defined as the training loss:

$$\mathcal{L} = -\sum_{\tau=1}^{T} \log p_\theta\big(w_{t+\tau} = w_{t+\tau}^* \mid \mathcal{O}_t, w_{<t+\tau-1}^*\big), \tag{1}$$

where θ denotes the learnable model parameters, and $w_{<t+\tau}^* = \{w_{t+1}^*, \ldots, w_{t+\tau-1}^*\}$.

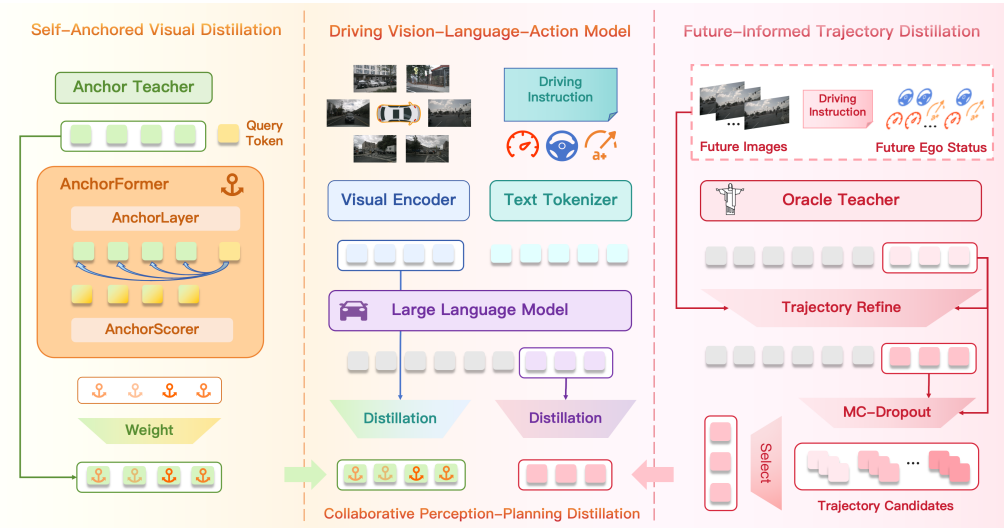


Figure 2: Overview of the EvoDriveVLA framework. **(Left)** Self-anchored visual distillation imposes token-leve visual anchoring constraints across the scene; **(Right)** Future-informed trajectory distillation leverages future ground-truth information for trajectory refinement and diversity sampling; **(Middle)** Collaborative perception-planning distillation enhances autonomous driving VLA model capabilities in both perception and planning to achieve superior driving performance.

3.2 Self-Anchored Visual Distillation

Tajectory-Guided Anchoring Constraints. To address the degradation-adaptation dilemma of visual encoders during supervised fine-tuning, we propose a self-anchored visual distillation. Specifically, we create a self-anchor teacher by copying the student visual encoder before fine-tuning. During training, the stable visual representations produced by this self-anchor teacher are used as distillation constraints, ensuring that the student visual encoder enhances its perception capability for autonomous driving scenarios while preserving its original visual representation power. We further improve granularity by introducing trajectory-guided token-level anchored distillation. To this end, we design AnchorFormer, which assigns adaptive anchor weights to different spatial regions in the scene, where higher weights correlate with intensified anchoring constraints for those regions.

AnchorFormer Architecture. AnchorFormer consists of an AnchorLayer and an AnchorScorer. The AnchorLayer shares the same architecture as a single LLM decoder layer, while the AnchorScorer is implemented as a single linear layer. Given multi-view images I_t, the self-anchor teacher and student visual encoders produce visual tokens $\mathbf{z}_v^{tea}$ and $\mathbf{z}_v^{stu}$, respectively. The textual instruction prompt P_t, ego-vehicle state S_t, and ground-truth future waypoints W_t^* are encoded into token representations $\mathbf{z}_p$, $\mathbf{z}_s$, and $\mathbf{z}_{w^*}$. To enable the self-anchor teacher to assign adaptive anchoring weights to visual tokens conditioned on the instruction, ego state, and future trajectory, we introduce a set of learnable query tokens $\mathbf{q}$. These tokens are concatenated with the observation tokens $\mathbf{z}_o = [\mathbf{z}_v^{tea}, \mathbf{z}_p, \mathbf{z}_s]$ and the trajectory tokens $\mathbf{z}_{w^*}$, and then fed into the AnchorLayer:

$$\tilde{\mathbf{z}}_o,\ \tilde{\mathbf{z}}_{w^*},\ \tilde{\mathbf{q}} = \text{AnchorLayer}\left(\mathbf{z}_o, \mathbf{z}_{w^*}, \mathbf{q}\right), \tag{2}$$

where $\tilde{\mathbf{z}}_o = (\tilde{\mathbf{z}}_v^t, \tilde{\mathbf{z}}_p, \tilde{\mathbf{z}}_s)$. We compute token-level anchor scores by applying the AnchorScorer to the Hadamard product between the updated visual tokens $\tilde{\mathbf{z}}_v^t$ and query tokens $\tilde{\mathbf{q}}$:

$$\mathbf{S}_a = \text{AnchorScorer}\left(\tilde{\mathbf{z}}_v^t \odot \tilde{\mathbf{q}}\right). \tag{3}$$

The anchor scores are first mapped to positive values and then normalized across all visual tokens:

$$\widetilde{W}_a^{(i)} = \sigma\left(S_a^{(i)}/\tau_v\right), \qquad W_a^{(i)} = \frac{N_v \widetilde{W}_a^{(i)}}{\sum_{j=1}^{N_v} \widetilde{W}_a^{(j)}}, \tag{4}$$

where $\tau_v = 2.0$ is the temperature. This normalization satisfies $\frac{1}{N_v}\sum_i W_a^{(i)} = 1$. Therefore, AnchorFormer can only redistribute the visual-distillation strength across spatial tokens rather than trivially minimizing the objective by uniformly suppressing all anchor weights.

Visual Distillation Loss. We adopt a mean squared error (MSE) loss to constrain the student's visual tokens $\mathbf{z}_v^{stu}$ with the self-anchor teacher's visual tokens $\mathbf{z}_v^{tea}$, weighted by the token-level anchor weights $\mathbf{W}_a$. The resulting self-anchored distillation loss $\mathcal{L}_a$ is defined as:

$$\mathcal{L}_a = \frac{1}{N_v} \sum_{i=1}^{N_v} \mathbf{W}_a^{(i)} \left\| \mathbf{z}_v^{tea(i)} - \mathbf{z}_v^{stu(i)} \right\|_2^2, \tag{5}$$

where N_v denotes the number of visual tokens.

3.3 Future-Informed Trajectory Distillation

The Future-Aware Oracle Teacher. We contend that the bottleneck of trajectory prediction in autonomous driving lies in the model's capacity to anticipate complex scene evolutions and potential agent interactions. Existing methods Liu et al. (2025); Hegde et al. (2025) typically employ teachers that share the same observation space as the student. Consequently, these teachers fail to provide insights beyond the student's own perceptual limits, offering little more than saturated guidance. To transcend this limitation, we introduce an oracle teacher with access to privileged future information.

