Orion-Lite: Distilling LLM Reasoning into Efficient Vision-Only Driving Models

Jing Gu Niccolò Cavagnero Gijs Dubbelman
Eindhoven University of Technology
{j.gu1, n.cavagnero, g.Dubbelman}@tue.nl
https://github.com/tue-mps/Orion-Lite

Abstract

*Leveraging the general world knowledge of Large Language Models (LLMs) holds significant promise for improving the ability of autonomous driving systems to handle rare and complex scenarios. While integrating LLMs into Vision-Language-Action (VLA) models has yielded state-of-the-art performance, their massive parameter counts pose severe challenges for latency-sensitive and energy-efficient deployment. Distilling LLM knowledge into a compact driving model offers a compelling solution to retain these reasoning capabilities while maintaining a manageable computational footprint. Although previous works have demonstrated the efficacy of distillation, these efforts have primarily focused on relatively simple scenarios and open-loop evaluations. Therefore, in this work, we investigate LLM distillation in more complex, interactive scenarios under closed-loop evaluation. We demonstrate that through a combination of latent feature distillation and ground-truth trajectory supervision, an efficient vision-only student model **Orion-Lite** can even surpass the performance of its massive VLA teacher, ORION. Setting a new state-of-the-art on the rigorous Bench2Drive benchmark, with a Driving Score of 80.6. Ultimately, this reveals that vision-only architectures still possess significant, untapped potential for high-performance reactive planning.*

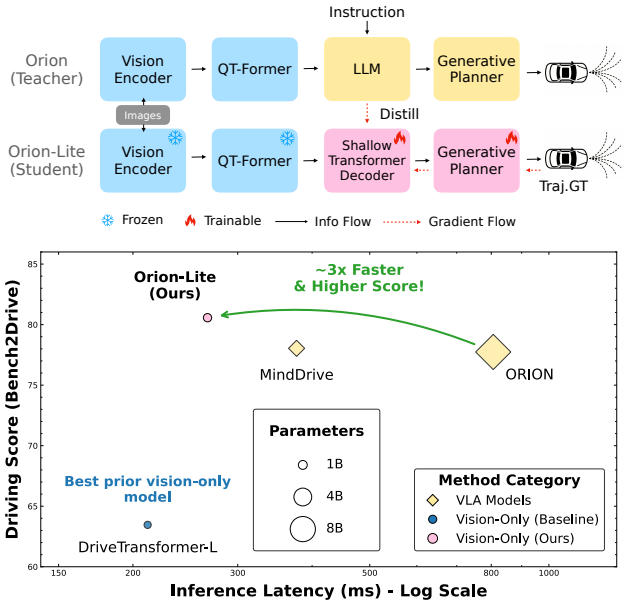

Figure 1. **Overview of the proposed distillation framework.** A joint distillation and trajectory supervision strategy (top) yields a student model, **Orion-Lite**, that is 3× faster than its teacher, establishing a new state-of-the-art on the closed-loop Bench2Drive benchmark (bottom).

1. Introduction

Recently, Vision-Language Models (VLMs) and Vision-Language-Action (VLA) architectures [9, 30, 33, 34, 37, 39] have emerged as a dominant paradigm in autonomous driving research. By integrating Large Language Models (LLMs) with vision encoders and aligning them with through visual question-answering (VQA), these methods can leverage the rich world knowledge embedded in LLMs. This integration of LLM modules introduces explicit causal reasoning into VLA driving models, allowing them to better optimize driving trajectories in complex, interactive scenarios [9, 10, 29, 38].

Consequently, VLA models currently achieve state-of-the-art performance across multiple autonomous driving benchmarks. However, their reliance on massive LLMs introduces severe computational bottlenecks, including prohibitive GPU memory consumption and high inference latency. While many of these architectures possess Chain-of-Thought (CoT) [9, 29, 38] capabilities for multi-turn visual reasoning, their practical closed-loop deployment typically relies on "direct" modes to mitigate latency. In this direct mode, intermediate reasoning steps are bypassed entirely, and the LLM is prompted with a static, pre-defined instruction template to generate latent waypoints [9, 10, 34, 38]. This essentially renders the LLM as a feature extractor.

While recent works have explored knowledge distillation to mitigate these bottlenecks, the efficacy of distilling LLMs for more challenging closed-loop driving scenarios remains an open question. For instance, DiMA [11] demonstrates improved inference speed by distilling LLM knowledge into a vision-only model, but its evaluation is strictly limited to open-loop metrics. Similarly, VERDI [8] performs distillation from Qwen-2.5-VL [1] for closed-loop evaluation on HugSim [43]. However, it primarily compares its student model against older baselines like UniAD [14] rather than against its own teacher model. Consequently, how much performance a student model can maintain relative to its teacher in more realistic, interactive environments remains underexplored. This brings us to our core research question: *how can the "reasoning" capabilities of an LLM inside a VLA be efficiently distilled without suffering a performance gap in challenging, closed-loop scenarios?*

To answer this question, we select Bench2Drive [18] as our rigorous closed-loop evaluation environment. Bench2Drive is the first benchmark comprehensively designed to assess an end-to-end autonomous driving (E2E-AD) system's multi-ability performance in a closed-loop manner, introducing 44 interactive scenarios (e.g., cut-ins, overtaking, detours), 23 weather conditions, and 12 distinct towns. Historically, evaluating E2E-AD methods relied on open-loop datasets (e.g., nuScenes [3]) using L2 displacement errors and collision rates, which often fail to reflect actual driving performance [18]. In fact, the average open-loop box collision rate on Bench2Drive is $4.5\times$ higher than on nuScenes (using UniAD, VAD, and ORION as baselines [9, 38]), highlighting the dataset's inherent complexity. Conversely, existing closed-loop protocols (e.g., Town05Long and Longest6[5, 28]) typically rely on a small set of fixed routes, where the standard driving score exhibits high variance due to unsmoothed metric functions and route randomness. Bench2Drive bridges this gap by having more environments and scenarios, offering a stable, highly interactive evaluation standard.

