Plug-and-Play Traffic Element Awareness for End-to-End Autonomous Driving

Zongzheng Zhang[1*], Jijun Wang[1*], Saining Zhang[1], Wang Shuo[2], Yiru Wang[2], Hai Yang[2], Yang Chen[2], Yuwen Heng[2], Hao Sun[2], Anqing Jiang[2], and Hao Zhao[1†]

[1] Institute for AI Industry Research (AIR), Tsinghua University, Beijing, China
[2] Bosch Corporate Research, Shanghai, China
[*]Equal contribution. [†]Corresponding author.
https://zzongzheng0918.github.io/TE-Aware-E2E-AD/

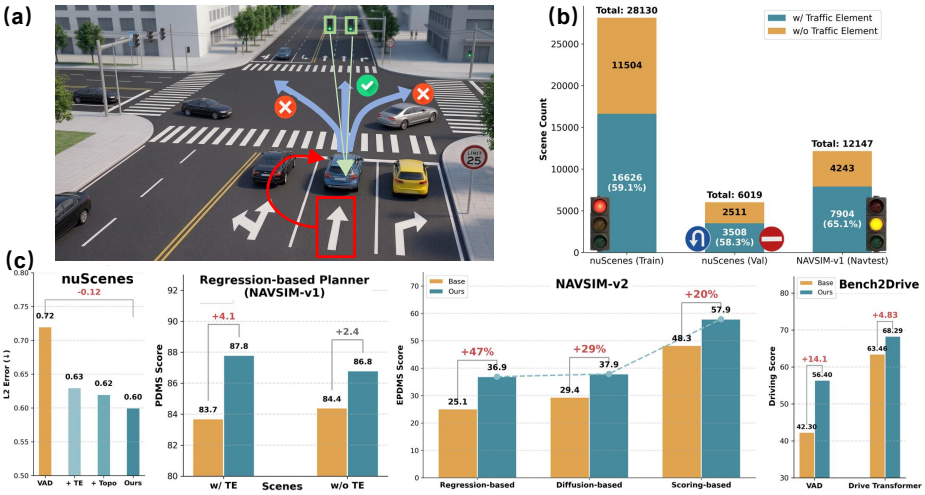

Fig. 1: TE-Aware End-to-End Planning. (a) In complex intersections, correct planning requires recognizing rule-critical traffic elements to avoid topologically plausible but rule-violating maneuvers. (b) Traffic elements are **pervasive**: across three benchmarks, over 55% of scenes contain at least one traffic element. (c) Leveraging traffic elements (and topology) yields **consistent improvements** across representative end-to-end planners in both open-loop and closed-loop evaluations.

Abstract. Traffic elements such as traffic lights and road signs play a fundamental role in human driving decisions and should naturally influence end-to-end driving performance. However, existing end-to-end driving research predominantly focuses on dynamic road participants (e.g., vehicles and pedestrians), while the role of traffic elements remains largely unexplored. To date, the community lacks a systematic study quantifying how traffic elements affect end-to-end driving models. This gap stems from two main challenges: first, existing public datasets rarely provide structured annotations for traffic elements; second, modern end-to-end driving systems vary widely in architectures and training paradigms, making conclusions drawn from a single method difficult to generalize. In this work, we present the first systematic investigation

of **traffic element awareness** for end-to-end autonomous driving. We begin by constructing a unified research infrastructure by augmenting multiple public driving datasets with comprehensive traffic element annotations. To ensure broad applicability across diverse model families, we intentionally adopt a minimal and universal integration design, allowing traffic element signals to be incorporated into existing pipelines in a **plug-and-play** manner with negligible architectural modification. We evaluate this design across a wide spectrum of modern paradigms, including perception–prediction-planning pipelines, vision-language-action models (VLA), regression-based planners, diffusion-based policies, and trajectory scoring frameworks. Experiments are conducted on multiple widely used benchmarks, including nuScenes, NAVSIM-v1, NAVSIM-v2, and Bench2Drive. Across all paradigms and datasets, our simple integration consistently improves driving performance, demonstrating that traffic element awareness provides a robust and generalizable signal for end-to-end driving systems. Notably, on the challenging NAVSIM-v2 benchmark, our approach significantly boosts the performance of state-of-the-art architectures and data pipelines, establishing a new state of the art.

Keywords: End-to-End Driving · Traffic Elements · Topology

1 Introduction

End-to-end (E2E) autonomous driving has rapidly moved from a research prototype to a deployable paradigm. Today's systems span a wide spectrum of architectural and training choices: ranging from perception–prediction–planning hybrids [26,34], to pure regression-based planners [9,31,67], to diffusion policies [48, 82, 93] and trajectory scoring frameworks [44, 50], and recently to VLM/VLA-style agentic planners [17, 69, 73, 92]. Despite this diversity, a surprisingly fundamental factor in human driving is still under-discussed in the E2E literature: traffic elements (TE), such as traffic lights and road signs. From first principles, TE are not *optional context*: they are regulatory signals that directly constrain the feasible action space (e.g., stop/go, forbidden turns, lane-level permissions) and thus should causally shape planning decisions. Yet, mainstream E2E benchmarks and methods overwhelmingly center their perception objectives and representations on dynamic agents (vehicles and pedestrians) and dense geometric cues, leaving the role of TE largely unquantified. The community does not have a systematic answer to a simple question: *how much does traffic-element awareness actually matter for E2E driving performance, across modern paradigms?*

Fig. 1 motivates why this question cannot be dismissed as a corner case. Fig. 1a illustrates a typical intersection where multiple signals and signs jointly determine what actions are legal and safe. Meanwhile, TE are prevalent rather than long-tailed. As shown in Fig. 1b, a majority of scenes in widely used datasets contain traffic elements: nuScenes [2] includes TE in 59.1% of training scenes (16,626 / 28,130) and 58.3% of validation scenes (3,508 / 6,019), while NAVSIM-v1 (`navtest`) [11] contains TE in 65.1% of scenes (7,904 / 12,147). This preva-

lence means that even within academic benchmarks, TE are not a rare special case; and in real-world deployment, E2E systems inevitably interact with traffic lights/signs continuously. The lack of systematic TE studies is therefore not due to irrelevance, but rather due to two practical barriers: (i) missing structured TE annotations in most public datasets [2, 3, 11], and (ii) the heterogeneity of E2E driving systems, where conclusions drawn from a single architecture or training recipe are difficult to generalize.

To close this gap, we present the first systematic investigation of **plug-and-play traffic element awareness for E2E autonomous driving**. Our goal is deliberately not to propose yet another highly customized planner, but to build a unified research infrastructure and derive generalizable conclusions that hold across model families. Concretely, we (1) augment multiple mainstream driving benchmarks with comprehensive TE annotations; (2) design a minimal universal integration mechanism, implemented as a lightweight auxiliary 3D traffic element supervision (and optional topology conditioning when available [71]), so that TE signals can be incorporated into existing pipelines in a plug-and-play manner with negligible architectural modifications; and (3) evaluate this integration across a broad set of representative paradigms, including regression-based planners, perception–planning pipelines, diffusion-based policies, trajectory scoring frameworks, and VLM/VLA-style models, on nuScenes [2], NAVSIM-v1 [11], NAVSIM-v2 [3], and the closed-loop Bench2Drive [30] benchmark.

Our experiments reveal a consistent and practically important conclusion: traffic element awareness yields stable improvements across datasets and paradigms, even under a minimal integration design. As previewed in Fig. 1c, adding TE supervision reduces open-loop trajectory error on nuScenes, boosts NAVSIM performance for all three paradigm planners, and improves closed-loop driving metrics on Bench2Drive. Interestingly, on the highly challenging NAVSIM-v2 benchmark, TE awareness produces large, consistent gains across representative families (regression / diffusion / scoring), and remains effective even when combined with strong modern data engines [68] (e.g., pairing a state-of-the-art scoring-based backbone with large-scale simulated co-training data) continues to improve performance and establishes a new state of the art in our evaluation.

Beyond aggregate gains, we further conduct targeted ablations to explain why TE supervision works and how to integrate it robustly. We find that (i) explicit 3D TE representations outperform 2D-only cues for planning; (ii) selectively modeling TE depth is more effective than feeding global depth maps, suggesting TE distills the most decision-relevant spatial cues; (iii) traffic signs (not only lights) are crucial and often overlooked; and (iv) for sparse TE targets, an independent prediction head, focal loss, and max-preserving pooling are key to avoiding gradient domination by dense background. When lane topology is available, we show that encoding ego-relevant topology into a compact conditioning signal (language-style embeddings), which is more effective than GNN-based graph encoding [61], complements local TE cues. Together, these results argue that TE awareness is a broadly useful and underutilized signal for E2E driving.