By conditioning on observations from the subsequent T seconds ($\mathcal{O}_{t+1}, \ldots, \mathcal{O}_{t+T-1}$), the teacher can capture the actual spatio-temporal evolution of the scene, establishing a stronger temporal upper bound that surpasses the student's perceptual horizon. The essence of our future-informed trajectory distillation lies in enabling the student model to internalize the future-aware insights encapsulated within the oracle teacher. Crucially, this mechanism preserves a fair evaluation setting, as the student remains strictly confined to current observations during both training and inference. To further bolster the teacher's predictive quality, we implement a progressive coarse-to-fine refinement strategy:

$$\begin{aligned} p_\theta(W_t^c \mid \cdot) &= \prod_{\tau=1}^{T} p_\theta\big(w_{t+\tau} \mid \mathcal{O}_{<t+T},\, w^*_{<t+\tau}\big)\,, \\ p_\theta(W_t^f \mid \cdot) &= \prod_{\tau=1}^{T} p_\theta\big(w_{t+\tau} \mid \mathcal{O}_{<t+T},\, W_t^c,\, w^*_{<t+\tau}\big)\,, \end{aligned} \tag{6}$$

where W_t^c and W_t^f denote coarse and refined trajectory sequences, respectively. During training, a joint sampling-based strategy is employed to optimize both schemes, empowering the teacher model to simultaneously capture both coarse and fine-grained trajectory representations.

Coarse-to-Fine Trajectory Refinement. We feed the oracle teacher's coarse trajectories back into the model for iterative refinement. Leveraging future-aware perception, the oracle teacher rectifies candidate trajectories for spatio-temporal consistency, yielding smoother and physically plausible optimized paths. This recursive generate-refine process effectively simulates the progressive trajectory evolution under oracle guidance. To help the student inherit these future-informed priors, we include hidden states and logits from both coarse and fine-grained trajectories in the candidate sets $\mathcal{S}_h = \{\mathbf{h}_\mathrm{c}, \mathbf{h}_\mathrm{f}\}$ and $\mathcal{S}_l = \{\mathbf{l}_\mathrm{c}, \mathbf{l}_\mathrm{f}\}$, respectively. This strategy allows the student to effectively internalize the nuanced corrective capabilities and future-oriented insights of the oracle teacher.

MC-Dropout Trajectory Sampling. Although the coarse-to-fine strategy enables the oracle teacher to rectify its predictions, we further introduce a Monte Carlo Dropout (MC-Dropout) trajectory sampling strategy to enhance both the precision and robustness of the supervision signal. Instead of relying on a single deterministic forward pass, we apply N stochastic dropout perturbations to the oracle teacher's output hidden state $\mathbf{h} \in \mathcal{S}_h$ while keeping parameters fixed, resulting in a set of diversified hidden state samples:

$$\mathbf{h}^{(n)} = \mathrm{Dropout}(\mathbf{h};\, p)\,, \quad n = 1, \ldots, N, \tag{7}$$

where $p = 0.1$ denotes the dropout rate and $N = 10$. The sampled hidden states are then fed into the model's $\mathbf{lm_head}$ to obtain the corresponding logits. Finally, all sampled hidden states and their associated logits are incorporated into the candidate sets $\mathcal{S}_h$ and $\mathcal{S}_l$, respectively:

Table 1: Open-loop evaluation on nuScenes. We conduct open-loop evaluation on the nuScenes, comparing our method against traditional, LLM-based, and distillation-based baselines.

Method	ST-P3								UniAD							
	L2 (m) ↓				Collision (%) ↓				L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Traditional																
ST-P3 [ECCV 22]	1.44	2.11	2.90	2.11	0.23	0.62	1.27	0.71	-	-	-	-	-	-	-	-
VAD [ICCV 23]	0.17	0.34	0.60	0.37	0.04	0.27	0.67	0.33	-	-	-	-	-	-	-	-
BEV-Planner [CVPR 24]	0.16	0.32	0.57	0.35	0.00	0.29	0.73	0.34	-	-	-	-	-	-	-	-
DiffusionDrive [CVPR 25]	0.27	0.54	0.90	0.57	0.03	0.05	0.16	0.08	-	-	-	-	-	-	-	-
FF [CVPR 21]	-	-	-	-	-	-	-	-	0.55	1.20	2.54	1.43	0.06	0.17	1.07	0.43
EO [ECCV 22]	-	-	-	-	-	-	-	-	0.67	1.36	2.78	1.60	0.04	0.09	0.88	0.33
PreWorld [ICLR 25]	-	-	-	-	-	-	-	-	0.49	1.22	2.32	1.34	0.19	0.57	2.65	1.14
UniAD [CVPR 23]	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
LLM-Based																
DriveVLM [CoRL 24]	0.18	0.34	0.68	0.40	0.10	0.22	0.45	0.27	-	-	-	-	-	-	-	-
OmniDrive [CVPR 25]	0.14	0.29	0.55	0.33	0.00	0.13	0.78	0.30	-	-	-	-	-	-	-	-
ORION [ICCV 25]	0.17	0.31	0.55	0.34	0.05	0.25	0.80	0.37	-	-	-	-	-	-	-	-
ELM [ECCV 24]	-	-	-	-	-	-	-	-	0.34	1.23	2.57	1.38	0.12	0.50	2.36	0.99
DME-Driver [AAAI 25]	-	-	-	-	-	-	-	-	0.45	0.91	1.58	0.98	0.05	0.28	0.55	0.29
GPT-Driver [NeurIPS 23]	0.20	0.40	0.70	0.44	0.04	0.12	0.36	0.17	0.27	0.74	1.52	0.84	0.07	0.15	1.10	0.44
OccWorld [ECCV 24]	0.39	0.73	1.18	0.77	0.11	0.19	0.67	0.32	0.52	1.27	2.41	1.40	0.12	0.40	2.08	0.87
OpenDriveVLA [AAAI 26]	0.14	0.30	0.55	0.33	0.02	0.07	0.22	0.10	0.19	0.58	1.24	0.67	0.02	0.18	0.70	0.30
Distillation-Based																
DistillDrive [ICCV 25]	0.28	0.54	0.83	0.57	0.00	0.03	0.17	0.06	-	-	-	-	-	-	-	-
DiMA [CVPR 25]	0.12	0.25	0.44	0.27	0.04	0.06	0.15	0.08	0.18	0.48	1.01	0.57	0.00	0.05	0.16	0.07
EvoDriveVLA(Ours)	**0.12**	**0.24**	**0.43**	**0.26**	0.02	0.05	**0.12**	**0.06**	**0.16**	**0.44**	**0.96**	**0.52**	0.02	**0.02**	0.33	0.12

$$\mathcal{S}_h \leftarrow \mathcal{S}_h \cup \left\{\mathbf{h}^{(n)}\right\}_{n=1}^{N}, \quad \mathcal{S}_l \leftarrow \mathcal{S}_l \cup \left\{\mathbf{l}^{(n)} = \text{lm_head}\left(\mathbf{h}^{(n)}\right)\right\}_{n=1}^{N}. \tag{8}$$

We apply the $N = 10$ perturbations independently to both the coarse and refined hidden states. Together with the two deterministic trajectories, this produces $2 + 2N = 22$ teacher candidates. Since MC-Dropout is applied only to hidden states and logits are computed through the lightweight $\mathbf{lm_head}$, this strategy enables the oracle teacher to identify and select reliable trajectory paths from an expanded search space.

Trajectory Distillation Loss. We compute the cross-entropy loss between predicted logits in $\mathcal{S}_l$ and the ground-truth, and select the optimal trajectory with the minimum loss:

$$\hat{k} = \arg\min_{\mathbf{l}_k \in \mathcal{S}_l} \mathcal{L}_{\text{CE}}(\mathbf{l}_k, \mathbf{W}^*). \tag{9}$$

The truth trajectory is used only during training to select the teacher candidate, and never provided to the student and is unavailable during inference. Each decoded candidate remains paired with the hidden-state sequence and logits at its trajectory-token positions. We then distill the student model using the paired hidden states and logits associated with the selected future-informed trajectory as soft targets. This encourages the student to align with the teacher's privileged knowledge in both the latent representation space and the predictive distribution. The formulation is given as follows:

$$\mathcal{L}_{\text{h}} = \frac{1}{N_t}\sum_{i=1}^{N_t}\left\|\mathbf{h}_{stu}^{(i)} - \mathbf{h}_{\hat{k}}^{(i)}\right\|_2^2, \quad \mathcal{L}_{\text{l}} = \text{KL}\left(\text{softmax}(\mathbf{l}_{\hat{k}}/\tau_t) \parallel \text{softmax}(\mathbf{l}_{stu}/\tau_t)\right) \tag{10}$$

where $\tau_t = 5$, N_t denotes the number of trajectory tokens, $\mathbf{h}_{stu}$ and $\mathbf{l}_{stu}$ are student's hidden states and logist, respectively. Essentially, this dual-level alignment enables the student to internalize the reasoning required for trajectory prediction.