Focusing on this challenging benchmark, we take the currently publicly available state-of-the-art VLA model, ORION [9], and investigate the effect of different distillation strategies on our proposed student model, **Orion-Lite**, which replaces the heavy LLM with a lightweight transformer decoder. We find that, when employing a synergistic combination of latent distillation and ground-truth trajectory supervision, Orion-Lite surprisingly surpasses its 7B-parameter ORION teacher. Not relying on an LLM, Orion-Lite accelerates the reasoning module's inference by $150\times$, tripling the overall system speed. Moreover, it reduces total GPU memory usage from 31 GB to 8 GB, making it more suitable for actual deployment. Furthermore, qualitative analysis reveals that our distilled model exhibits superior robustness in complex edge cases where the original ORION model hesitates or fails. This lightweight, vision-only architecture achieves new state-of-the-art performance on the standard Bench2Drive benchmark, outperforming its teacher VLA-based ORION [9], the online Reinforcement Learning based MindDrive [10], and the World Model based UniDrive-WM [38].

While we do not introduce a fundamentally novel distillation algorithm, the core contribution of our work lies in the empirical demonstration that a standard distillation loss, combined with ground-truth trajectory supervision, allows a highly compressed student model to surpass its teacher's performance ceiling for challenging scenarios and under closed-loop evaluation. This suggests that the causal reasoning capability required for autonomous driving does not strictly need to manifest through a massive LLM during inference, positioning efficient visual reasoning as a highly promising direction for future research. In this work, we report these empirical findings and perform ablation studies to explore this phenomenon. A more exhaustive analysis of the underlying mechanisms is deferred to future work.

In summary, our main contributions are twofold:

- **Architectural Simplification:** we demonstrate that a compact and lightweight transformer decoder can replace a 7B-parameter LLM in a VLA driving model without compromising performance on current challenging closed-loop benchmarks. This heavily suggests that for standard, reactive E2E trajectory planning, massive LLMs may not be strictly necessary for inference.
- **State-of-the-Art Vision-Only Model:** we show that when our shallow student model is trained jointly with feature distillation and ground-truth supervision, it effectively outperforms the teacher's driving capabilities. Our method achieves new state-of-the-art results on the Bench2Drive closed-loop evaluation, outperforming competing VLA, RL, and WM methods using only a fraction of the computational resources.

Open Source: To facilitate reproducibility and support future research in efficient autonomous driving, all code, model weights, and evaluation scripts will be made publicly available upon acceptance.

2. Related Work

2.1. End-to-End Autonomous Driving

End-to-end (E2E) autonomous driving frameworks effectively map raw sensor inputs directly to planning trajectories or control signals. UniAD [14] pioneered this direction by unifying perception, prediction, and planning into a single framework. To enhance safety, VAD [20] incorporated vectorized planning constraints, while VADv2 [4] transitioned to a probabilistic paradigm. Recently, approaches like DiffusionDrive [26] have been employed to capture multimodal trajectory distributions, and methods such as

LAW [23] and WoTE [24] integrate next-frame prediction as an implicit world model to enhance the vision encoder's spatial-temporal representations. Despite these advancements, many E2E methods still suffer from error accumulation in closed-loop scenarios [40]. To address this, DriveTransformer [19] proposed a unified transformer framework processing perception and planning in parallel, achieving strong results on Bench2Drive. Our work builds upon this vision-only E2E paradigm but diverges significantly: we demonstrate that a high-performance vision-only model can be derived by distilling the latent cognitive capabilities of a complex VLA teacher into a drastically lighter decoder, indicating that current vision-only architectures still possess significant, untapped potential.

2.2. Vision-Language Models

The integration of LLMs into autonomous driving has led to the emergence of VLA models. EMMA [15] utilizes Gemini [32] to generate future trajectories natively as text tokens. OpenEMMA [37] extends this to open-source VLMs but relies on auxiliary 3D modules [27]. DriveVLM [33] adopts a dual-system approach for trajectory refinement. Recently, Alpamayo-R1 [35] demonstrated that reinforcement learning post-training can further improve reasoning quality and reasoning-action consistency. ORION [9] and OmniDrive [34] explore using LLMs to condition generative planners. More recent works incorporate online Reinforcement Learning (MindDrive [10]) and World Models (UniDrive-WM [38]). Crucially, in architectures like ORION and MindDrive, the final driving trajectory is decoded directly from the LLM's latent hidden states rather than its textual output. This architectural trait effectively renders the LLM as an overparameterized feature extractor. Our work capitalizes on this insight, challenging the necessity of the massive LLM during inference for standard E2E reactive planning tasks.

2.3. Knowledge Distillation

Knowledge distillation (KD) transfers capabilities from heavy teacher models to lightweight student models [2, 12]. DriveAdapter [16], Hydra-MDP [25], and Hydra-MDP++ [22] distill knowledge from a heavy vision-centric teacher to a more efficient vision-only student model. Distinct from these methods, our work focuses specifically on distilling the reasoning capabilities of the LLM within a VLA model into a vision-only student.