2 Related Work

2.1 End-to-End Autonomous Driving

End-to-end autonomous driving methods can be broadly categorized into several paradigms according to how driving decisions are produced. Perception–planning methods [5, 19, 25, 26, 28, 29, 34, 36, 65, 67] construct intermediate scene representations (e.g.,vectorized map) and then perform motion planning based on them. Regression-based methods directly predict trajectories or control commands from sensor observations, including [9,21,31,47,76,77,81,84]. Trajectory-scoring methods [44–46,63,83] generate multiple candidate trajectories and select the optimal one through a scoring module. Generative planning methods [48,51, 52,66,72,82,90,93], including diffusion/flow-based approaches, model trajectory distributions with generative models. Recently, vision–language–action (VLA) methods [6,8,13,27,32,33,42,64,73,75,92] integrate visual perception, language understanding, and driving actions in a unified framework. In addition, world-model-based approaches [37, 41, 62, 91] learn latent environment dynamics for decision making. We evaluate representative methods from these categories to verify the plug-and-play generality of our approach.

2.2 Traffic-Aware Scene Understanding

Prior work injects traffic-aware scene understanding into end-to-end planning via structured intermediates such as occupancy [26, 70, 76], HD map [25, 34, 72, 88–90], 3D detection-tracking [31, 65, 67], to enforce safety and rule compliance. VLM-based planners [69, 73, 92] trained on specific VQA datasets [22, 54, 56, 58, 64, 79] provide higher-level semantic reasoning but at substantial runtime cost. Dedicated benchmarks like Openlane-V2 [4, 71] evaluate these scene-understanding components, but most topology-predicted methods [18, 38–40, 53, 78, 87] primarily optimize benchmark scores without validating downstream planning impact.

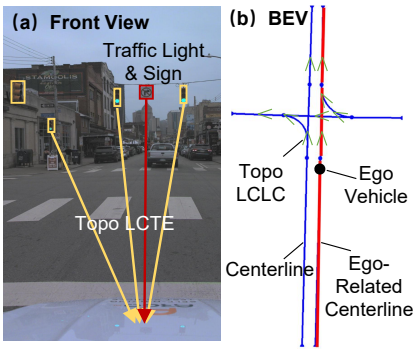

Fig. 2: Definitions of driving topologies. (a) LCTE topology mapping traffic elements (light/sign) to the ego-lane in FV. (b) BEV representation of centerlines, ego-related centerlines, and LCLC topology.

We provide the first systematic evaluation of how such cues transfer to planning, and propose a lightweight alternative based on explicit traffic elements and lane topology that is more amenable to deployment.

3 Preliminaries: OpenLane-V2 Scene Representation

Components. Openlane-V2 [71] has four components: **Centerlines:** A set of M 3D lane centerlines (Fig. 2b); **Traffic Elements (TE):** A set of N entities (4

traffic light states and 9 traffic sign categories), parameterized as 2D bounding boxes in the image coordinate system (Fig. 2a); **LCTE Topology:** The Lane-Centerline to Traffic-Element relationship, formulated as a binary adjacency matrix $\mathbf{R}_{\text{LCTE}} \in \{0,1\}^{M\times N}$, where an entry is 1 if the j-th TE directly governs the i-th centerline (Fig. 2a); **LCLC Topology:** The Centerline to Centerline directed connectivity, formulated as an adjacency matrix $\mathbf{R}_{\text{LCLC}} \in \{0,1\}^{M\times M}$, defines the valid navigable transitions between lanes (Fig. 2b).

Topology Matters. Scene representation is crucial because it makes the driving constraints explicit. In this intersection (Fig. 2), a no-right-turn sign prohibits right-turn maneuvers, the LCLC topology indicates that the ego lane has no lateral connectivity, and the green traffic light permits forward motion. Together, these cues constrain the feasible action space: the ego vehicle should proceed into the adjacent forward lane at an appropriate speed, rather than turning left or right—a failure mode that vision-only end-to-end planners always exhibit.

4 Method

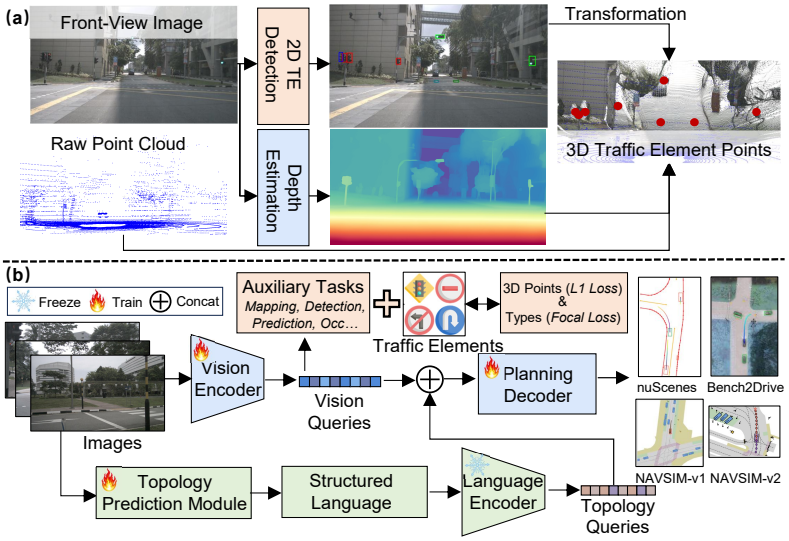

Fig. 3: (a) **Pipeline for 3D traffic element extraction.** We first perform 2D traffic-element detection and monocular depth estimation on the FV image, then combine them and LiDAR geometry to localize each traffic element as a 3D center point (•). (b) **Overview of our method.** Multi-view images are encoded into vision queries and enhanced by an auxiliary 3D traffic element detection task. In parallel, predicted topological adjacency matrices are formatted into structured language and processed by a frozen language encoder. The resulting topology queries are concatenated with the BEV queries to guide the planning decoder in generating the ego-vehicle trajectory.

4.1 3D Dataset Construction

To effectively integrate traffic elements into the 3D planning space, we establish a robust pipeline to extract their 3D coordinates (x, y, z) (Fig. 3a). Given a

front-view image, we process it through two parallel branches: a 2D TE detection model to obtain bounding boxes, and a monocular depth foundation model [57] to estimate dense depth maps. Using the camera intrinsic and extrinsic parameters, we project the 2D bounding boxes into the LiDAR coordinate system. The precise 3D center point (x, y, z) of each TE is then jointly determined by aggregating the 2D box constraints, the estimated depth, and the corresponding LiDAR point cloud (details in the Appendix). We adapt this pipeline across different benchmarks based on their available annotations: **nuScenes [2] & Bench2Drive [30]:** For nuScenes, we directly utilize the 2D TE bounding boxes and topology annotations provided by OpenLane-V2, and apply our extraction pipeline to acquire 3D centers. The Bench2Drive dataset natively provides detailed 3D TE coordinates and topological relationships. **NAVSIM (v1 [11] & v2 [3]):** Since these datasets lack TE annotations, we first train a highly accurate YOLO-based [60] 2D TE detector on OpenLane-V2, adopting progressive training strategies (e.g., resampling and reweighting, pseudo labeling) from [20, 80]. We then deploy this detector alongside our depth-LiDAR fusion pipeline to automatically construct the 3D TE pseudo-labels for NAVSIM.

4.2 Auxiliary 3D Traffic Element Supervision

In typical end-to-end autonomous driving frameworks, multi-view images are encoded [7, 14, 16, 23, 55, 74] into latent vision queries (e.g., BEV queries $\mathbf{Q}_{\text{bev}}$), which are then used for downstream planning (Fig. 3b). Building on the original auxiliary tasks of each planner (e.g., detection of 3D vehicle boxes, prediction, tracking, occupancy, etc), we introduce an additional 3D traffic element detection objective. Specifically, for each TE, we supervise its 3D center location with an L_1 loss and its category with a focal loss. This TE loss is aggregated into the overall training objective with a weight equivalent to other auxiliary tasks:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{plan}} + \sum_{k \in \mathcal{K}} \lambda_k \mathcal{L}_{\text{aux}}^{(k)} + \lambda_{\text{TE}} \big(\mathcal{L}_{\text{L1}}^{\text{loc}} + \mathcal{L}_{\text{focal}}^{\text{cls}} \big), \tag{1}$$

where $\mathcal{L}_{\text{plan}}$ is the primary trajectory planning loss, $\mathcal{L}_{\text{aux}}$ includes all original auxiliary losses of the baseline planner, and λ_{TE} is set identically to $\mathcal{L}_{\text{aux}}$.

4.3 Language-Guided Topology Conditioning

In parallel with TE-aware BEV learning, a topology prediction module estimates the global adjacency matrices for the entire driving scene, denoted as $\mathbf{R}_{\text{LCLC}}$ and $\mathbf{R}_{\text{LCTE}}$ (Fig. 3b). To prevent the planner from being distracted by redundant distant elements, we perform an **ego-centric topology filtering**. With the ego vehicle's spatial coordinates, we identify its lane ID. We then query $\mathbf{R}_{\text{LCLC}}$ to retrieve the ego vehicle-related centerlines (highlighted in red, Fig. 2b) and query $\mathbf{R}_{\text{LCTE}}$ to isolate the traffic elements directly governing this active lane (Fig. 2a). We convert this ego-centric topology graph into a structured language sequence, *e.g., "There are a 'green traffic_light', a 'green traffic_light', a 'green traffic_light', a*

'no_right_turn road_sign' ahead controlling the current ego-lane, which connects '1' 'straight' centerline." Here, parameters such as the traffic element categories, the number of connected centerlines ('1'), and their directional semantics ('straight', 'left') are dynamically instantiated based on the retrieved local topology.