3.4 Overall Training Loss

The overall training loss $\mathcal{L}_{all}$ of the student model is a weighted combination composed of the trajectory prediction loss $\mathcal{L}$, the self-anchored visual distillation loss $\mathcal{L}_a$, and the future-informed trajectory distillation loss terms $\mathcal{L}_{\text{h}}$ and $\mathcal{L}_{\text{l}}$, which can be formulated as follows:

Table 2: Performance comparison on NAVSIM *navtest* using closed-loop metrics. The Qwen2.5-VL rows provide matched backbone comparisons.

Method	NC↑	DAC↑	TTC↑	Comf. ↑	EP↑	PDMS↑
Constant Velocity	68.0	57.8	50.0	100	19.4	20.6
Ego Status MLP	93.0	77.3	83.6	100	62.8	65.6
VADv2-$\mathcal{V}_{8192}$ [ICCV 23]	97.2	89.1	91.6	100	76.0	80.9
UniAD [CVPR 23]	97.8	91.9	92.9	100	78.8	83.4
TransFuser [TPAMI 23]	97.7	92.8	92.8	100	79.2	84.0
PARA-Drive [CVPR 24]	97.9	92.4	93.0	99.8	79.3	84.0
Qwen2.5-VL-3B [arXiv 25]	97.3	90.4	92.9	99.6	77.6	81.9
EvoDriveVLA (3B)	**98.0**	**93.3**	**93.1**	**100.0**	**81.1**	**85.3**
Qwen2.5-VL-8B [arXiv 25]	97.8	92.1	92.8	100.0	78.3	83.3
InternVL3-8B [arXiv 25]	97.0	92.4	91.8	100.0	78.9	83.3
EvoDriveVLA (8B)	**98.9**	**94.0**	**93.3**	**100.0**	**82.9**	**86.7**

Table 3: Ablation study on algorithmic components.

Ablation					L2 (m) ↓				Collision (%) ↓			
Traj KD	Refine	MC-Drop	Frozen Enc	Visual KD	1s	2s	3s	Avg.	1s	2s	3s	Avg.
×	×	×	×	×	0.17	0.47	1.02	0.55	0.03	0.08	0.45	0.19
✓	×	×	×	×	0.17	0.46	1.00	0.54	0.05	0.07	0.38	0.17
✓	✓	×	×	×	0.16	0.46	0.99	0.53	0.05	0.03	0.40	0.16
✓	✓	✓	×	×	0.16	0.45	0.98	0.53	0.03	0.05	0.35	0.14
✓	✓	✓	✓	×	0.17	0.47	1.04	0.56	0.03	0.10	0.42	0.18
✓	✓	✓	×	✓	**0.16**	**0.44**	**0.96**	**0.52**	**0.02**	**0.02**	**0.33**	**0.12**

$$\mathcal{L}_{all} = \mathcal{L} + \lambda_a * \mathcal{L}_a + \lambda_h * \mathcal{L}_h + \lambda_l * \mathcal{L}_l, \tag{11}$$

where we set $\lambda_a = 0.05$, $\lambda_h = 0.1$, and $\lambda_l = 0.2$.

4 Experiments

4.1 Experimental Settings

Implementation Details. Both the student model and the oracle teacher share the same Qwen2.5-VL 3B Bai et al. (2025) architecture, while the corresponding visual encoder serves as the self-anchor teacher. Furthermore, the AnchorLayer is initialized with the weights from its final LLM layer. During the distillation , the weights of both the oracle teacher and the self-anchor teacher remain frozen, while only the parameters of the student model and the AnchorFormer are actively optimized.

Datasets and Evaluations. We evaluate open-loop performance on nuScenes Caesar et al. (2020) following ST-P3 Hu et al. (2022) and UniAD Hu et al. (2023) protocols, measuring L2 errors and collision rates. For closed-loop assessment, we use the NAVSIM Dauner et al. (2024) benchmark. We report the PDM-Score (PDMS), a composite metric including No Collision (NC), Drivable Area Compliance (DAC), Time to Collision (TTC), Comfort (Comf.), and Ego Progress (EP).

4.2 Open-loop Evaluation

We evaluate the open-loop trajectory planning performance on the nuScenes Caesar et al. (2020) benchmark. Specifically, we compare our method against three categories of baselines: traditional Hu et al. (2022); Jiang et al. (2023); Li et al. (2024b); Liao et al. (2025a); Hu et al. (2021); Khurana et al. (2022); Li et al. (2025b); Hu et al. (2023), LLM-based Tian et al. (2024); Wang et al. (2025); Fu et al. (2025); Zhou et al. (2024); Han et al. (2025); Mao et al. (2023); Zheng et al. (2024); Zhou et al. (2025), and distillation-based Yu et al. (2025); Hegde et al. (2025) approaches. As illustrated in Tab. 1, EvoDriveVLA improves DiMA's average L2 error from 0.27 m to 0.26 m and its average

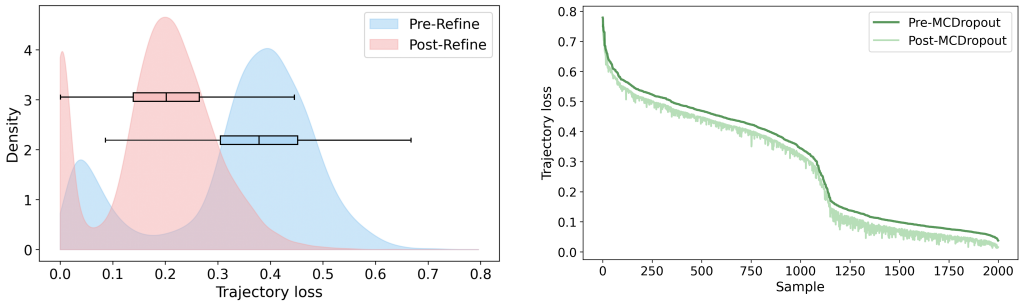


Figure 3: **(Left)**: Kernel density estimation of trajectory loss distributions for pre-refine and post-refine trajectories. The overlaid boxplots summarize the median, interquartile range, and extreme values. **(Right)**: Comparison of trajectory loss distributions before and after MC-Dropout sampling.

Table 4: Oracle teacher performance on nuScenes.

Metric	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
ST-P3	0.10	0.14	0.18	0.14	0.02	0.03	0.05	0.04
UniAD	0.13	0.20	0.27	0.20	0.02	0.05	0.05	0.04

collision rate from 0.08% to 0.06% under ST-P3. Under UniAD, EvoDriveVLA improves average L2 error from 0.57 m to 0.52 m. Compared to OpenDriveVLA, our method improves average L2 error and collision rate by 21% and 40% under ST-P3, and by 22% and 60% under UniAD, respectively.

4.3 Close-loop Evaluation

In the closed-loop evaluation, we compare our approach with camera-only methods Chen et al. (2024); Hu et al. (2023); Chitta et al. (2022); Weng et al. (2024) and matched VLM backbones on NAVSIM. As shown in Tab. 2, EvoDriveVLA improves the Qwen2.5-VL-3B backbone from 81.9 to 85.3 PDMS and the Qwen2.5-VL-8B backbone from 83.3 to 86.7 PDMS, corresponding to gains of 3.4 points for both model scales. These controlled comparisons show that the proposed distillation framework remains effective for a stronger backbone without changing the student inference inputs or procedure.