More directly relevant to our paradigm are VERDI [8] and DiMA [11], which distill VLM knowledge into vision models. VERDI aligns the VLM's text output with the vision model's predictions using complex progressive feature projectors. DiMA explores distilling VLM features via KL-divergence but limits its evaluation to open-loop metrics. We advance this research by directly distilling the continuous latent LLM features using simple $\mathcal{L}_1$ regression, avoiding auxiliary text encoders, offline rule-based experts, and suboptimal distributional metrics. Furthermore, we validate our framework in highly complex, realistic closed-loop evaluations.

3. Method

In this section, we detail our proposed knowledge distillation framework, designed to effectively compress the massive LLM inside a VLA driving model without sacrificing performance in complex, closed-loop scenarios. The overall pipeline is illustrated in Figure 1. Our framework utilizes the state-of-the-art ORION [9] as the teacher model. To create our highly efficient, vision-only end-to-end model, we introduce a lightweight distillation module (Section 3.3) that transfers the latent reasoning representations from the teacher LLM to a shallow transformer decoder. To avoid computational redundancy during distillation, we utilize the teacher's intermediate state embeddings (Section 3.2) to represent the dense visual and contextual information.

3.1. Preliminaries

ORION is a fully map-free E2E method that relies on Navigation Commands (NC) as trajectory conditions, utilizing EVA-02-L [7] as the vision encoder, Vicuna-v1.5 [41] as the LLM, and a VAE-based generative planner [21].

ORION takes multi-view camera streams as input, employing a vision encoder to extract spatial features. High-level driving commands are vectorized and fused with these visual representations. To ensure robust feature learning, an auxiliary perception head is attached to the image features and supervised via downstream perception tasks. For temporal context, a QT-Former module extracts compact embeddings from historical frames, maintaining them in a memory bank. Concurrently, text prompts are tokenized and embedded before being fused with the visual features. The resulting multimodal representations are fed into an LLM, which serves two purposes: it either generates intermediate latent features for motion planning or it performs auto-regressive token generation to answer visual queries and explicit reasoning. Ultimately, the intermediate latent features are processed by a VAE-based generative planner to produce the final trajectory prediction.

3.2. State Embedding Extraction

Following the ORION architecture [9], we first extract multi-view image features F_m from the frozen vision encoder. The QT-Former, a query-based temporal module, employs learnable scene queries $Q_s \in \mathbb{R}^{N_s \times C_q}$ and perception queries $Q_p \in \mathbb{R}^{N_p \times C_q}$, where N_s and N_p denote the number of queries and C_q represents the channel dimension.

These queries exchange information via self-attention and subsequently interact with the image features F_m

through cross-attention. The perception queries are then routed to task-specific heads for object detection, traffic state recognition, and dynamic agent motion prediction. To efficiently aggregate historical context, ORION utilizes history queries $Q_h \in \mathbb{R}^{N_h \times C_q}$ alongside a long-term memory bank $M \in \mathbb{R}^{(N_h \times n) \times C_q}$.

The final concatenated features comprising the map vision embedding, ego-vehicle status, and the driving command serve as the input tokens $T_c \in \mathbb{R}^{N_c \times C_c}$ for our student model, where N_c is the sequence length and C_c is the channel dimension. In the teacher model, the LLM processes T_c alongside tokenized text prompts to output the latent planning tokens $T_p \in \mathbb{R}^{1 \times C_p}$. For our student framework, we discard the text prompts entirely to achieve a vision-only architecture, utilizing the teacher's T_p as the pseudo-ground truth distillation target.

3.3. Lightweight Distillation Module

Our student module replaces the massive 7B-parameter LLM with a highly efficient transformer-based architecture. This module consists of an input projection layer, a learnable planning query, a shallow standard transformer decoder, and an output projection layer.

The input projection, implemented as a linear layer followed by layer normalization, compresses the input token channels from C_c to a hidden dimension C_h, yielding the compressed tokens $T_{cc} \in \mathbb{R}^{N_c \times C_h}$. To emulate the generative planning mechanism of the LLM, we initialize a learnable planning query $Q_{plan} \in \mathbb{R}^{1 \times C_h}$. Through the cross-attention mechanism within the 6-layer transformer decoder, Q_{plan} (Query) attends to the dense compressed tokens T_{cc} (Key/Value), extracting the critical spatio-temporal features necessary for trajectory generation. Finally, the output projection layer maps the decoder's output back to the original planning token dimension C_p, perfectly aligning the student's output space with the teacher's generative planner. This bottleneck design effectively retains essential planning information to manage complex scenarios while drastically reducing computational overhead.

3.4. Training Objectives

Feature Mimic Loss. Let $T_{student} \in \mathbb{R}^{1 \times C_p}$ denote the final projected output of the student decoder. Using the teacher's generated planning tokens T_p as the target, we apply an $\mathcal{L}_1$ regression loss, $\mathcal{L}_{mimic}$, to minimize the representational divergence. The pure distillation loss over a batch size B is formulated as:

$$\mathcal{L}_{mimic} = \frac{1}{B \cdot C_p} \sum_{b=1}^{B} \left\| T_{student}^{(b)} - T_{p}^{(b)} \right\|_1 \quad (1)$$

Joint Distillation and E2E Supervision. Rather than relying on the LLM's latent features only, our training phase jointly optimizes the newly initialized transformer decoder alongside the pre-trained VAE generative planner. Throughout this process, the pre-trained vision encoder and QT-Former are kept strictly frozen with weights initialized from ORION. Because the vision encoder already captures robust spatial-temporal representations, we focus the gradient updates entirely on the lightweight student decoder and the ensuing planner.