This text is encoded by a pretrained BERT-base [12] language encoder (∼110M parameters) to obtain topology queries $\mathbf{Q}_{\text{topo}}$. Finally, we concatenate $\mathbf{Q}_{\text{topo}}$ with the TE-enhanced vision/BEV queries $\mathbf{Q}_{\text{bev}}$, and feed the joint queries $\mathbf{Q}_{\text{plan}} = [\mathbf{Q}_{\text{bev}}; \mathbf{Q}_{\text{topo}}]$ into the planning decoder. This design injects lane connectivity and rule-control information into the planner in a compact form, enabling trajectory prediction to respect both geometric feasibility and traffic regulations.

We validate our method on **six** representative end-to-end methods across **four** benchmarks. Implementation details for integrating our TE supervision and topology conditioning into each specific backbone are provided in the Appendix.

5 Experiment

We first specify the experimental protocol (Sec. 5.1). We then report quantitative (Sec. 5.2) and qualitative comparisons (Sec. 5.3) to assess the planning performance of our approach across benchmarks. Finally, we conduct targeted ablation studies to isolate the contribution of each component and design choice, and to explain why the proposed mechanism yields consistent gains (Sec. 5.4).

5.1 Experimental Setup

Datasets and Metrics. We evaluate on **four datasets**: **three open-loop** benchmarks (nuScenes [2], NAVSIM-v1 [11], and NAVSIM-v2 [3]), and **one closed-loop** benchmark (Bench2Drive [30]). NuScenes provides large-scale real-world logs for open-loop ego-trajectory prediction. NAVSIM-v1 (`navtest`) provides a non-reactive, data-driven evaluation setting with standardized simulation-based scoring. NAVSIM-v2 (`navhard`) further introduces a two-stage pseudo closed-loop aggregation to better reflect compounding-error effects without requiring full closed-loop simulation. Bench2Drive evaluates end-to-end planners in closed loop in CARLA [15] across a large set of short, scenario-isolated routes.

For nuScenes, we report **L2 trajectory error** and **Collision Rate** under the standard open-loop protocol. For NAVSIM-v1, we use the official **PDMS** as the primary score. For NAVSIM-v2, we report the aggregate **EPDMS** along with the per-stage breakdown (Stage 1/Stage 2) over the core compliance/safety terms. For Bench2Drive, we report **Driving Score** as the main closed-loop indicators, and additionally include Success Rate, Efficiency, Comfortness, and the open-loop Avg. L2. Detailed metric definitions are provided in the Appendix.

Training Details. We follow the original training schedules and hyperparameters of each baseline (details in the Appendix). On nuScenes, we train on 8×A100 with batch size 1. On NAVSIM, we train on 4×RTX 3090 with batch size 32. On Bench2Drive, we train on 8×A100 with batch size 1.

5.2 Quantitative Results

Table 1: Comparison of methods on the nuScenes dataset. ◇: Lidar-based methods. †: The ego status is utilized. FPS is measured on an NVIDIA A100 GPU.

Method	Auxiliary Task	L2 (m) ↓				Collision Rate (%) ↓				FPS↑
		1s	2s	3s	Avg.	1s	2s	3s	Avg.	
NMP◇ [85]	Det & Motion	0.53	1.25	2.67	1.48	0.04	0.12	0.87	0.34	-
FF◇ [24]	FreeSpace	0.55	1.20	2.54	1.43	0.06	0.17	1.07	0.43	-
EO◇ [35]	FreeSpace	0.67	1.36	2.78	1.60	0.04	0.09	0.88	0.33	-
ST-P3 [25]	Det & Map & Depth	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71	1.6
UniAD [26]	Det&Track&Map&Motion&Occ	0.44	0.67	0.96	0.69	0.04	0.08	0.23	0.12	1.8
VAD-Tiny [34]	Det & Map & Motion	0.46	0.76	1.12	0.78	0.21	0.35	0.58	0.38	16.8
BEV-Planner [47]	None	0.28	0.42	0.68	0.46	0.04	0.37	1.07	0.49	-
PARA-Drive [76]	Det&Track&Map&Motion&Occ	0.25	0.46	0.74	0.48	0.14	0.23	0.39	0.25	5.0
LAW [41]	None	0.26	0.57	1.01	0.61	0.14	0.21	0.54	0.30	19.5
GenAD [90]	Det & Map & Motion	0.28	0.49	0.78	0.52	0.08	0.14	0.34	0.19	6.7
SparseDrive [67]	Det & Track & Map & Motion	0.29	0.58	0.96	0.61	0.01	0.05	0.18	0.08	9.0
UAD [21]	Det	0.28	0.41	0.65	0.45	0.01	**0.03**	0.14	0.06	7.2
MomAD [65]	Det & Track & Map & Motion	0.31	0.57	0.91	0.60	0.01	0.05	0.22	0.09	7.8
VAD-Base [34]	Det & Map & Motion	0.41	0.70	1.05	0.72	0.07	0.17	0.41	0.22	5.7
VAD + TE	Det & Map & Motion	0.36	0.61	0.92	0.63	0.09	0.14	0.28	**0.17**	5.7
VAD + Topo	Det & Map & Motion & Topo	0.34	0.59	0.92	0.62	0.05	0.15	0.29	0.16	5.4
VAD + Ours	Det & Map & Motion & Topo	0.34	0.56	0.90	**0.60**	0.04	0.21	0.26	**0.17**	5.4
VLM/VLA-based Method										
Qwen-2.5-VL [1]	Scene Understanding	0.46	1.33	2.55	1.45	–	–	–	–	0.8
Senna [33]	Det & Motion	0.37	0.54	0.86	0.59	0.09	0.12	0.33	**0.18**	1.6
DriveVLM [69]	Scene Understanding	0.18	0.34	0.68	0.40	0.10	0.22	0.45	0.27	–
OmniDrive [73]	Scene Understanding	0.14	0.29	0.55	0.33	0.00	0.13	0.78	0.30	1.2
EMMA [27]	Scene Understanding	0.14	0.29	0.54	0.32	–	–	–	–	–
ImpromptuVLA [8]	Scene Understanding	0.13	0.27	0.53	**0.30**	–	–	–	–	0.9
Orion† [17]	Det & Motion	0.17	0.31	0.55	0.34	0.05	0.25	0.80	0.37	0.9
Orion†+ TE	Det & Motion	0.14	0.27	0.47	0.29	0.06	0.20	0.55	0.27	0.9
Orion†+ Topo	Det & Motion & Topo	0.12	0.25	0.45	0.27	0.04	0.21	0.50	0.25	0.8
Orion†+ Ours	Det & Motion & Topo	0.11	0.25	0.43	**0.26**	0.04	0.19	0.47	**0.23**	0.8

NuScenes. We construct 3D traffic elements using the OpenLane-V2 subset annotated on nuScenes, and use the same detection-aligned perception range as prior work [34]: lateral $[-15, 15]$ m and longitudinal $[0, 30]$ m. Depth is estimated by UniDepthv2 [57] with frozen weights. We encode topological cues by first predicting a topology adjacency matrix with a lightweight TopoMLP [78], and then mapping it to a language embedding via a frozen BERT encoder [12].

We compare against two representative families of end-to-end planners: the conventional perception-planning baseline VAD [34] and the VLM-based planner Orion [17]. Tab. 1 shows that introducing traffic elements as auxiliary supervision improves both L2 ($\downarrow$0.09; $\downarrow$0.05) and CR ($\downarrow$0.05%; $\downarrow$0.10%) over the baselines. We attribute this gain to the fact that the additional TE objective makes the learned BEV queries more **traffic-aware**: latent vision queries are typically dominated by dense scene geometry and dynamic agents (e.g., via standard occupancy or vehicle detection tasks), which inherently overshadow spatially sparse yet logically critical traffic elements. We force the network to allocate dedicated represen-

Table 2: Camera-only methods on the NAVSIM-v1 benchmark (navtest).