4.4 Ablation Study

We conduct ablation studies on the nuScenes benchmark using UniAD metrics, with results summarized in Tab. 3. The results show that future-informed trajectory distillation significantly enhances prediction accuracy. This improvement is attributed to the superior trajectory prediction capability of the oracle teacher, as well as the coarse-to-fine refinement and MC-Dropout sampling strategies. As illustrated in Tab. 4, it is evident that the oracle teacher, empowered by future information input, significantly outperforms existing methods in terms of both L2 error and collision rate. Meanwhile, self-anchored visual distillation constrains the student model's visual representations and improves the robustness of downstream trajectory planning under adverse visual conditions. Across three runs, EvoDriveVLA reduces average L2 error from 0.557 ± 0.012 to 0.523 ± 0.006 and average collision rate from 0.177 ± 0.023 to 0.127 ± 0.021.

We statistically analyze the loss distribution between teacher-predicted trajectories and ground truth before and after the coarse-to-fine trajectory refinement, visualized via kernel density estimation (KDE) plots. As illustrated in Fig. 3 (left), the refinement process causes the trajectory loss distribution to shift significantly toward the lower-value region. Notably, the density near zero markedly increases, while the long-tail distribution of outliers is substantially alleviated. These observations demonstrate the effectiveness of coarse-to-fine refinement in enhancing teacher trajectory prediction.

Furthermore, we analyze teacher trajectory loss variation before and after MC-Dropout sampling. As shown in Fig. 3 (right), this strategy further reduces the teacher's prediction error, providing the

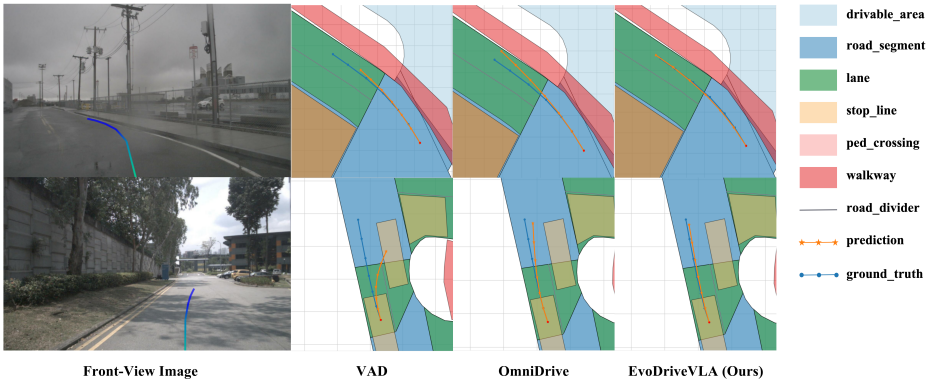


Figure 4: Qualitative comparison on nuScenes.

Table 5: Ablation on MC-Dropout trajectory sampling.

Methods	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
w/o	0.16	0.46	0.99	0.53	0.05	0.03	0.40	0.16
Parameter	0.17	0.46	1.01	0.54	0.07	0.07	0.38	0.17
Hidden State	0.16	0.44	0.97	0.52	0.03	0.03	0.30	0.12

student with more precise trajectory guidance. Notably, the loss in the near-zero region drops by approximately 50%, with nearly 30% of teacher trajectories achieving an L2 loss below 0.1. We also investigate the placement of MC-Dropout, as shown in Tab. 5. Applying MC-Dropout to model parameters degrades performance; however, performing it on hidden states not only boosts accuracy but also enables multi-trajectory sampling with negligible latency.

4.5 Qualitative Results

Fig. 4 presents a qualitative comparison between our method and other baselines on nuScenes. It is evident that our approach significantly outperforms VAD and OmniDrive in long-horizon prediction across diverse weather (sunny/overcast) and road geometries (straight/curved). Specifically, VAD tends to produce short longitudinal predictions, while OmniDrive often exhibits lateral deviations. Additionally, Fig. 5 provides a visualization of the learned visual anchors. The results show that AnchorFormer precisely attends to key elements such as vehicles and roadside pedestrians. This demonstrates the effectiveness of our self-anchored visual distillation in capturing critical environmental features.

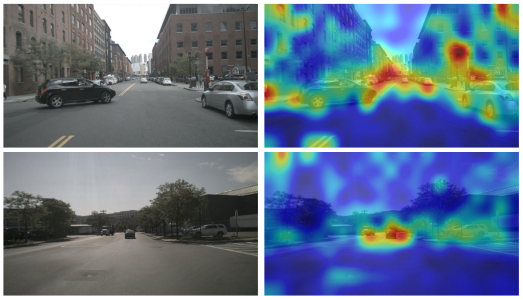


Figure 5: Visualization of the visual anchors.

5 Conclusion

We present EvoDriveVLA, a collaborative perception-planning framework featuring self-anchored and future-informed distillation for autonomous driving. To mitigate visual degradation and trajectory imprecision in current models, we employ self-anchored visual distillation to preserve the encoder's intrinsic perceptual strength. Simultaneously, we leverage an oracle teacher with access to privileged future insights to provide high-fidelity trajectory guidance. Enhanced by coarse-to-fine refinement and MC-Dropout sampling, this approach enables the student to internalize the oracle teacher's future-predictive capabilities. This research establishes a novel and effective distillation paradigm to enhance VLA models for autonomous driving.

AI USE STATEMENT

Generative AI tools were not used to generate synthetic datasets, develop the theoretical or conceptual framework, formulate mathematical claims or proofs, propose hypotheses, design the methodology or experiments, implement the proposed method, process datasets, or interpret experimental results. Generative AI tools were used to improve the readability of the manuscript, check its structure against the ICLR 2027 submission instructions, and assist with LaTeX formatting. All AI-assisted edits were reviewed and verified by the authors. The authors take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work studies learning-based trajectory planning for autonomous driving, a safety-critical application. The proposed method is evaluated only in established research benchmarks and is not intended for direct deployment in real vehicles without extensive validation, safety assurance, and compliance with applicable regulations. The experiments use the public nuScenes and NAVSIM datasets under their respective terms; the authors collected no new human-subject data. As these benchmarks cover limited geographic regions, weather conditions, and traffic conventions, the resulting models may exhibit geographic, environmental, or demographic biases. We report these limitations and encourage evaluation under broader conditions before any practical use. The authors have read and adhered to the ICLR Code of Ethics.

REPRODUCIBILITY STATEMENT

The model architecture, training objectives, and distillation procedures are specified in the Methodology section, while the datasets, evaluation protocols, implementation details, and comparison settings are described in the Experiments section. The appendix provides additional sensitivity and robustness analyses together with further qualitative results. Code and configuration files needed to reproduce the experiments are included in the anonymous supplementary material.

REFERENCES

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.

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*, pp. 11621–11631, 2020.

Yuxuan Cai, Jiangning Zhang, Haoyang He, Xinwei He, Ao Tong, Zhenye Gan, Chengjie Wang, Zhucun Xue, Yong Liu, and Xiang Bai. Llava-kd: A framework of distilling multimodal large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 239–249, 2025.

Jiajun Cao, Qizhe Zhang, Peidong Jia, Xuhui Zhao, Bo Lan, Xiaoan Zhang, Zhuo Li, Xiaobao Wei, Sixiang Chen, Liyun Li, et al. Fastdrivevla: Efficient end-to-end driving via plug-and-play reconstruction-based token pruning. *arXiv preprint arXiv:2507.23318*, 2025a.