Following standard E2E formulations [20], we incorporate environmental feedback via collision loss $\mathcal{L}_{col}$ and boundary loss $\mathcal{L}_{bd}$. Furthermore, we apply an $\mathcal{L}_1$ regression loss $\mathcal{L}_{reg}$ for deterministic trajectory prediction. To properly align the reasoning and action spaces within the generative planner, we retain the Kullback–Leibler divergence loss $\mathcal{L}_{vae}$ adopted by the ORION teacher. To clearly separate supervision signals, we decompose the overall objective into two components: (i) a ground-truth supervision term that aggregates all driving-related penalties, and (ii) a distillation term that regularizes the student with the teacher's latent distribution. Specifically, we define:

$$\mathcal{L}_{GT} = \mathcal{L}_{col} + \lambda_{bd}\mathcal{L}_{bd} + \lambda_{reg}\mathcal{L}_{reg} + \lambda_{vae}\mathcal{L}_{vae} \quad (2)$$

In practice, we set $\lambda_{bd} = \lambda_{reg} = \lambda_{vae} = 3$. The final training objective is then expressed as the sum of the GT supervision and distillation terms:

$$\mathcal{L}_{total} = \mathcal{L}_{GT} + \mathcal{L}_{mimic} \quad (3)$$

This combination of ground truth and distillation losses ensures that the student model effectively distills knowledge from the ORION teacher, while retaining the foundational driving skills learned from the recorded trajectories.

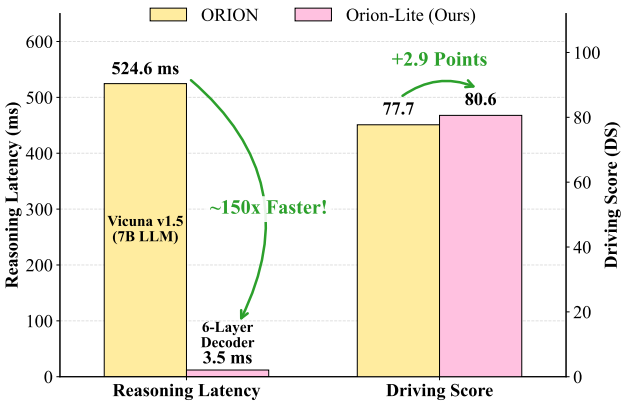

Figure 2. **Latency and Driving Score Comparison.** Our distilled framework demonstrates a massive reduction in inference latency compared to the teacher model while improving the overall Driving Score. Latency is measured by the averaged inference steptime on CARLA evaluated on an A6000 GPU.

Method	Reference	Condition	Modality	Closed-loop Metrics				Open-loop Metric	Latency (ms) ↓
				DS ↑	SR (%) ↑	Efficiency ↑	Comfortness ↑	Avg. L2 ↓	
TCP* [36]	NeurIPS 22	TP	C	40.7	15.0	54.3	47.8	1.70	83
TCP-ctrl*	NeurIPS 22	TP	C	30.5	7.3	56.0	51.5	-	83
TCP-traj*	NeurIPS 22	TP	C	59.9	30.0	76.5	18.1	1.70	83
TCP-traj w/o distillation	NeurIPS 22	TP	C	49.3	20.5	78.8	23.0	1.96	83
ThinkTwice* [17]	CVPR 23	TP	C	62.4	31.2	69.3	16.2	0.95	762
DriveAdapter* [16]	ICCV 23	TP	C&L	64.2	33.1	70.2	16.0	1.01	931
AD-MLP [40]	arXiv 23	NC	C	18.1	0.0	48.5	22.6	3.64	3
UniAD-Tiny [14]	CVPR 23	NC	C	40.7	13.2	123.9	47.0	0.80	420
UniAD-Base [14]	CVPR 23	NC	C	45.8	16.4	129.2	43.6	0.73	663
VAD [20]	ICCV 23	NC	C	42.4	15.0	157.9	46.0	0.91	278
GenAD [42]	ECCV 24	NC	C	44.8	15.9	-	-	-	121
MomAD [31]	CVPR 25	NC	C	44.5	16.7	170.2	48.6	0.87	242
DriveTransformer-Large [19]	ICLR 25	NC	C	63.5	35.0	100.6	20.8	0.62	212
MindDrive [10]	arXiv 25	NC	C	78.0	55.1	-	-	-	377
UniDrive-WM [38]	arXiv 26	NC	C	79.2	**56.4**	158.4	28.0	0.64	-
SimLingo† [29]	CVPR 25	NC	C	85.9	66.8	244.2	25.5	-	4139
ORION (0.5B) [10]	ICCV 25	NC	C	72.9	45.8	-	-	-	-
ORION (7B Teacher) [9]	ICCV 25	NC	C	77.7	54.6	151.5	17.4	0.68	806
Orion-Lite (**0.1B Ours**)	-	NC	C	**80.6** (+2.9)	55.5 (+0.9)	157.7	10.3	0.79	267

Table 1. **Closed-loop and Open-loop Results of E2E-AD Methods in Bench2Drive.** All models are trained on the standard base set (1K clips), except for SimLingo†, which utilizes extended external training data (e.g., 3.1M samples). C/L refers to camera/LiDAR. Avg. L2 is averaged over predictions in 2 seconds under 2Hz. Latency is measured by the average inference step-time during CARLA evaluation on an A6000 GPU. * denotes expert feature distillation. NC: navigation command, TP: target point, DS: Driving Score, SR: Success Rate. Note: 0.5B and 7B indicate the parameters of the LLM module alone, rather than the full model's parameter count.