Method	Backbone	Venue	NC ↑	DAC ↑	TTC ↑	Comf. ↑	EP ↑	**PDMS ↑**
PDM-Closed [10]	–	PMLR'23	94.6	99.8	89.9	86.9	99.9	89.1
Human driver [11]	–	NeurIPS'24	100	100	100	99.9	87.5	**94.8**
Ego-stat. MLP [11]	–	NeurIPS'24	93.0	77.3	83.6	100	62.8	65.6
UniAD [26]	ResNet34	CVPR'23	97.8	91.9	92.9	100	78.8	83.4
VAD-v2 [5]	ResNet34	ICLR'26	98.1	94.8	94.3	100	80.6	86.2
ReCogDrive [43]	InternViT	ICLR'26	97.9	97.3	94.9	100	87.3	90.8
Hydra-MDP [44]	ResNet34	arXiv'24	98.3	96.0	94.6	100	78.7	86.5
Centaur [63]	ResNet34	arXiv'25	99.5	98.9	98.0	100	85.9	**92.6**
DriveSuprim [83]	ResNet34	AAAI'26	97.8	97.3	93.6	100	86.7	89.9
Regression-based Planner								
LTF [9]	ResNet34	TPAMI'22	97.8	92.8	93.3	100	78.9	84.1
LTF + Ours	ResNet34	ECCV'26	97.8	93.9	93.8	100	79.8	85.2 +1.1
LTF (+SimScale [68])	ResNet34	ECCV'26	98.3	95.6	94.6	100	81.3	87.3 +3.2
LTF (+SimScale) + Ours	ResNet34	ECCV'26	98.2	95.9	94.5	100	82.0	**87.6** +3.5
Diffusion-based Planner								
DiffusionDrive [48]	ResNet34	CVPR'25	97.9	94.6	93.6	100	80.7	86.0
DiffusionDrive + Ours	ResNet34	ECCV'26	98.1	96.0	94.2	100	82.3	87.7 +1.7
DiffusionDrive (+SimScale [68])	ResNet34	ECCV'26	98.5	97.0	94.7	100	83.1	88.9 +2.9
DiffusionDrive (+SimScale) + Ours	ResNet34	ECCV'26	98.6	97.2	94.7	100	83.5	**89.1** +3.1
Scoring-based Planner								
DrivoR [36]	ViT-S	CVPR'26	98.9	98.3	96.2	100	89.1	93.1
DrivoR + Ours	ViT-S	ECCV'26	99.0	98.7	96.9	100	90.8	94.4 +1.3
DrivoR (+SimScale [68])	ViT-S	ECCV'26	99.1	99.2	96.9	100	91.6	94.6 +1.5
DrivoR (+SimScale) + Ours	ViT-S	ECCV'26	99.7	99.7	98.1	100	92.2	**95.1** +2.0

tational capacity to these vital regulatory signals. Consequently, this prevents traffic rules from being marginalized in the shared feature space.

Furthermore, injecting topological cues yields additional gains, indicating that high-level connectivity/route structure complements local rule signals. The best performance is obtained when both traffic elements and topology are incorporated. Crucially, these scene-understanding signals are predicted by lightweight networks, so the runtime impact is small (FPS drops only 0.3). Compared to VLM-based scene understanding methods [1, 8, 69], this design delivers $> 10\times$ higher throughput while still capturing the key cues that drive performance, making the approach substantially more amenable to real-time deployment.

NAVSIM. Since NAVSIM does not provide traffic element annotations, we generate pseudo-labels by running our OpenLaneV2-trained TE detector on the full front-view image, and supervise a BEV TE heatmap head with focal loss (α=2, β=4). We adopt the NAVSIM perception range lateral $[-32, 32]$ m and longitudinal $[0, 32]$ m, and estimate depth using UniDepthv2 as in nuScenes. Due to the different camera setup in NAVSIM compared to nuScenes, we do not train a topology predictor in this setting; instead, we focus on TE only and feed the predicted TE representation to the decoder to guide planning.

On NAVSIM-v1, we evaluate **three** representative end-to-end camera-only planning paradigms: the regression-based planner LTF [9], the diffusion-based planner DiffusionDrive [48], and scoring-based planners DrivoR [36]. Tab. 2 shows that adding traffic elements as explicit supervision and conditioning the

Table 3: Comparison to existing methods on the **NAVSIM-v2** `navhard-two-stage` benchmark using EPDMS. †: Metrics reported prior to the official benchmark bug fix.

Method	Stage 1									Stage 2									**EPDMS** ↑
	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	
ZTRS† [45]	98.9	97.6	100	100	66.7	98.9	96.2	96.7	44.0	91.1	90.4	95.8	99.0	63.6	89.8	60.4	97.6	66.1	45.5
DiffVLA† [32]	95.7	99.2	100	100	85.9	96.4	97.1	95	84.2	81.2	88.8	94.6	99.0	86.0	76.4	59.8	98.6	80.4	45.0
GTRS-Dense† [46]	98.9	94.9	99.1	100	76.1	98.4	93.8	94.9	37.8	89.9	90.5	94.1	99.3	77.6	88.5	56.0	92.0	30.2	41.9
GuideFlow† [51]	97.8	97.1	100	100	81.4	98.5	91.4	92.8	34.2	87.3	92.3	98.0	96.9	75.8	85.5	59.3	95.4	53.5	**46.7**
Regression-based Planner																			
LTF [9]	96.2	79.5	99.1	99.5	84.1	95.1	94.2	97.5	79.1	77.7	70.2	84.2	98.0	85.1	75.6	45.4	95.7	75.9	25.1
LTF + Ours	96.4	79.8	98.9	99.6	84.2	95.8	93.8	97.5	78.2	80.9	71.6	84.8	98.9	85.2	77.7	47.0	96.1	76.3	28.9 ↑15%
LTF(+SimScale [68])	96.1	85.3	99.4	99.3	84.7	94.7	93.5	97.5	77.3	85.5	66.9	91.5	99.1	93.0	81.1	58.2	95.1	42.9	33.6 ↑33%
LTF(+SimScale)+Ours	97.0	85.3	99.7	99.6	84.2	96.2	96.0	97.6	77.3	88.1	71.8	93.1	99.0	86.7	83.0	54.0	95.2	55.4	**36.9** ↑47%
Diffusion-based Planner																			
DiffusionDrive [48]	96.7	86.7	98.7	99.3	84.3	94.9	95.3	97.6	77.8	78.9	72.4	84.2	98.2	87.1	74.9	47.3	96.2	71.0	29.4
DiffusionDrive + Ours	97.1	86.9	98.8	99.6	84.2	95.1	95.6	97.6	79.5	80.2	74.0	85.0	98.1	86.3	77.2	48.4	96.6	74.4	32.7 ↑11%
DiffusionDrive(+SimScale [68])	97.2	88.0	99.2	99.3	82.8	96.7	98	97.5	58.2	86.7	72.1	93.0	98.8	92.1	80.6	61.1	95.3	33.1	35.8 ↑12%
DiffusionDrive(+SimScale)+Ours	97.1	88.4	99.3	99.3	84.1	95.6	98.2	97.6	72.9	83.7	76.3	92.5	98.9	90.3	78.9	57.7	94.4	56.1	**37.9** ↑29%
Scoring-based Planner																			
DrivoR [36]	98.8	95.1	98.9	100	72.6	98.7	94.0	97.6	73.3	90.2	88.4	91.9	98.6	70.0	88.0	50.1	98.5	76.2	48.3
DrivoR + Ours	98.9	94.9	99.1	100	75.1	98.9	93.9	97.5	74.4	91.9	91.4	96.1	99.3	74.6	87.5	56.0	97.0	78.5	51.8 ↑7.2%
DrivoR(+SimScale [68])	99.1	98.2	99.3	99.8	75.4	98.7	94.9	97.6	70.2	92.3	91.6	97.3	99.1	75.7	90.6	56.1	98.4	44.7	54.6 ↑13%
DrivoR(+SimScale)+Ours	99.5	98.2	99.3	100	76.1	99.0	94.1	97.9	72.6	93.5	91.8	97.0	99.3	76.6	90.9	56.0	98.0	55.6	**57.9** ↑20%

planner on the predicted TE representation yield consistent gains across all paradigms. The improvements are primarily driven by higher NC (No Collision) and DAC (Drivable Area Compliance)—two key safety/compliance factors that carry substantial weight in the overall PDMS—highlighting the importance of traffic-element understanding for reliable planning. To test **data scalability**, we further leverage SimScale [68] to generate simulated data and co-train using traffic-element cues extracted from both simulated and real logs. This co-training strategy leads to a further increase in PDMS, indicating that our supervision signal transfers effectively across domains and benefits from larger-scale training. Notably, with DrivoR, our approach surpasses both the rule-based method [10] and the human-driver score [11], suggesting that **stronger backbones** and **more training data** can unlock additional gains.

We further validate these findings on NAVSIM-v2 with same representative planners, and observe the same consistent trend (Tab. 3). Across methods, incorporating traffic elements improves the overall score by roughly **+10 EPDMS**. The gains are mainly attributed to stronger rule compliance and safety, reflected by improvements in the core terms NC, DAC, DDC, and TLC—suggesting that TE supervision helps the model internalize traffic rules. We also observe that adding SimScale data can noticeably reduce EC (Extended Comfort), likely because some counterfactual simulated scenarios introduce distribution shifts that affect ride quality. Importantly, combining SimScale with our TE-centric training mitigates this effect and brings EC back to a higher level. Overall, these results demonstrate that our method improves safety-critical behavior and exhibits **strong scalability** with **model** and **data** scale.

Bench2Drive. Bench2Drive provides instance-level traffic elements with 3D locations, as well as an HD-map lane topology that encodes lane-level topology and connectivity. We therefore supervise our traffic-element head directly using the provided 3D TE coordinates and encode map topology in the same manner as in nuScenes by mapping topology cues to a compact embedding that conditions the planner. Traffic elements used by the planner are obtained from our model pre-

Table 4: Closed-Loop Planning Performance in Bench2Drive. Avg. L2 is averaged over the predictions in 2 seconds. ∗: denotes expert feature distillation. All latency is measured by the averaged inference step-time on CARLA [15] evaluation in A6000.