Jiajun Cao, Yuan Zhang, Tao Huang, Ming Lu, Qizhe Zhang, Ruichuan An, Ningning Ma, and Shanghang Zhang. Move-kd: Knowledge distillation for vlms with mixture of visual encoders. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 19846–19856, 2025b.

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

Haohan Chi, Huan-ang Gao, Ziming Liu, Jianing Liu, Chenyu Liu, Jinwei Li, Kaisen Yang, Yangcheng Yu, Zeda Wang, Wenyi Li, et al. Impromptu vla: Open weights and open data for driving vision-language-action models. *arXiv preprint arXiv:2505.23757*, 2025.

Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022.

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

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.

Hao Gao, Shaoyu Chen, Bo Jiang, Bencheng Liao, Yiang Shi, Xiaoyang Guo, Yuechuan Pu, Haoran Yin, Xiangyu Li, Xinbang Zhang, et al. Rad: Training an end-to-end driving policy via large-scale 3dgs-based reinforcement learning. *arXiv preprint arXiv:2502.13144*, 2025.

Wencheng Han, Dongqian Guo, Cheng-Zhong Xu, and Jianbing Shen. Dme-driver: Integrating human decision logic and 3d scene perception in autonomous driving. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3347–3355, 2025.

Deepti Hegde, Rajeev Yasarla, Hong Cai, Shizhong Han, Apratim Bhattacharyya, Shweta Mahajan, Litian Liu, Risheek Garrepalli, Vishal M Patel, and Fatih Porikli. Distilling multi-modal large language models for autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27575–27585, 2025.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Peiyun Hu, Aaron Huang, John Dolan, David Held, and Deva Ramanan. Safe local motion planning with self-supervised freespace forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12732–12741, 2021.

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*, pp. 533–549. Springer, 2022.

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*, pp. 17853–17862, 2023.

Nan Huang, Xiaobao Wei, Wenzhao Zheng, Pengju An, Ming Lu, Wei Zhan, Masayoshi Tomizuka, Kurt Keutzer, and Shanghang Zhang. S3gaussian: Self-supervised street gaussians for autonomous driving. *arXiv preprint arXiv:2405.20323*, 2024.

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*, pp. 8340–8350, 2023.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Feeza Khan Khanzada and Jaerock Kwon. Driving beyond privilege: Distilling dense-reward knowledge into sparse-reward policies. *arXiv preprint arXiv:2512.04279*, 2025.

Tarasha Khurana, Peiyun Hu, Achal Dave, Jason Ziglar, David Held, and Deva Ramanan. Differentiable raycasting for self-supervised occupancy forecasting. In *European Conference on Computer Vision*, pp. 353–369. Springer, 2022.

Kailin Li, Zhenxin Li, Shiyi Lan, Yuan Xie, Zhizhong Zhang, Jiayi Liu, Zuxuan Wu, Zhiding Yu, and Jose M Alvarez. Hydra-mdp++: Advancing end-to-end driving via expert-guided hydra-distillation. *arXiv preprint arXiv:2503.12820*, 2025a.

Xiang Li, Pengfei Li, Yupeng Zheng, Wei Sun, Yan Wang, and Yilun Chen. Semi-supervised vision-centric 3d occupancy world model for autonomous driving. *arXiv preprint arXiv:2502.07309*, 2025b.

Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, et al. Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. *arXiv preprint arXiv:2506.08052*, 2025c.

Zhenxin Li, Kailin Li, Shihao Wang, Shiyi Lan, Zhiding Yu, Yishen Ji, Zhiqi Li, Ziyue Zhu, Jan Kautz, Zuxuan Wu, et al. Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. *arXiv preprint arXiv:2406.06978*, 2024a.

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*, pp. 14864–14873, 2024b.

Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12037–12047, 2025a.

Dacheng Liao, Mengshi Qi, Peng Shu, Zhining Zhang, Yuxin Lin, Liang Liu, and Huadong Ma. Robodrivevlm: A novel benchmark and baseline towards robust vision-language models for autonomous driving. *arXiv preprint arXiv:2512.01300*, 2025b.

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

Wenru Liu, Pei Liu, and Jun Ma. Dsdrive: Distilling large language model for lightweight end-to-end autonomous driving with unified reasoning and planning. *arXiv preprint arXiv:2505.05360*, 2025.

Jiageng Mao, Yuxi Qian, Junjie Ye, Hang Zhao, and Yue Wang. Gpt-driver: Learning to drive with gpt. *arXiv preprint arXiv:2310.01415*, 2023.

Qucheng Peng, Chen Bai, Guoxiang Zhang, Bo Xu, Xiaotong Liu, Xiaoyin Zheng, Chen Chen, and Cheng Lu. Navigscene: Bridging local perception and global navigation for beyond-visual-range autonomous driving. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 4193–4202, 2025.

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*, pp. 256–274. Springer, 2024.

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.

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*, pp. 22442–22452, 2025.

Xiaobao Wei, Qingpo Wuwu, Zhongyu Zhao, Zhuangzhe Wu, Nan Huang, Ming Lu, Ningning Ma, and Shanghang Zhang. Emd: Explicit motion modeling for high-quality street gaussian splatting. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 28462–28472, 2025.

Xiaobao Wei, Zhangjie Ye, Yuxiang Gu, Zunjie Zhu, Yunfei Guo, Yingying Shen, Shan Zhao, Ming Lu, Haiyang Sun, Bing Wang, et al. Parkgaussian: Surround-view 3d gaussian splatting for autonomous parking. *arXiv preprint arXiv:2601.01386*, 2026.

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*, pp. 15449–15458, 2024.

Rui Yu, Xianghang Zhang, Runkai Zhao, Huaicheng Yan, and Meng Wang. Distilldrive: End-to-end multi-mode autonomous driving distillation by isomorphic hetero-source planning model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26188–26197, 2025.

Jinhao Zhang, Wenlong Xia, Zhexuan Zhou, Youmin Gong, and Jie Mei. Lap: Fast latent diffusion planner with fine-grained feature distillation for autonomous driving. *arXiv preprint arXiv:2512.00470*, 2025a.

Qizhe Zhang, Aosong Cheng, Ming Lu, Renrui Zhang, Zhiyong Zhuo, Jiajun Cao, Shaobo Guo, Qi She, and Shanghang Zhang. Beyond text-visual attention: Exploiting visual cues for effective token pruning in vlms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 20857–20867, 2025b.

Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. In *European conference on computer vision*, pp. 55–72. Springer, 2024.

Xingcheng Zhou, Xuyuan Han, Feng Yang, Yunpu Ma, and Alois C Knoll. Opendrivevla: Towards end-to-end autonomous driving with large vision language action model. *arXiv preprint arXiv:2503.23463*, 2025.

Yunsong Zhou, Linyan Huang, Qingwen Bu, Jia Zeng, Tianyu Li, Hang Qiu, Hongzi Zhu, Minyi Guo, Yu Qiao, and Hongyang Li. Embodied understanding of driving scenarios. In *European Conference on Computer Vision*, pp. 129–148. Springer, 2024.

Jialv Zou, Shaoyu Chen, Bencheng Liao, Zhiyu Zheng, Yuehao Song, Lefei Zhang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Diffusiondrivev2: Reinforcement learning-constrained truncated diffusion modeling in end-to-end autonomous driving. *arXiv preprint arXiv:2512.07745*, 2025.

A Distillation Weight Factor Sensitivity Analysis

Table 6: Weight factor sensitivity analysis.