Method	Reference	Condition	Modality	Ability (%) ↑					
				Merging	Overtaking	Emergency Brake	Give Way	Traffic Sign	Mean
TCP* [36]	NeurIPS 22	TP	C	16.2	20.0	20.0	10.0	7.0	14.6
TCP-ctrl*	NeurIPS 22	TP	C	10.3	4.4	10.0	10.0	6.5	8.2
TCP-traj*	NeurIPS 22	TP	C	8.9	24.3	51.7	40.0	46.3	34.2
TCP-traj w/o distillation	NeurIPS 22	TP	C	17.1	6.7	40.0	**50.0**	28.7	28.5
ThinkTwice* [17]	CVPR 23	TP	C	27.4	18.4	35.8	**50.0**	54.2	37.2
DriveAdapter* [16]	ICCV 23	TP	C&L	28.8	26.4	48.8	**50.0**	56.4	42.1
AD-MLP [40]	arXiv 23	NC	C	0.0	0.0	0.0	0.0	4.35	0.87
UniAD-Tiny [14]	CVPR 23	NC	C	8.9	9.3	20.0	20.0	15.4	14.7
UniAD-Base [14]	CVPR 23	NC	C	14.1	17.8	21.7	10.0	14.2	15.6
VAD [20]	ICCV 23	NC	C	8.1	24.4	18.6	20.0	19.2	18.1
DriveTransformer-Large [19]	ICLR 25	NC	C	17.6	35.0	48.4	40.0	52.1	38.6
MindDrive [10]	arXiv 25	NC	C	**32.9**	**75.8**	68.3	**50.0**	57.9	56.9
UniDrive-WM [38]	arXiv 26	NC	C	29.8	74.0	**79.8**	40.0	**71.3**	59.0
ORION (0.5B) [10]	ICCV 25	NC	C	26.3	62.2	55.6	**50.0**	63.3	51.4
ORION (7B Teacher) [9]	ICCV 25	NC	C	25.0	71.1	78.3	30.0	69.2	54.7
Orion-Lite (**0.1B Ours**)	-	NC	C	28.8	75.6	78.3	**50.0**	70.0	**60.5** (+5.8)

Table 2. **Multi-Ability Results of E2E-AD Methods under base set.** * denote expert feature distillation. C/L refers to camera/LiDAR. NC: navigation command, TP: target point.

Mimic Loss	Traj. GT	Closed-loop		Ability (%) ↑					
		DS ↑	SR ↑	M	O	EB	GW	TS	Mean
-	✓	73.9	50.0	26.3	66.7	71.7	10.0	63.7	47.7
✓	-	76.0	50.7	26.6	66.7	68.3	40.0	64.7	53.3
✓	✓	**80.6**	**55.5**	**28.8**	**75.6**	**78.3**	**50.0**	**70.0**	**60.5**

Table 3. **Impact of supervision.** We evaluate the model's performance on closed-loop metrics and specific driving abilities. **M**: Merging, **O**: Overtaking, **EB**: Emergency Brake, **GW**: Give Way, **TS**: Traffic Sign, **DS/SR**: Driving Score/Success Rate.

Setting	Epochs	Closed-loop		Ability (%) ↑					
		DS ↑	SR ↑	M	O	EB	GW	TS	Mean
Orion	18 (default)	77.7	54.6	25.0	71.1	**78.3**	30.0	69.2	54.7
Orion	24	77.1	50.7	25.3	57.8	70.0	40.0	69.0	52.0
Orion-Lite	20	**80.6**	**55.5**	**28.8**	**75.6**	**78.3**	**50.0**	**70.0**	**60.5**

Table 4. **Impact of training duration.** Comparison between the standard Orion model, Orion with extended training duration and our Orion-Lite.

Figure 3. **Qualitative comparison in interactive scenarios.** Top rows: rollouts from the Orion teacher model. Bottom rows: rollouts from our student model. Sequences are visualized at 5-frame intervals, with overlaid points indicating predicted future trajectories. Green checks (✓) denote successful, intervention-free maneuvers, while red crosses (×) indicate task failures. While Orion frequently hesitates or fails to safely merge or overtake behind obstacles, our student model demonstrates robust spatial awareness, successfully executing smooth and decisive maneuvers.

Loss	L2 (m) ↓				Collision (%) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
$\mathcal{L}_1$	0.32	0.76	1.30	0.79	0.27	0.51	0.85	0.54
$\mathcal{L}_2$	0.32	0.75	1.30	0.79	0.36	0.55	0.90	0.60
KL	0.32	0.75	1.30	0.79	0.34	0.63	0.91	0.63
Huber	0.31	0.70	1.24	0.75	0.41	0.64	1.04	0.70

Table 5. **Impact of distance metrics.** We evaluate the impact of different distance metrics for distillation.

Encoder Init.	Frozen	Closed-loop		Ability (%) ↑					
		DS ↑	SR ↑	M	O	EB	GW	TS	Mean
EVA-02-L	✓	54.3	21.8	11.3	33.3	28.3	**30.0**	41.1	28.8
EVA-02-L		72.4	47.3	23.8	53.3	**70.0**	**30.0**	**67.9**	49.0
Orion	✓	**77.4**	**51.8**	**30.0**	**66.7**	68.3	**30.0**	65.8	**52.2**

Table 6. **Ablation on vision encoder initialization.** We evaluate driving performance using different pre-trained weights.