Method	Open-loop Metric	Closed-loop Metric				Latency
	Avg. L2 ↓	Driving Score ↑	Success Rate(%) ↑	Efficiency ↑	Comfortness ↑	
TCP∗ [81]	1.70	40.70	15.00	54.26	47.80	**86ms**
TCP-ctrl∗	-	30.47	7.27	55.97	**51.51**	**86ms**
TCP-traj∗	1.70	59.90	30.00	76.54	18.08	**86ms**
TCP-traj w/o distillation	1.96	49.30	20.45	**78.78**	22.96	**86ms**
ThinkTwice∗ [29]	**0.95**	62.44	31.23	69.33	16.22	698ms
DriveAdapter∗ [28]	1.01	**64.22**	**33.08**	70.22	16.01	958ms
AD-MLP [86]	3.64	18.05	0.00	48.45	22.63	**4.7ms**
UniAD-Tiny [26]	0.80	40.73	13.18	123.92	**47.04**	400.3ms
UniAD-Base [26]	0.73	45.81	16.36	**129.21**	43.58	692.6ms
VAD [34]	0.91	42.3	15.00	**157.94**	46.01	278.3ms
VAD + Ours	**0.75** -0.16	**56.4** +14.1	**21.30**	125.64	**48.02**	282.5ms
DriveTransformer-Large [31]	0.62	63.46	35.01	**100.64**	20.78	211.7ms
DriveTransformer-Large + Ours	**0.57** -0.05	**68.29** +4.83	**39.61**	82.45	**23.58**	216.5ms

dictions at inference time, rather than from reading annotation states. We run closed-loop planning using two baselines (VAD [34] and DriveTransformer [31]) at 2 Hz. Tab. 4 shows consistent improvements for both methods in L2 and the main closed-loop metrics (notably Driving Score), with a small change in latency. Interestingly, Efficiency decreases after adding our TE supervision and topology. We interpret this as a correction of over-aggressive behaviors encouraged by efficiency-dominated objectives: baseline planners can achieve high efficiency while under-modeling rule-critical traffic elements, whereas TE awareness biases the policy toward safer and more compliant maneuvering.

5.3 Qualitative Results

Fig. 4 shows a challenging NAVSIM-v2 intersection. Without traffic-element awareness, **LTF** and **LTF+SimScale** exhibit noticeable lateral drift (Fig. 4a-b). In contrast, **Ours** detects the green traffic lights and the straight-ahead sign for the ego lane, and thus continues forward while staying lane-consistent (Fig. 4c). This example highlights that traffic elements provide decision-critical cues that stabilize planning in complex junctions (see Appendix for more visualization).

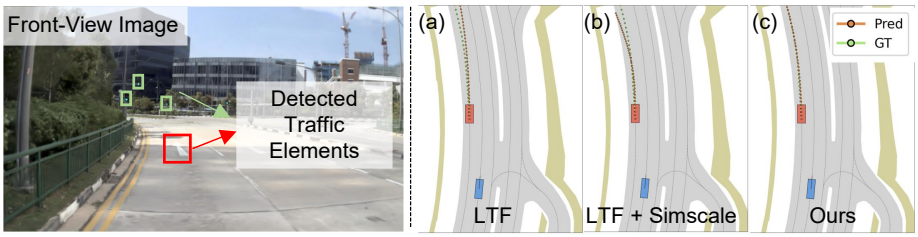

Fig. 4: Qualitative comparison on **NAVSIM-v2**. **Left:** front-view image with detected traffic elements. **Right:** predicted trajectories of LTF, LTF+SimScale, and Ours.

5.4 Ablation Study and Analysis

Table 5: Ablation study of **traffic element (TE) representations** on the **NAVSIM-v2** using **LTF** (Latent Transfuser). **TE(2D)** denotes 2D traffic elements in the front-view (**FV**) image; **depth(FV)** uses the full FV depth; **LiDAR** uses the NAVSIM-v2 point-cloud input; **TL** denotes traffic lights (green/yellow/red light) only.

Method	Stage 1									Stage 2									EPDMS ↑
	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	
LTF [9]	96.2	79.5	99.1	99.5	84.1	95.1	94.2	97.5	79.1	77.7	70.2	84.2	98.0	85.1	75.6	45.4	95.7	75.9	25.1
LTF+TE(2D)	96.4	82.0	99.2	99.6	84.0	95.8	93.1	97.8	79.1	81.6	68.4	85.5	98.6	85.4	78.4	46.9	96.2	76.5	28.1$_{+3.0}$
LTF+depth(FV)	96.9	81.8	99.0	99.6	84.2	95.8	92.7	97.8	78.2	81.3	70.1	84.3	98.6	85.7	78.4	45.9	96.3	76.5	27.7$_{+2.6}$
LTF+TE(Lidar)	97.2	80.9	99.0	99.5	83.9	96.0	92.9	97.8	79.5	81.1	70.2	85.4	98.6	85.0	79.1	45.9	96.0	77.2	27.9$_{+2.8}$
LTF+TE [49]	96.7	81.8	99.1	99.3	84.1	95.6	92.4	97.6	78.2	81.2	69.4	84.9	98.6	85.6	78.3	46.6	96.2	76.0	27.8$_{+2.7}$
LTF+TL [57]	97.1	80.7	99.1	99.3	84.1	95.3	93.6	97.8	78.2	80.4	69.5	84.8	98.6	85.7	77.6	45.0	96.3	76.4	26.7$_{+1.6}$
LTF+Ours	96.4	79.8	98.9	99.6	84.2	95.8	93.8	97.5	78.2	80.9	71.6	84.8	98.85	85.2	77.7	47.0	96.1	76.3	**28.9**$_{+3.8}$

Ablation on Traffic-Element Representations for Planning. To dissect the efficacy of our proposed 3D traffic element (TE) representation, we conduct ablation studies on the NAVSIM-v2 [3] using the LTF baseline [9] (Tab. 5).

(1) **Explicit 3D vs. 2D Representations.** While 2D traffic elements (LTF+TE(2D)) provide basic semantic context, they inherently lack the depth and spatial precision crucial for motion planning. In contrast, our method explicitly constructs these elements within a 3D ego-centric space (28.1 vs. 28.9).

(2) **Targeted Depth vs. Global Depth.** Incorporating spatial auxiliary tasks generally improves the LTF baseline (25.1 EPDMS). However, simply feeding the full front-view depth to the model (LTF+depth(FV), +**2.6**) yields suboptimal gains compared to our explicit 3D TE formulation (LTF+Ours, +**3.8**). This indicates that unconstrained global depth may introduce redundant geometric noise, whereas selectively isolating and estimating the depth of specific traffic elements distills the most actionable spatial cues for the planner.

(3) **Vision-based Depth Estimation vs. LiDAR Clustering.** We compare our method against a geometry-heuristic approach (LTF+TE(Lidar)), which directly projects 2D bounding boxes into the 3D LiDAR coordinate system and extracts element centers via point clustering. Although helpful (+**2.8**), it still underperforms our approach (27.9 vs. 28.9). This validates our hypothesis that LiDAR returns on certain traffic elements—particularly flat road surface signs—are often too sparse or severely affected by dynamic occlusions.

(4) **The Impact of Depth Quality.** The accuracy of the underlying depth prior bottlenecks planning performance. When we replace our autonomous-driving-aligned depth estimator (UniDepthV2 [57]) with a general-purpose foundation model (Depth Anything 3 [49]), the EPDMS drops to 27.8.

(5) **The Crucial Role of Traffic Signs.** Modeling traffic lights only (LTF+TL) is insufficient. It lags behind our full TE setting (26.7 vs. 28.9), highlighting the often-overlooked importance of traffic signs in shaping planning decisions.

Design Choices for Traffic Element Integration. To validate the effectiveness of our proposed modules, we systematically ablate the design choices of the traffic element conditioning mechanism. Results are summarized in Tab. 6.

(1) **Prediction Head and Loss Function.** Comparing ID 1 and 2, decoupling TE prediction into an independent head yields a significant performance boost (+**1.8** EPDMS). Traffic elements are extremely sparse; treating them as an

Table 6: Ablation study of **traffic element integration** on the **NAVSIM-v2** using **LTF** (Latent Transfuser). We systematically investigate the design choices for the prediction head, loss function, spatial pooling strategy, and planning interaction method.

ID	TE Head		Loss		Pooling		Interaction		EPDMS ↑
	w/ BEV	Indep.	Cross-Entropy	Focal	Average	Max	C.A.	Concat	
0									25.1
1	✓		✓		✓				24.3$_{-0.8}$
2		✓		✓	✓				26.1$_{+1.0}$
3		✓	✓		✓				25.4$_{+0.3}$
4	✓		✓		✓		✓		24.9$_{-0.2}$
5		✓		✓		✓	✓		27.5$_{+2.4}$
6		✓		✓		✓		✓	**28.9**$_{+3.8}$

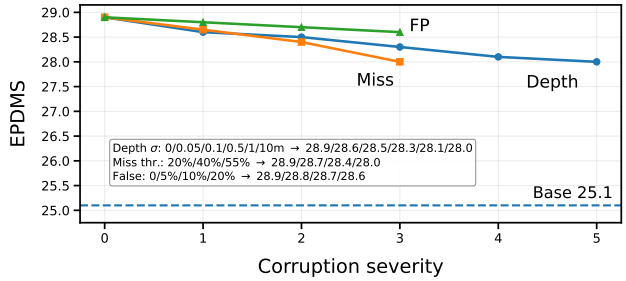

Fig. 5: Robustness analysis on NAVSIM-v2 with LTF. We corrupt the predicted TE representation at inference time by adding TE depth noise, randomly dropping TE detections, or injecting false-positive TE predictions. The performance decreases smoothly under stronger corruption while remaining above the LTF baseline, indicating robustness to moderate upstream TE perception noise.

additional BEV semantic class with standard cross-entropy (ID 1) even degrades performance (-0.8), as the TE gradients are dominated by the dense background. Moreover, replacing CE with focal loss under the independent head (ID 2 vs. ID 3) brings a further +**0.7** EPDMS, consistent with focal loss being better suited for the severe foreground–background imbalance of tiny TE targets.