Weight Factors	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Traj KD Weights(Temp/Weight)								
6/0.1	0.16	0.47	1.00	0.56	0.05	0.07	0.38	0.17
5/0.1	**0.16**	**0.45**	**0.98**	**0.53**	**0.02**	0.05	0.38	**0.15**
5/0.2	0.17	0.48	1.01	0.55	0.03	0.06	0.42	0.17
4/0.1	0.16	0.46	1.00	0.54	0.03	0.03	0.38	0.15
3/0.1	0.16	0.46	1.01	0.54	0.08	0.10	0.33	0.17
Encoder KD Weights								
0.05	**0.16**	**0.46**	**1.01**	**0.54**	**0.05**	**0.05**	**0.38**	**0.16**
0.1	0.17	0.49	1.04	0.57	0.07	0.07	0.39	0.18
0.2	0.18	0.51	1.08	0.59	0.10	0.05	0.43	0.19

Tab. 6 provides a weight sensitivity analysis for the distillation of both the trajectory and encoder parts, including the analysis of distillation weights and distillation temperature. Our method maintains stable effects within a reasonable range of weight factors.

B Robustness Analysis of the Method under Different Visual Conditions

Table 7: Verification of perceptual robustness

Methods	ST-P3 L2 (m) ↓				UniAD L2 (m) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Rainy/Night	0.12	0.23	0.38	0.24	0.16	0.41	0.76	0.43
Sunny	0.15	0.24	0.40	0.27	0.17	0.36	0.80	0.44
Rainy/Night+KD	0.10	0.21	0.35	0.22	0.13	0.39	0.68	0.40
Sunny+KD	0.15	0.22	0.37	0.25	0.17	0.36	0.78	0.44

Tab. 7 evaluates downstream trajectory planning under challenging visual conditions, including rainy, cloudy, and nighttime scenes. The gains from visual distillation are larger in these adverse conditions, indicating improved planning robustness rather than performance on a standalone perception task.

C Teacher Candidate Generation and Selection

Each trajectory is represented as six (x, y) waypoints at 0.5-second intervals over a three-second horizon. Coordinates are rounded to two decimal places and serialized as the text sequence `[(x1,y1), (x2,y2), ..., (x6,y6)]`, which is tokenized by the native Qwen2.5-VL tokenizer. At inference time, the generated tokens are decoded to text and the six coordinate pairs are parsed as the predicted trajectory. For every candidate, the hidden states and logits at the trajectory-token positions remain paired with its decoded waypoint sequence.

The oracle teacher first generates one coarse trajectory and then conditions on it to generate one refined trajectory. We apply $N = 10$ independent hidden-state dropout perturbations to each deterministic trajectory, giving $2 + 2N = 22$ candidates. Ground truth is used only during training to select the teacher candidate with minimum cross-entropy loss. Neither the student nor the inference procedure accesses ground-truth trajectories or future observations.

Algorithm 1 Teacher Candidate Generation and Selection

Require: Current observations $\mathcal{O}$; future observations $\mathcal{F}$ available only to the teacher; ground-truth trajectory-token sequence $\mathbf{W}^*$; number of stochastic samples $N = 10$
Ensure: Selected teacher hidden states $\mathbf{h}_{\hat{k}}$, logits $\mathbf{l}_{\hat{k}}$, and decoded trajectory $W_{\hat{k}}$

$(\mathbf{h}_c, \mathbf{l}_c, W_c) \leftarrow \mathrm{Teacher}(\mathcal{O}, \mathcal{F})$
$(\mathbf{h}_f, \mathbf{l}_f, W_f) \leftarrow \mathrm{Teacher}(\mathcal{O}, \mathcal{F}, W_c)$
$\mathcal{C} \leftarrow \{(\mathbf{h}_c, \mathbf{l}_c, W_c), (\mathbf{h}_f, \mathbf{l}_f, W_f)\}$
for $s \in \{c, f\}$ **do**
  for $n = 1, \ldots, N$ **do**
    $\hat{\mathbf{h}}_s^{(n)} \leftarrow \mathrm{Dropout}(\mathbf{h}_s; p = 0.1)$
    $\hat{\mathbf{l}}_s^{(n)} \leftarrow \mathrm{LMHead}(\hat{\mathbf{h}}_s^{(n)})$
    $\hat{W}_s^{(n)} \leftarrow \mathrm{Decode}(\hat{\mathbf{l}}_s^{(n)})$
    Add $(\hat{\mathbf{h}}_s^{(n)}, \hat{\mathbf{l}}_s^{(n)}, \hat{W}_s^{(n)})$ to $\mathcal{C}$
  end for
end for
$\hat{k} \leftarrow \arg\min_{k \in \{1,\ldots,22\}} \mathcal{L}_{\mathrm{CE}}(\mathbf{l}_k, \mathbf{W}^*)$
return $(\mathbf{h}_{\hat{k}}, \mathbf{l}_{\hat{k}}, W_{\hat{k}})$ for student distillation

D Controlled Ablations of Privileged Inputs

We separately provide future visual observations and future ego states to the oracle teacher. Future ego states yield a larger improvement than future views, while their combination provides the strongest supervision.

Table 8: Controlled ablation of privileged teacher inputs on nuScenes using UniAD metrics.

Method	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Baseline	0.17	0.47	1.02	0.55	0.03	0.08	0.45	0.19
Only future views	0.17	0.46	1.00	0.55	0.03	0.05	0.42	0.17
Only future egos	0.16	0.45	0.97	0.53	0.03	0.03	0.35	0.14
EvoDriveVLA	**0.16**	**0.44**	**0.96**	**0.52**	**0.02**	**0.02**	**0.33**	**0.12**

E AnchorFormer versus Simpler Alternatives

We compare AnchorFormer with uniform visual-token distillation (No Weights Structure) and weights predicted by a simple MLP. AnchorFormer provides the lowest average L2 error and the lowest three-second collision rate among these visual-distillation variants.

Table 9: Comparison of AnchorFormer with simpler visual-token weighting alternatives.

Method	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Baseline	0.17	0.47	1.02	0.55	0.03	0.08	0.45	0.19
No Weights Structure	0.17	0.48	1.04	0.56	0.02	0.02	0.40	0.15
MLP Weights	0.17	0.47	1.02	0.55	0.02	0.06	0.40	0.16
AnchorFormer	**0.17**	**0.46**	**1.00**	**0.54**	0.03	0.03	**0.38**	**0.15**

F Component-wise Distillation Analysis

The two branches provide complementary benefits: trajectory guidance primarily improves trajectory accuracy, AnchorFormer improves robustness through visual representation constraints, and their combination achieves the best overall result.

Table 10: Individual and combined contributions of trajectory guidance and AnchorFormer.

Method	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
Baseline	0.17	0.47	1.02	0.55	0.03	0.08	0.45	0.19
Trajectory Guidance	0.16	0.45	0.98	0.53	0.05	0.07	0.38	0.17
AnchorFormer	0.17	0.46	1.00	0.54	0.03	0.03	0.38	0.15
EvoDriveVLA	**0.16**	**0.44**	**0.96**	**0.52**	**0.02**	**0.02**	**0.33**	**0.12**

G Cross-dataset Evaluation on Bench2Drive

We additionally evaluate the same framework on Bench2Drive. Relative to the matched baseline, EvoDriveVLA reduces open-loop average L2 error and improves both closed-loop Driving Score (DS) and Success Rate (SR).

Table 11: Cross-dataset evaluation on Bench2Drive.

Method	Avg. L2 ↓ (Open-loop)	DS ↑ (Closed-loop)	SR (%) ↑ (Closed-loop)
Baseline	1.01	42.32	16.08
EvoDriveVLA	**0.97**	**44.46**	**16.74**

H Comparison with Related Distillation Paradigms

Table 12: Conceptual comparison with related distillation paradigms.