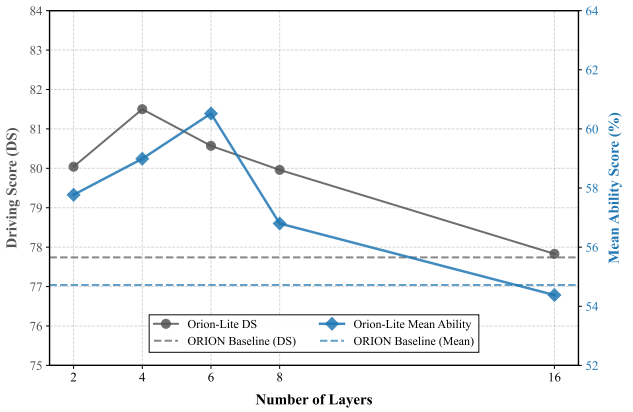

Figure 4. **Impact of Decoder Depth.** Driving Score and mean Multi-ability Score across varying numbers of transformer decoder layers.

4. Experiments

4.1. Dataset

We train and evaluate our models utilizing the Bench2Drive dataset [18], which employs CARLA V2 [6] as its closed-loop evaluation protocol for E2E autonomous driving. The official training set contains 1,000 annotated clips, each comprising multi-view camera data (6 cameras), 5 radars, and 1 LiDAR sweep. Radar and LiDAR sweeps are discarded, as our model leverages RGB cameras only. Each clip spans roughly 150 meters and captures a specific interactive driving scenario. For our distillation pipeline, we utilize 950 clips for training and 50 for open-loop validation. Comprehensive closed-loop evaluations are conducted on the official Bench2Drive CARLA simulator, encompassing 220 short routes across 44 complex, interactive scenarios.

4.2. Evaluation Metrics

Following the Bench2Drive benchmark, we report five different metrics for closed-loop evaluation: Driving Score (DS), Success Rate (SR), Efficiency, Comfortness, and Multi-Ability. *Driving Score*, standard in CARLA [6], is the primary metric, multiplying the route completion percentage by a penalty discount factor for any infractions (e.g., collisions, running red lights). *Success Rate* measures the percentage of successfully completed routes within a designated time limit. *Efficiency* and *Comfortness* quantify the agent's navigational speed and kinematic smoothness, respectively. *Multi-Ability* independently evaluates the model across five advanced urban driving skills (Merging, Overtaking, Emergency Braking, Giving Way, and Traffic Signs). For the sake of completeness, we also report open-loop validation. In this case, analogously to ST-P3 [13], we report the L2 trajectory displacement error (Avg. L2) and the bounding box collision rate.

4.3. Implementation Details

Model Settings. Unlike VLA-based frameworks [9, 38], our proposed Orion-Lite is a vision-only architecture that receives raw camera frames as input and directly yields trajectory predictions.

Training Process. We initialize the vision encoder, QT-Former, and VAE planner with pre-trained ORION [9] weights. The 7B LLM is replaced by our randomly initialized 6-layer decoder (reducing reasoning parameters from 7B to 0.1B). During distillation, the vision encoder and QT-Former remain frozen; only the student decoder and VAE planner are updated. All ablations adopt this setting unless otherwise specified. Models are trained for 20 epochs at 640×640 resolution using AdamW with learning rate 5×10^{-5} and weight decay 1×10^{-4}. Training requires $\sim$20 hours on a single RTX A6000 (48GB) GPU. Closed-loop evaluations utilize the same device. Further details are provided in our open-source repository.

5. Results

5.1. Main Results

As detailed in Figure 2 and Table 1, our distilled model yields exceptional efficiency gains while achieving superior performance with respect to the teacher model. During the inference phase, the reasoning module of our student model is 150$\times$ faster than the VLA teacher's LLM and it reduces the inference GPU memory usage from 31 GB to 8 GB. This results in a 3$\times$ decrease of the overall end-to-end system latency. On top of these massive benefits in latency and memory consumption, our distilled model surpasses the ORION teacher by +2.9 DS, +0.9 SR, and +5.8 in Mean Multi-Ability (see Table 2).

Evaluated under the Bench2Drive closed-loop evaluation, our lightweight vision-only framework establishes a new state-of-the-art across closed-loop metrics, also outperforming the recent online RL and WM-based E2E-AD models[10, 38]. This explicitly answers our core research question: *through our distillation framework, the reasoning capabilities of a massive LLM can be effectively compressed into a lightweight transformer decoder without compromising and, in fact, improving performance in challenging closed-loop scenarios.*

Figure 3 visualizes the closed-loop behaviour of our student model compared to the ORION teacher in highly interactive scenarios. The comparative rollout highlights a specific case where the ego-vehicle is navigating around dynamic obstacles. In this scenario, the ORION teacher's generative planner frequently hesitates or fails to execute a safe merge or overtake when positioned behind an obstacle. Conversely, our student model maintains robust spatial awareness and successfully executes a smooth maneuver.

In the subsequent ablation study, we conduct further experiments to verify whether both the mimic loss and trajectory supervision are strictly necessary for this performance leap, while also analyzing the role of the mimic loss during training and the efficacy of various mimic loss formulations.

5.2. Ablation Study

The Role of Mimic Loss. Table 3 isolates the impact of the mimic loss. When training the shallow decoder from scratch relying solely on trajectory ground truth ($\mathcal{L}_{GT}$ without $\mathcal{L}_{mimic}$), performance drops to 73.9 DS. Notably, this still represents $\sim$95% of the teacher's performance, indicating that the frozen ORION vision encoder already provides a highly robust spatio-temporal foundation. Conversely, applying the mimic loss alone recovers 98% of the teacher's performance (76.0 DS). However, when applied jointly, the model achieves SOTA performance (80.6 DS). This shows that a synergistic combination of latent feature distillation and standard GT supervision can deliver better performance than the teacher itself, for a much lower inference footprint.