(2) **Pooling and Interaction Mechanism.** Explicitly routing TE constraints to the planner via Cross-Attention (C.A.) improves the baseline (4 vs. 1, +**0.6**), proving the necessity of interactive planning. However, since critical signals occupy very few pixels, Average Pooling severely dilutes their activations during spatial downsampling. Upgrading to Adaptive Max Pooling (ID 5) guarantees these peak activations are perfectly preserved in the low-resolution grid. Our optimal configuration (ID 6) replaces C.A. by concatenating the max-pooled TE features directly into the BEV memory. This creates a strictly aligned "Dual Spatial Stream" that outperforms C.A. (+**1.4**). We attribute this to a more direct and spatially consistent conditioning signal, which helps the planner associate traffic rules with the correct ego-relevant lane context.

Robustness and semantic analysis. Since our framework uses predicted traffic elements at inference time, we further evaluate whether the planning gain is robust to upstream perception noise. We perform inference-stage sensitivity tests on NAVSIM-v2 with LTF by corrupting the predicted TE representation before it is fed to the planner. Specifically, we consider three common failure modes: noisy TE depth, missed TE detections, and false-positive TE predictions. As shown in Fig. 5, the planning score degrades smoothly as the corruption severity increases, but remains consistently above the original LTF baseline. This indicates that the proposed TE-aware planner does not rely on perfectly accurate TE predictions and is reasonably resilient to moderate upstream perception errors. We also verify that the improvement is not merely caused by generic multi-task regularization. To this end, we train a class-agnostic TE variant that only supervises TE localization with the L1 loss and removes the semantic classification loss. This variant reaches only 26.2 EPDMS on NAVSIM-v2, compared with 28.9 EPDMS for the full TE-aware model. The gap shows that regulatory semantics, rather than spatial localization alone, are critical for planning.

Ablation on Topology Information Integration. We systematically evaluate our topology integration strategies in Tab. 7 to determine the most effective mechanism for routing topological priors into the end-to-end planner.

(1) **Topology Synergy.** Introducing either centerline-to-centerline connectivity (LCLC, ID 1) or centerline-to-traffic-element constraints (LCTE, ID 2) substantially improves planning metrics. Combining both (ID 3) provides the most comprehensive spatial-logical context, yielding better performance.

(2) **Encoding Strategy: Language vs. Graph.** When encoding the merged directed topology graph, language-based encoding via BERT [12] (ID 3) outperforms Graph Convolutional Networks (GCN [61], ID 4), particularly in reducing the Collision Rate. Driving topologies consist of highly heterogeneous nodes (e.g., continuous spatial centerlines vs. discrete logical traffic lights). While GCN message-passing tends to over-smooth these semantic features, language models naturally map these heterogeneous attributes into a unified semantic space. This prevents information loss and perfectly preserves long-range causal driving rules.

(3) **Interaction Scope: Ego vs. Global.** Restricting the topology to ego-relevant elements (ID 4) proves superior to utilizing the global topology (ID 5). Global structures introduce redundant noise from irrelevant lanes and unaffected traffic elements, which distract the planner's attention mechanism. Ego-centric filtering ensures the model focuses solely on actionable causal relationships.

(4) **Robustness to Predicted Topology.** Finally, replacing the ground-truth (GT) topology [71] with real-time TopoMLP [78] predictions (ID 6, our final optimal setting) maintains performance comparable to the GT oracle (ID 3). This demonstrates the strong robustness of our interaction mechanism, proving it can effectively guide safe trajectory generation without relying on perfectly accurate perception priors.

Table 7: Ablation study of **topology information integration** on the **nuScenes** dataset. We ablate the topology types, encoding methods, topology source and interaction scope upon the **VAD** baseline.

ID	Topology		Encoder		Source		Scope		L2 (m) ↓				Collision Rate (%) ↓			
	LCTE	LCLC	BERT	GCN	GT [71]	Pred [78]	Global	Ego	1s	2s	3s	Avg.	1s	2s	3s	Avg.
0									0.41	0.70	1.05	0.72	0.07	0.17	0.41	0.22
1		✓	✓		✓			✓	0.34	0.60	0.95	0.63	0.06	0.14	0.27	**0.16**
2	✓		✓		✓			✓	0.34	0.59	0.93	**0.62**	0.06	0.16	0.33	0.18
3	✓	✓	✓		✓			✓	0.33	0.58	0.94	**0.62**	0.04	0.16	0.28	**0.16**
4	✓	✓		✓	✓			✓	0.34	0.60	0.93	**0.62**	0.07	0.18	0.43	0.23
5	✓	✓		✓	✓		✓		0.34	0.61	0.96	0.64	0.33	0.50	0.68	0.50
6	✓	✓	✓			✓		✓	0.34	0.59	0.92	**0.62**	0.05	0.15	0.29	**0.16**

Ablation on Traffic Element Detection Optimization. To obtain a highly robust traffic element detector for datasets lacking TE coordinates annotations (e.g., NAVSIM), we progressively optimize a baseline YOLOv8 [59] detector using datasets with available ground truth. As illustrated in Fig. 6, we progressively optimize the baseline detector into a robust feature extractor for the downstream planner by systematically applying strong data augmentation, class-balanced resampling, loss reweighting, pseudo-labeling, test-time augmentation (TTA), and an advanced YOLO backbone [60].

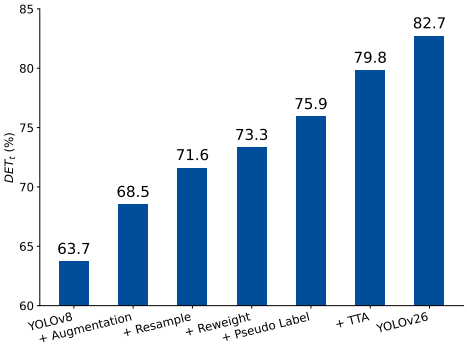

Fig. 6: Stepwise performance improvements of our traffic element detector on the OpenLane-V2 validation set, evaluated by the DET_t metric.

6 Conclusion

We propose a lightweight way to inject traffic elements (lights/signs) and lane topology into vision-based end-to-end planning. By constructing a unified 3D traffic-element representation, adding TE auxiliary supervision, and conditioning the planner with ego-centric topology encoded as structured language, we obtain consistent gains across four benchmarks (open-loop and closed-loop) and six representative planners, improving both trajectory accuracy and safety/compliance metrics with only minor runtime overhead. Importantly, our results demonstrate strong scalability: the gains persist with stronger backbones and further increase when training with more data, highlighting traffic-element reasoning as a previously under-emphasized but decision-critical factor in end-to-end driving.

References

1. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020)
3. Cao, W., Hallgarten, M., Li, T., Dauner, D., Gu, X., Wang, C., Miron, Y., Aiello, M., Li, H., Gilitschenski, I., et al.: Pseudo-simulation for autonomous driving. arXiv preprint arXiv:2506.04218 (2025)
4. Chang, X., Xue, M., Liu, X., Pan, Z., Wei, X.: Driving by the rules: A benchmark for integrating traffic sign regulations into vectorized hd map. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 6823–6833 (2025)
5. Chen, S., Jiang, B., Gao, H., Liao, B., Xu, Q., Zhang, Q., Huang, C., Liu, W., Wang, X.: Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. arXiv preprint arXiv:2402.13243 (2024)
6. Chen, Y., Wang, Y., Zhang, Z.: Drivinggpt: Unifying driving world modeling and planning with multi-modal autoregressive transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 26890–26900 (2025)
7. Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al.: Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 24185–24198 (2024)
8. Chi, H., Gao, H.a., Liu, Z., Liu, J., Liu, C., Li, J., Yang, K., Yu, Y., Wang, Z., Li, W., et al.: Impromptu vla: Open weights and open data for driving vision-language-action models. arXiv preprint arXiv:2505.23757 (2025)
9. Chitta, K., Prakash, A., Jaeger, B., Yu, Z., Renz, K., Geiger, A.: Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. IEEE transactions on pattern analysis and machine intelligence **45**(11), 12878–12895 (2022)
10. Dauner, D., Hallgarten, M., Geiger, A., Chitta, K.: Parting with misconceptions about learning-based vehicle motion planning. In: Conference on Robot Learning. pp. 1268–1281. PMLR (2023)
11. Dauner, D., Hallgarten, M., Li, T., Weng, X., Huang, Z., Yang, Z., Li, H., Gilitschenski, I., Ivanovic, B., Pavone, M., et al.: Navsim: Data-driven non-reactive autonomous vehicle simulation and benchmarking. Advances in Neural Information Processing Systems **37**, 28706–28719 (2024)
12. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers). pp. 4171–4186 (2019)
13. Ding, K., Chen, B., Su, Y., Gao, H.a., Jin, B., Sima, C., Zhang, W., Li, X., Barsch, P., Li, H., et al.: Hint-ad: Holistically aligned interpretability in end-to-end autonomous driving. arXiv preprint arXiv:2409.06702 (2024)
14. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)

15. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: Carla: An open urban driving simulator. In: Conference on robot learning. pp. 1–16. PMLR (2017)
16. Esser, P., Rombach, R., Ommer, B.: Taming transformers for high-resolution image synthesis. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12873–12883 (2021)
17. Fu, H., Zhang, D., Zhao, Z., Cui, J., Liang, D., Zhang, C., Zhang, D., Xie, H., Wang, B., Bai, X.: Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 24823–24834 (2025)
18. Fu, Y., Liao, W., Liu, X., Xu, H., Ma, Y., Zhang, Y., Dai, F.: Topologic: An interpretable pipeline for lane topology reasoning on driving scenes. Advances in Neural Information Processing Systems **37**, 61658–61676 (2024)
19. Gao, Y., Wang, J., Zhang, Z., Jiang, A., Wang, Y., Heng, Y., Wang, S., Sun, H., Hu, Z., Zhao, H.: Uniuncer: Unified dynamic static uncertainty for end to end driving. arXiv preprint arXiv:2603.07686 (2026)
20. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 (2021)
21. Guo, M., Zhang, Z., He, Y., Wang, K., Jing, L., Ling, H.: End-to-end autonomous driving without costly modularization and 3d manual annotation. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
22. Guo, X., Zhang, R., Duan, Y., He, Y., Nie, D., Huang, W., Zhang, C., Liu, S., Zhao, H., Chen, L.: Surds: Benchmarking spatial understanding and reasoning in driving scenarios with vision language models. Advances in Neural Information Processing Systems **38** (2026)
23. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
24. Hu, P., Huang, A., Dolan, J., Held, D., Ramanan, D.: 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)
25. Hu, S., Chen, L., Wu, P., Li, H., Yan, J., Tao, D.: 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)
26. Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al.: Planning-oriented autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 17853–17862 (2023)
27. Hwang, J.J., Xu, R., Lin, H., Hung, W.C., Ji, J., Choi, K., Huang, D., He, T., Covington, P., Sapp, B., et al.: Emma: End-to-end multimodal model for autonomous driving. arXiv preprint arXiv:2410.23262 (2024)
28. Jia, X., Gao, Y., Chen, L., Yan, J., Liu, P.L., Li, H.: Driveadapter: Breaking the coupling barrier of perception and planning in end-to-end autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 7953–7963 (2023)
29. Jia, X., Wu, P., Chen, L., Xie, J., He, C., Yan, J., Li, H.: Think twice before driving: Towards scalable decoders for end-to-end autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21983–21994 (2023)
30. Jia, X., Yang, Z., Li, Q., Zhang, Z., Yan, J.: Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. Advances in Neural Information Processing Systems **37**, 819–844 (2024)

31. Jia, X., You, J., Zhang, Z., Yan, J.: Drivetransformer: Unified transformer for scalable end-to-end autonomous driving. arXiv preprint arXiv:2503.07656 (2025)
32. Jiang, A., Gao, Y., Sun, Z., Wang, Y., Wang, J., Chai, J., Cao, Q., Heng, Y., Jiang, H., Dong, Y., et al.: Diffvla: Vision-language guided diffusion planning for autonomous driving. arXiv preprint arXiv:2505.19381 (2025)
33. Jiang, B., Chen, S., Liao, B., Zhang, X., Yin, W., Zhang, Q., Huang, C., Liu, W., Wang, X.: Senna: Bridging large vision-language models and end-to-end autonomous driving. arXiv preprint arXiv:2410.22313 (2024)
34. Jiang, B., Chen, S., Xu, Q., Liao, B., Chen, J., Zhou, H., Zhang, Q., Liu, W., Huang, C., Wang, X.: Vad: Vectorized scene representation for efficient autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8340–8350 (2023)
35. Khurana, T., Hu, P., Dave, A., Ziglar, J., Held, D., Ramanan, D.: Differentiable raycasting for self-supervised occupancy forecasting. In: European Conference on Computer Vision. pp. 353–369. Springer (2022)
36. Kirby, E., Boulch, A., Xu, Y., Yin, Y., Puy, G., Zablocki, É., Bursuc, A., Gidaris, S., Marlet, R., Bartoccioni, F., et al.: Driving on registers. arXiv preprint arXiv:2601.05083 (2026)
37. Li, P., Cui, D.: Navigation-guided sparse scene representation for end-to-end autonomous driving. arXiv preprint arXiv:2409.18341 (2024)
38. Li, T., Chen, L., Wang, H., Li, Y., Yang, J., Geng, X., Jiang, S., Wang, Y., Xu, H., Xu, C., et al.: Graph-based topology reasoning for driving scenes. arXiv preprint arXiv:2304.05277 (2023)
39. Li, T., Jia, P., Wang, B., Chen, L., Jiang, K., Yan, J., Li, H.: Lanesegnet: Map learning with lane segment perception for autonomous driving. arXiv preprint arXiv:2312.16108 (2023)
40. Li, Y., Zhang, Z., Qiu, X., Li, X., Liu, Z., Wang, L., Li, R., Zhu, Z., Gao, H.a., Lin, X., et al.: Reusing attention for one-stage lane topology understanding. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 16977–16984. IEEE (2025)
41. Li, Y., Fan, L., He, J., Wang, Y., Chen, Y., Zhang, Z., Tan, T.: Enhancing end-to-end autonomous driving with latent world model. arXiv preprint arXiv:2406.08481 (2024)
42. Li, Y., Shang, S., Liu, W., Zhan, B., Wang, H., Wang, Y., Chen, Y., Wang, X., An, Y., Tang, C., et al.: Drivevla-w0: World models amplify data scaling law in autonomous driving. arXiv preprint arXiv:2510.12796 (2025)
43. Li, Y., Xiong, K., Guo, X., Li, F., Yan, S., Xu, G., Zhou, L., Chen, L., Sun, H., Wang, B., et al.: Recogdrive: A reinforced cognitive framework for end-to-end autonomous driving. arXiv preprint arXiv:2506.08052 (2025)
44. Li, Z., Li, K., Wang, S., Lan, S., Yu, Z., Ji, Y., Li, Z., Zhu, Z., Kautz, J., Wu, Z., et al.: Hydra-mdp: End-to-end multimodal planning with multi-target hydra-distillation. arXiv preprint arXiv:2406.06978 (2024)
45. Li, Z., Yao, W., Wang, Z., Sun, X., Chen, J., Chang, N., Shen, M., Song, J., Wu, Z., Lan, S., et al.: Ztrs: Zero-imitation end-to-end autonomous driving with trajectory scoring. arXiv preprint arXiv:2510.24108 (2025)
46. Li, Z., Yao, W., Wang, Z., Sun, X., Chen, J., Chang, N., Shen, M., Wu, Z., Lan, S., Alvarez, J.M.: Generalized trajectory scoring for end-to-end multimodal planning. arXiv preprint arXiv:2506.06664 (2025)
47. Li, Z., Yu, Z., Lan, S., Li, J., Kautz, J., Lu, T., Alvarez, J.M.: 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 (2024)
48. Liao, B., Chen, S., Yin, H., Jiang, B., Wang, C., Yan, S., Zhang, X., Li, X., Zhang, Y., Zhang, Q., 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 (2025)
49. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
50. Liu, D., Gao, Y., Qian, D., Zhang, Q., Ye, X., Han, J., Zheng, Y., Liu, X., Xia, Z., Ding, D., et al.: Takead: Preference-based post-optimization for end-to-end autonomous driving with expert takeover data. IEEE Robotics and Automation Letters **11**(2), 1738–1745 (2025)
51. Liu, L., Jia, C., Yu, G., Song, Z., Li, J., Jia, F., Wu, P., Hao, X., Luo, Y.: Guideflow: Constraint-guided flow matching for planning in end-to-end autonomous driving. arXiv preprint arXiv:2511.18729 (2025)
52. Liu, S., Chen, W., Li, W., Wang, Z., Yang, L., Huang, J., Zhang, Y., Huang, Z., Cheng, Z., Yang, H.: Bridgedrive: Diffusion bridge policy for closed-loop trajectory planning in autonomous driving. arXiv preprint arXiv:2509.23589 (2025)
53. Lv, C., Qi, M., Liu, L., Ma, H.: T2sg: Traffic topology scene graph for topology reasoning in autonomous driving. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 17197–17206 (2025)
54. Nie, M., Peng, R., Wang, C., Cai, X., Han, J., Xu, H., Zhang, L.: Reason2drive: Towards interpretable and chain-based reasoning for autonomous driving. In: European Conference on Computer Vision. pp. 292–308. Springer (2024)
55. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
56. Park, S.Y., Cui, C., Ma, Y., Moradipari, A., Gupta, R., Han, K., Wang, Z.: Nuplanqa: A large-scale dataset and benchmark for multi-view driving scene understanding in multi-modal large language models. arXiv preprint arXiv:2503.12772 (2025)
57. Piccinelli, L., Sakaridis, C., Yang, Y.H., Segu, M., Li, S., Abbeloos, W., Van Gool, L.: Unidepthv2: Universal monocular metric depth estimation made simpler. IEEE Transactions on Pattern Analysis and Machine Intelligence (2025)
58. Qian, T., Chen, J., Zhuo, L., Jiao, Y., Jiang, Y.G.: Nuscenes-qa: A multi-modal visual question answering benchmark for autonomous driving scenario. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 38, pp. 4542–4550 (2024)
59. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
60. Sapkota, R., Cheppally, R.H., Sharda, A., Karkee, M.: Yolo26: key architectural enhancements and performance benchmarking for real-time object detection. arXiv preprint arXiv:2509.25164 (2025)
61. Schlichtkrull, M., Kipf, T.N., Bloem, P., Van Den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: European Semantic Web Conference. pp. 593–607. Springer (2018)
62. Shi, C., Shi, S., Sheng, K., Zhang, B., Jiang, L.: Drivex: Omni scene modeling for learning generalizable world knowledge in autonomous driving. In: Proceedings of