Method	Teacher Input	Visual Distillation	Trajectory Distillation	Candidate Generation	Student Supervision
Feature Distillation	Same observation	Feature matching	–	–	Visual features
Privileged Distillation	Future information	–	Single oracle trajectory	–	Trajectory/logits
Multi-Trajectory Distillation	Same observation	–	Multiple future modes	Explicit multi-modal prediction	Multi-trajectory supervision
EvoDriveVLA	Future-aware oracle teacher	Trajectory-guided token-level anchoring	Coarse-to-fine correction process	Refinement + MC-Dropout sampling	Visual tokens + selected hidden states/logits

I Limitations

Our evaluation is primarily conducted on the nuScenes and NAVSIM benchmarks, which predominantly feature driving scenes from specific geographic regions. Further validation on more diverse datasets with varying weather conditions and traffic rules would help to verify the model's zero-shot generalization across different countries.

J DATASET EXAMPLES



Front Left | Front | Front Right



Back Left | Back | Back Right

Student-Prompt: You're an autonomous vehicle's brain…Here are current six images from the car: 'CAM_FRONT': <image> …Here's some information you'll need:\nHistorical Trajectory (last 2 seconds): [(-2.0s):(0.11,-14.51)…(0.00, 0.00)]. Historical ego (last 2 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(-2.0s):(6.73 m/s, 0.15 m/s^2, 0.02 m/s^2, -0.13)… please output the plan waypoints (0.5s intervals) for the next 3 seconds.

Student-Output: [(0.05,3.76), (0.11,7.53), (0.22,11.20), (0.30,15.20), (0.73,19.99), (1.19,22.61)] These are the future waypoints.



Future 0 s | Future 1 s | Future 2 s | Future 3 s

Teacher-Prompt: You're an autonomous vehicle's brain…Here are current front cam image from the car: 'CURRENT CAM_FRONT': <image>…Here's some information you'll need:\nHistorical Trajectory (last 2 seconds): [(-2.0s): (0.11,-14.51)…Historical ego (last 2 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(-2.0s):(6.73 m/s, 0.15 m/s^2, 0.02 m/s^2, -0.13)… Future ego (next 3 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(0.5s):(7.40 m/s, -0.60 m/s^2, -0.11 m/s^2, -0.04)…Based on the provided particulars, please output the plan waypoints (0.5s intervals) for the next 3 seconds.

Teacher-Output: [(0.04,3.76), (0.12,7.51), (0.26,11.31), (0.43,15.10), (0.72,19.99), (1.09,22.66)] These are the future waypoints.

Refine-Teacher-Prompt: You're an autonomous vehicle's brain…Here are current front cam image from the car: 'CURRENT CAM_FRONT': <image>…Here's some information you'll need…Based on the provided particulars, please output the plan waypoints (0.5s intervals) for the next 3 seconds. I can provide the trajectory predictions from the student model here, and you can also refine them based on this: [(0.04,3.76), (0.12,7.51), (0.26,11.31), (0.43,15.10), (0.72,19.99), (1.09,22.66)].

Refine-Teacher-Output: [(0.05,3.76), (0.13,7.53), (0.23,11.33), (0.44,15.20), (0.72,18.99), (1.08,22.65)] These are the future waypoints.



Student-Prompt: You're an autonomous vehicle's brain…Here are current six images from the car: 'CAM_FRONT': <image> …Here's some information you'll need:\nHistorical Trajectory (last 2 seconds): [(-2.0s):(-0.02,-4.27)…(0.00, 0.00)]. Historical ego (last 2 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(-2.0s):(2.38 m/s, -0.24 m/s^2, 0.10 m/s^2, -0.21)…please output the plan waypoints (0.5s intervals) for the next 3 seconds.

Student-Output: [(0.01,0.92), (0.02,1.84), (0.05,2.73), (0.07,3.67), (0.00,4.50), (0.13,5.70)] These are the future waypoints.

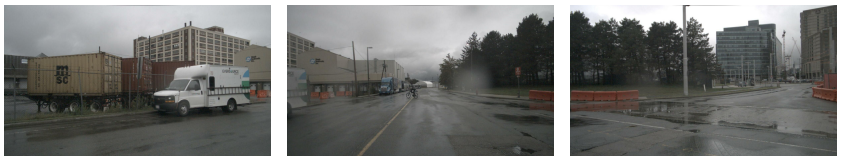


Teacher-Prompt: You're an autonomous vehicle's brain…Here are current front cam image from the car: 'CURRENT CAM_FRONT': <image>…Here's some information you'll need:\nHistorical Trajectory (last 2 seconds): [(-2.0s): (-0.02,-4.27)…Historical ego (last 2 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(-2.0s):(2.38 m/s, -0.24 m/s^2, 0.10 m/s^2, -0.21)… Future ego (next 3 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(0.5s):(1.83 m/s, -0.01 m/s^2, 0.09 m/s^2, -0.14)…Based on the provided particulars, please output the plan waypoints (0.5s intervals) for the next 3 seconds.

Teacher-Output: [(0.01,0.93), (0.02,1.84), (0.04,2.74), (0.07,3.79), (0.00,4.72), (0.13,5.84)] These are the future waypoints.

Refine-Teacher-Prompt: You're an autonomous vehicle's brain…Here are current front cam image from the car: 'CURRENT CAM_FRONT': <image>…Here's some information you'll need…Based on the provided particulars, please output the plan waypoints (0.5s intervals) for the next 3 seconds. I can provide the trajectory predictions from the student model here, and you can also refine them based on this: [(0.01,0.93), (0.02,1.84), (0.04,2.74), (0.07,3.79), (0.00,4.72), (0.13,5.84)].

Refine-Teacher-Output: [(0.01,0.93), (0.02,1.84), (0.05,2.76), (0.07,3.68), (0.10,4.72), (0.13,5.85)] These are the future waypoints.



Student-Prompt: You're an autonomous vehicle's brain…Here are current six images from the car: 'CAM_FRONT': <image> …Here' s some information you'll need:\nHistorical Trajectory (last 2 seconds): [(-2.0s):(-0.15,-16.49)…(0.00, 0.00)]. Historical ego (last 2 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(-2.0s):(8.49 m/s, -0.45 m/s^2, 0.01 m/s^2, -0.05)… please output the plan waypoints (0.5s intervals) for the next 3 seconds.

Student-Output: [(0.03,3.99), (0.08,7.72), (0.19,11.60), (0.03,15.64), (-0.02,19.60), (-0.10,23.00)] These are the future waypoints.