Impact of Training Duration. As shown in Table 4, directly training the ORION teacher for extended epochs (from 18 to 24 epochs) purely on ground truth results in performance degradation (77.7 DS $\rightarrow$ 77.1 DS), likely due to overfitting to the training dataset. In contrast, our joint distillation model can be stably trained for 20 epochs to achieve 80.6 DS. We attribute this to the powerful regularizing effect of knowledge distillation. The teacher LLM's latent embeddings act as soft labels, smoothing the target distribution and preventing the student from overfitting to hard, deterministic trajectory targets.

Decoder Depth. Figure 4 illustrates the effect of scaling the student decoder's depth. When utilizing mimic loss, a highly compressed 4-layer model achieves a remarkable

peak in Driving Score (81.5 DS). However, the 6-layer configuration achieves a superior Mean Multi-Ability score (60.5%), demonstrating a more robust mastery across diverse, complex driving skills (e.g., merging) rather than merely optimizing the base navigation score. Consequently, we adopt the 6-layer architecture as our default Orion-Lite model. Scaling beyond 6 layers causes both DS and Mean Ability Score to steadily decline. This suggests that mapping to the LLM's latent driving intent requires relatively low representational capacity; an overly deep student network risks overfitting to the distillation task itself, thereby degrading generalization in unseen closed-loop scenarios.

Distance Metrics for Distillation. Table 5 evaluates various distance metrics for $\mathcal{L}_{mimic}$. Because the target planning tokens (T_p) are dense, continuous feature coordinates rather than discrete class logits, applying distributional metrics like KL-Divergence requires artificially normalizing the feature space (e.g. via softmax), which distorts the latent geometry. Consequently, standard Euclidean regression metrics ($\mathcal{L}_1$, $\mathcal{L}_2$) naturally outperform KL-Divergence for feature matching. Specifically, $\mathcal{L}_1$ yields the lowest collision rates. We attribute this to the fact that $\mathcal{L}_1$ regression is inherently more robust to the extreme outlier activations.

Vision Encoder Initialization. To assess whether the teacher training stage provides improved visual representations for driving, Table 6 compares different vision encoder initializations under trajectory-only supervision (i.e., without the mimic loss). We adopt Orion-Lite as the base architecture and keep all other settings unchanged unless stated otherwise. Initializing the encoder from the Orion teacher and keeping it frozen yields the best performance, achieving 77.4 DS and 51.8 SR. In contrast, replacing it with an EVA-02-L initialization while still freezing the encoder results in a substantial drop, to 54.3 DS and 21.8 SR. This large performance gap suggests that the teacher-trained encoder already captures driving-specific visual representations that are absent in a generic initialization. When the EVA-02-L encoder is unfrozen and fine-tuned, performance improves significantly to 72.4 DS and 47.3 SR, indicating that trajectory supervision can adapt a generic backbone to the driving domain. However, it still falls short of the frozen Orion initialization by 5.0 DS. These findings suggest that Orion's vision encoder acquires important semantic and spatial priors during joint training with the LLM, which are not easily recovered through trajectory supervision alone.

6. Limitations and Future Work

While our student model achieves a $150\times$ speedup in the reasoning module, the heavy pre-trained vision encoder (e.g., EVA-02-L) becomes the primary computational bottleneck during overall system inference. Furthermore, our distillation pipeline fundamentally relies on the prior existence of a fully trained, computationally expensive VLA teacher model. Additionally, our empirical validation is currently focused on the Bench2Drive benchmark. Although Bench2Drive is one of the most challenging benchmarks, future research should verify these findings across diverse driving datasets and explore the optimization the vision encoder itself together with the QT-Former. It is recommended to enhance the Bench2Drive test set with extra complex long-tail, edge-case scenarios to better research the need for VLA reasoning in autonomous driving. Additionally, developing novel frameworks capable of injecting broad world knowledge from an LLM directly into a visual reasoning module without the need to curate massive, domain-specific VQA datasets presents a highly promising direction to circumvent the need for massive teacher models entirely.

7. Conclusion

In this work, we address the severe computational bottlenecks of deploying Vision-Language-Action (VLA) models by introducing a streamlined, highly effective knowledge distillation framework. We show how to distill the latent reasoning capabilities of a massive 7B-parameter LLM into a shallow, efficient transformer decoder without suffering a performance gap in challenging, closed-loop scenarios. In fact, through joint distillation and ground truth trajectory supervision, our vision-only student model consistently surpasses its massive VLA teacher in these highly complicated driving scenarios.

Our proposed Orion-Lite architecture drastically reduces inference latency and memory footprint while establishing a new state-of-the-art on the complex Bench2Drive benchmark. Ultimately, our findings suggest that rather than indefinitely scaling architectural parameters for inference, optimizing training paradigms and distilling latent reasoning capabilities can unlock significant, untapped potential in vision-only end-to-end autonomous driving.