the IEEE/CVF International Conference on Computer Vision. pp. 28599–28609 (2025)
63. Sima, C., Chitta, K., Yu, Z., Lan, S., Luo, P., Geiger, A., Li, H., Alvarez, J.M.: Centaur: Robust end-to-end autonomous driving with test-time training. arXiv preprint arXiv:2503.11650 (2025)
64. Sima, C., Renz, K., Chitta, K., Chen, L., Zhang, H., Xie, C., Beißwenger, J., Luo, P., Geiger, A., Li, H.: Drivelm: Driving with graph visual question answering. In: European Conference on Computer Vision. pp. 256–274. Springer (2024)
65. Song, Z., Jia, C., Liu, L., Pan, H., Zhang, Y., Wang, J., Zhang, X., Xu, S., Yang, L., Luo, Y.: Don't shake the wheel: Momentum-aware planning in end-to-end autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 22432–22441 (2025)
66. Song, Z., Liu, L., Pan, H., Liao, B., Guo, M., Yang, L., Zhang, Y., Xu, S., Jia, C., Luo, Y.: Breaking imitation bottlenecks: Reinforced diffusion powers diverse trajectory generation. arXiv e-prints pp. arXiv–2507 (2025)
67. Sun, W., Lin, X., Shi, Y., Zhang, C., Wu, H., Zheng, S.: Sparsedrive: End-to-end autonomous driving via sparse scene representation. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 8795–8801. IEEE (2025)
68. Tian, H., Li, T., Liu, H., Yang, J., Qiu, Y., Li, G., Wang, J., Gao, Y., Zhang, Z., Wang, L., et al.: Simscale: Learning to drive via real-world simulation at scale. arXiv preprint arXiv:2511.23369 (2025)
69. Tian, X., Gu, J., Li, B., Liu, Y., Wang, Y., Zhao, Z., Zhan, K., Jia, P., Lang, X., Zhao, H.: Drivevlm: The convergence of autonomous driving and large vision-language models. arXiv preprint arXiv:2402.12289 (2024)
70. Tong, W., Sima, C., Wang, T., Chen, L., Wu, S., Deng, H., Gu, Y., Lu, L., Luo, P., Lin, D., et al.: Scene as occupancy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8406–8415 (2023)
71. Wang, H., Li, T., Li, Y., Chen, L., Sima, C., Liu, Z., Wang, B., Jia, P., Wang, Y., Jiang, S., et al.: Openlane-v2: A topology reasoning benchmark for unified 3d hd mapping. Advances in Neural Information Processing Systems **36**, 18873–18884 (2023)
72. Wang, J., Liu, X., Zheng, Y., Xing, Z., Li, P., Li, G., Ma, K., Chen, G., Ye, H., Xia, Z., et al.: Meanfuser: Fast one-step multi-modal trajectory generation and adaptive reconstruction via meanflow for end-to-end autonomous driving. arXiv preprint arXiv:2602.20060 (2026)
73. Wang, S., Yu, Z., Jiang, X., Lan, S., Shi, M., Chang, N., Kautz, J., Li, Y., Alvarez, J.M.: 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)
74. Wang, X., Cui, Y., Wang, J., Zhang, F., Wang, Y., Zhang, X., Luo, Z., Sun, Q., Li, Z., Wang, Y., et al.: Multimodal learning with next-token prediction for large multimodal models. Nature pp. 1–7 (2026)
75. Wang, Y., Li, X., Wang, W., Zhang, J., Li, Y., Chen, Y., Wang, X., Zhang, Z.: Unified vision-language-action model. arXiv preprint arXiv:2506.19850 (2025)
76. Weng, X., Ivanovic, B., Wang, Y., Wang, Y., Pavone, M.: 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)
77. Wozniak, M.K., Liu, L., Cai, Y., Jensfelt, P.: Prix: Learning to plan from raw pixels for end-to-end autonomous driving. arXiv preprint arXiv:2507.17596 (2025)

78. Wu, D., Chang, J., Jia, F., Liu, Y., Wang, T., Shen, J.: Topomlp: A simple yet strong pipeline for driving topology reasoning. arXiv preprint arXiv:2310.06753 (2023)
79. Wu, D., Han, W., Liu, Y., Wang, T., Xu, C.z., Zhang, X., Shen, J.: Language prompt for autonomous driving. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 39, pp. 8359–8367 (2025)
80. Wu, D., Jia, F., Chang, J., Li, Z., Sun, J., Han, C., Li, S., Liu, Y., Ge, Z., Wang, T.: The 1st-place solution for cvpr 2023 openlane topology in autonomous driving challenge. arXiv preprint arXiv:2306.09590 (2023)
81. Wu, P., Jia, X., Chen, L., Yan, J., Li, H., Qiao, Y.: Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. Advances in Neural Information Processing Systems **35**, 6119–6132 (2022)
82. Xing, Z., Zhang, X., Hu, Y., Jiang, B., He, T., Zhang, Q., Long, X., Yin, W.: Goalflow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 1602–1611 (2025)
83. Yao, W., Li, Z., Lan, S., Wang, Z., Sun, X., Alvarez, J.M., Wu, Z.: Drivesuprim: Towards precise trajectory selection for end-to-end planning. arXiv preprint arXiv:2506.06659 (2025)
84. Yuan, C., Zhang, Z., Sun, J., Sun, S., Huang, Z., Lee, C.D.W., Li, D., Han, Y., Wong, A., Tee, K.P., et al.: Drama: An efficient end-to-end motion planner for autonomous driving with mamba. arXiv preprint arXiv:2408.03601 (2024)
85. Zeng, W., Luo, W., Suo, S., Sadat, A., Yang, B., Casas, S., Urtasun, R.: End-to-end interpretable neural motion planner. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8660–8669 (2019)
86. Zhai, J.T., Feng, Z., Du, J., Mao, Y., Liu, J.J., Tan, Z., Zhang, Y., Ye, X., Wang, J.: Rethinking the open-loop evaluation of end-to-end autonomous driving in nuscenes. arXiv preprint arXiv:2305.10430 (2023)
87. Zhang, Z., Li, X., Zou, S., Chi, G., Li, S., Qiu, X., Wang, G., Zheng, G., Wang, L., Zhao, H., et al.: Chameleon: Fast-slow neuro-symbolic lane topology extraction. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 3752–3758. IEEE (2025)
88. Zhang, Z., Qiu, X., Zhang, B., Zheng, G., Gu, X., Chi, G., Gao, H.a., Wang, L., Liu, Z., Li, X., et al.: Delving into mapping uncertainty for mapless trajectory prediction. In: 2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 16969–16976. IEEE (2025)
89. Zhang, Z., Zou, S., Zheng, G., Zhu, Z., Gao, Y., Chi, G., Wang, S., Heng, Y., Sun, Z., Wang, Y., et al.: Unified map prior encoder for mapping and planning. arXiv preprint arXiv:2605.02762 (2026)
90. Zheng, W., Song, R., Guo, X., Zhang, C., Chen, L.: Genad: Generative end-to-end autonomous driving. In: European Conference on Computer Vision. pp. 87–104. Springer (2024)
91. Zheng, Y., Yang, P., Xing, Z., Zhang, Q., Zheng, Y., Gao, Y., Li, P., Zhang, T., Xia, Z., Jia, P., et al.: World4drive: End-to-end autonomous driving via intention-aware physical latent world model. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 28632–28642 (2025)
92. Zhou, Z., Cai, T., Zhao, S.Z., Zhang, Y., Huang, Z., Zhou, B., Ma, J.: Autovla: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. arXiv preprint arXiv:2506.13757 (2025)

93. Zou, J., Chen, S., Liao, B., Zheng, Z., Song, Y., Zhang, L., Zhang, Q., Liu, W., Wang, X.: Diffusiondrivev2: Reinforcement learning-constrained truncated diffusion modeling in end-to-end autonomous driving. arXiv preprint arXiv:2512.07745 (2025)