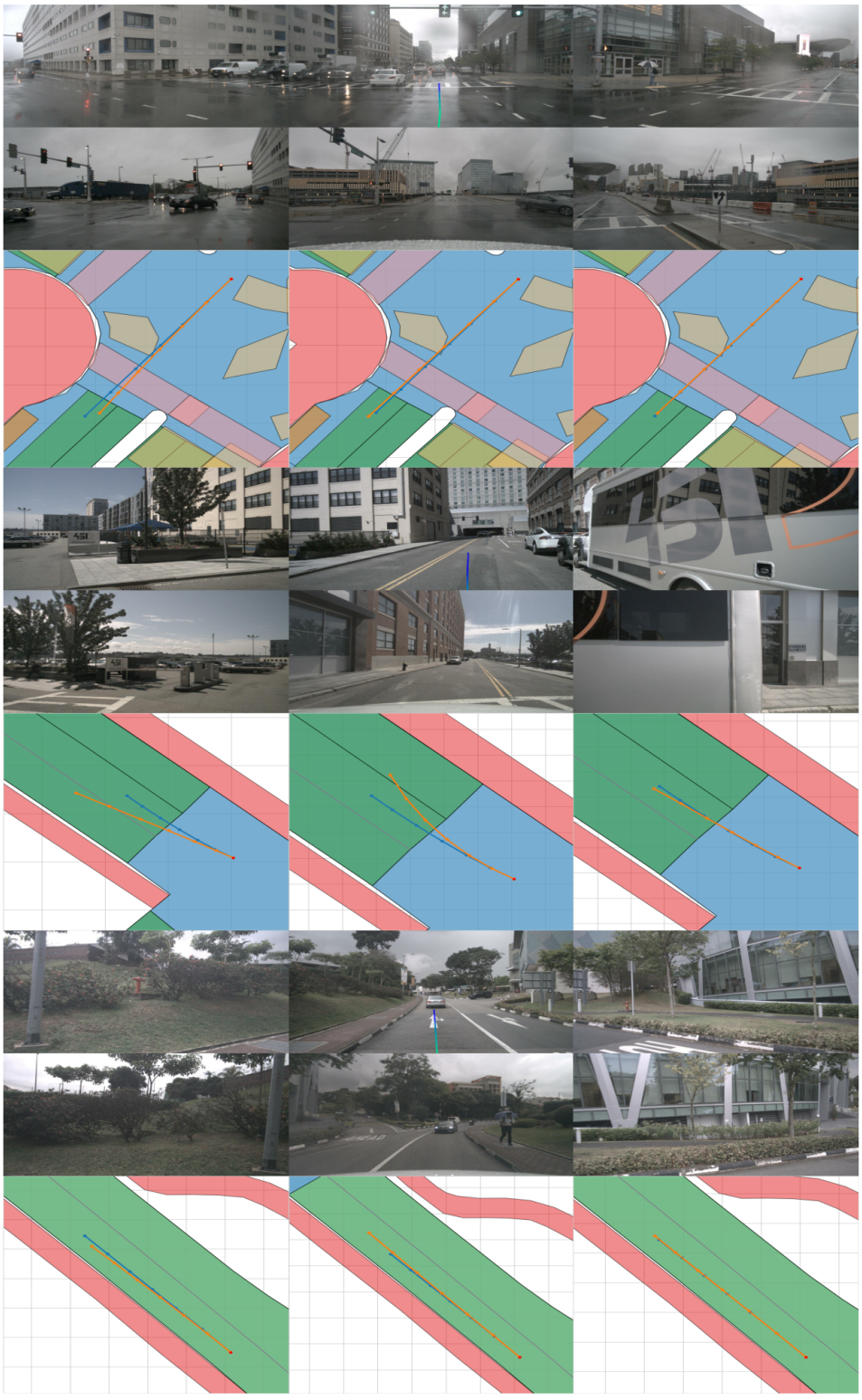
Teacher-Prompt: You're an autonomous vehicle's brain…Here are current front cam image from the car: 'CURRENT CAM_FRONT': <image>…Here' s some information you'll need:\nHistorical Trajectory (last 2 seconds): [(-2.0s): (-0.15,-16.49)…Historical ego (last 2 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(-2.0s):(8.49 m/s, -0.45 m/s^2, 0.01 m/s^2, -0.05)… Future ego (next 3 seconds), Format: (Velocity, Acceleration_x, Acceleration_y, steer): [(0.5s):(7.69 m/s, 0.09 m/s^2, -0.09 m/s^2, 0.03)…Based on the provided particulars, please output the plan waypoints (0.5s intervals) for the next 3 seconds.

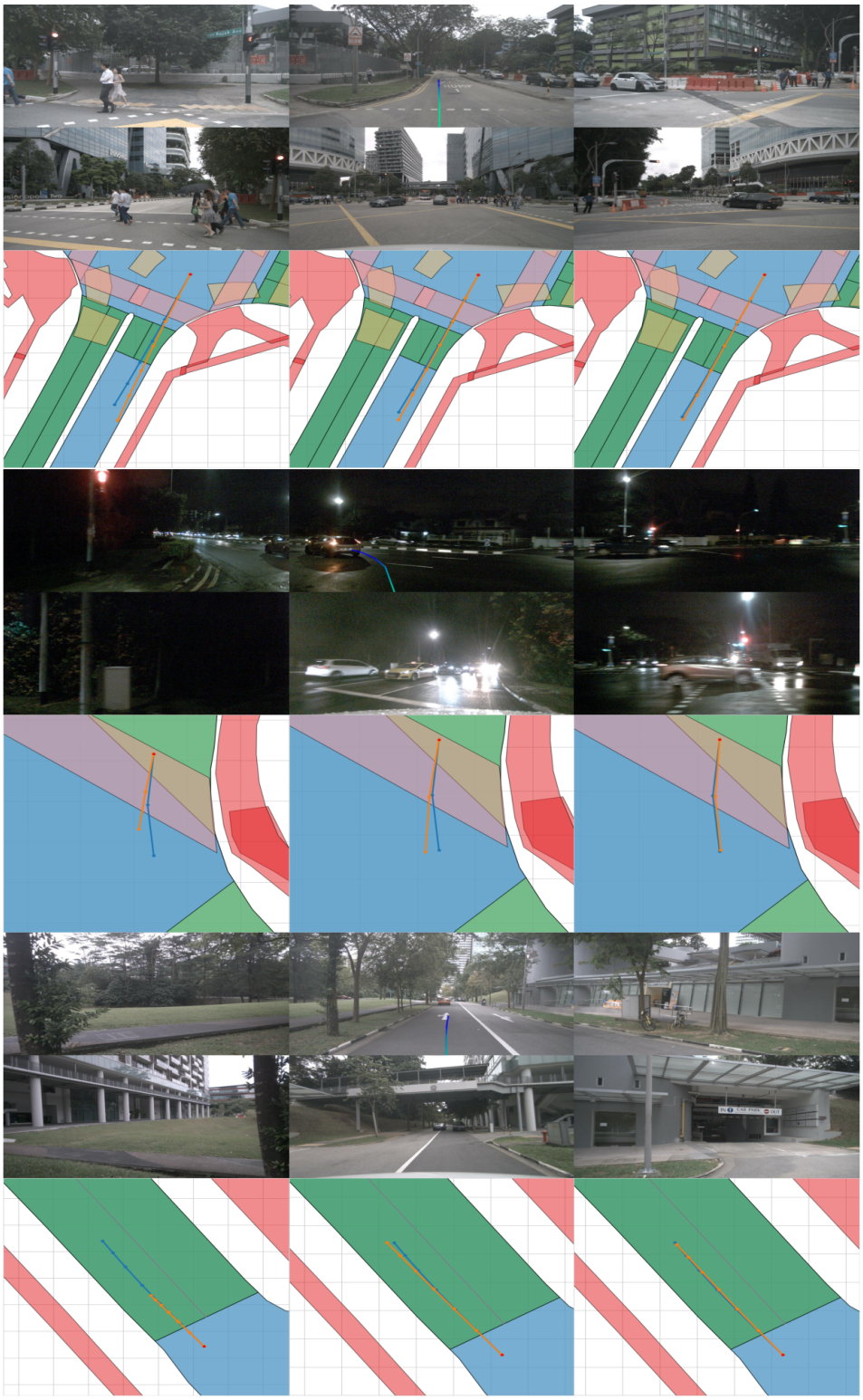
Teacher-Output: [(0.04,3.89), (0.06,7.73), (0.09,11.70), (0.05,15.79), (-0.01,19.76), (-0.00,23.40)] These are the future waypoints.

Refine-Teacher-Prompt: You're an autonomous vehicle's brain…Here are current front cam image from the car: 'CURRENT CAM_FRONT': <image>…Here' s some information you'll need…Based on the provided particulars, please output the plan waypoints (0.5s intervals) for the next 3 seconds. I can provide the trajectory predictions from the student model here, and you can also refine them based on this: [(0.04,3.89), (0.06,7.73), (0.09,11.70), (0.05,15.79), (-0.01,19.76), (-0.00,23.40)].

Refine-Teacher-Output: [(0.04,3.86), (0.07,7.77), (0.06,11.73), (0.03,15.69), (-0.02,19.30), (-0.10,23.00)] These are the future waypoints.

K Additional Qualitative Results




VAD
OmniDrive
EvoDriveVLA (Ours)