References

[1] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025. 2

[2] Lucas Beyer, Xiaohua Zhai, Amélie Royer, Larisa Markeeva, Rohan Anil, and Alexander Kolesnikov. Knowledge distillation: A good teacher is patient and consistent. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2022. 3

[3] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2020. 2

[4] 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. 2

[5] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(11):12878–12895, 2022. 2

[6] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. CARLA: An open urban driving simulator. In *CoRL*, 2017. 6, 7

[7] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. EVA-02: A visual representation for neon genesis. *Image and Vision Computing*, 149:105171, 2024. 3

[8] Bowen Feng, Zhiting Mei, Baiang Li, Julian Ost, Filippo Ghilotti, Roger Girgis, Anirudha Majumdar, and Felix Heide. VERDI: VLM-embedded reasoning for autonomous driving. *arXiv preprint arXiv:2505.15925*, 2025. 2, 3

[9] Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. *arXiv preprint arXiv:2503.19755*, 2025. 1, 2, 3, 5, 7

[10] Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Hongwei Xie, Bing Wang, Guang Chen, Dingkang Liang, and Xiang Bai. MindDrive: A vision-language-action model for autonomous driving via online reinforcement learning. *arXiv preprint arXiv:2512.13636*, 2025. 1, 2, 3, 5, 7

[11] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025. 2, 3

[12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3

[13] 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 *Eur. Conf. Comput. Vis.*, 2022. 7

[14] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023. 2, 5

[15] Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, et al. EMMA: End-to-end multimodal model for autonomous driving. *arXiv preprint arXiv:2410.23262*, 2024. 3

[16] Xiaosong Jia, Yulu Gao, Li Chen, Junchi Yan, Patrick Langechuan Liu, and Hongyang Li. DriveAdapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In *Int. Conf. Comput. Vis.*, 2023. 3, 5

[17] Xiaosong Jia, Penghao Wu, Li Chen, Jiangwei Xie, Conghui He, Junchi Yan, and Hongyang Li. Think twice before driving: Towards scalable decoders for end-to-end autonomous driving. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2023. 5

[18] Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2Drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. In *Adv. Neural Inform. Process. Syst.*, 2024. 2, 6

[19] Xiaosong Jia, Junqi You, Zhiyuan Zhang, and Junchi Yan. DriveTransformer: Unified transformer for scalable end-to-end autonomous driving. *arXiv preprint arXiv:2503.07656*, 2025. 3, 5

[20] 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 *Int. Conf. Comput. Vis.*, 2023. 2, 4, 5

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[22] 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*, 2025. 3

[23] Yingyan Li, Lue Fan, Jiawei He, Yuqi Wang, Yuntao Chen, Zhaoxiang Zhang, and Tieniu Tan. Enhancing end-to-end autonomous driving with latent world model. *arXiv preprint arXiv:2406.08481*, 2024. 3

[24] Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via BEV world model. *arXiv preprint arXiv:2504.01941*, 2025. 3

[25] 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*, 2024. 3

[26] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025. 2

[27] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3D bounding box estimation using deep learning and geometry. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2017. 3

[28] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. Multi-modal fusion transformer for end-to-end autonomous driving. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2021. 2

[29] Katrin Renz, Long Chen, Elahe Arani, and Oleg Sinavski. SimLingo: Vision-only closed-loop autonomous driving with language-action alignment. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025. 1, 5

[30] Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. LMDrive: Closed-loop end-to-end driving with large language models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2024. 1

[31] Ziying Song, Caiyan Jia, Lin Liu, Hongyu Pan, Yongchang Zhang, Junming Wang, Xingyu Zhang, Shaoqing Xu, Lei Yang, and Yadan Luo. Don't shake the wheel: Momentum-aware planning in end-to-end autonomous driving. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025. 5

[32] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 3

[33] 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. 1, 3

[34] 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 *IEEE Conf. Comput. Vis. Pattern Recog.*, 2025. 1, 3

[35] Yan Wang, Wenjie Luo, Junjie Bai, Yulong Cao, Tong Che, Ke Chen, Yuxiao Chen, Jenna Diamond, Yifan Ding, Wenhao Ding, et al. Alpamayo-r1: Bridging reasoning and action prediction for generalizable autonomous driving in the long tail. *arXiv preprint arXiv:2511.00088*, 2025. 3

[36] Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In *Adv. Neural Inform. Process. Syst.*, 2022. 5

[37] Shuo Xing, Chengyuan Qian, Yuping Wang, Hongyuan Hua, Kexin Tian, Yang Zhou, and Zhengzhong Tu. OpenEMMA: Open-source multimodal model for end-to-end autonomous driving. In *WACV*, 2025. 1, 3

[38] Zhexiao Xiong, Xin Ye, Burhan Yaman, Sheng Cheng, Yiren Lu, Jingru Luo, Nathan Jacobs, and Liu Ren. UniDrive-WM: Unified understanding, planning and generation world model for autonomous driving. *arXiv preprint arXiv:2601.04453*, 2026. 1, 2, 3, 5, 7

[39] Yi Xu, Yuxin Hu, Zaiwei Zhang, Gregory P Meyer, Siva Karthik Mustikovela, Siddhartha Srinivasa, Eric M Wolff, and Xin Huang. VLM-AD: End-to-end autonomous driving through vision-language model supervision. *arXiv preprint arXiv:2412.14446*, 2024. 1

[40] Jiang-Tian Zhai, Ze Feng, Jinhao Du, Yongqiang Mao, Jiang-Jiang Liu, Zichang Tan, Yifu Zhang, Xiaoqing Ye, and Jingdong Wang. Rethinking the open-loop evaluation of end-to-end autonomous driving in nuScenes. *arXiv preprint arXiv:2305.10430*, 2023. 3, 5

[41] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Adv. Neural Inform. Process. Syst.*, 2023. 3

[42] Wenzhao Zheng, Ruiqi Song, Xianda Guo, Chenming Zhang, and Long Chen. GenAD: Generative end-to-end autonomous driving. In *Eur. Conf. Comput. Vis.*, 2024. 5

[43] Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugsim: A real-time, photo-realistic and closed-loop simulator for autonomous driving. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2025. 2