TRIAD Planner: Three-Level Robust Imitation Learning for Autonomous Driving

Anonymous Author(s)
Affiliation
Address
email

Abstract: Imitation learning for autonomous driving suffers from distribution shift between logged expert trajectories and self-induced deployment states. In Transformer-based planners, this shift can appear as ego-state shortcut learning, incomplete scene understanding, and unsafe mode selection under interaction uncertainty. This paper presents a robust imitation-learning planner based on a three-level robustness formulation. 1) Channel-level robustness: constrained ego-state attention to reduce shortcut dependence on a small subset of state channels. 2) Scene-level robustness: offline vision-language model (VLM) guidance to encourage the ego query to retain interaction-critical agent evidence and route-relevant map evidence. 3) Decision-level robustness: training-time multimodal supervision reshaped with a tail-risk score based on mean excess above Value at Risk. The scene-level VLM guidance and decision-level tail-risk supervision are used only during training, and the deployed planner incurs no additional inference-time overhead. On nuPlan test14, the proposed planner achieves the strongest performance among the evaluated learning-based methods and improves over CAR Planner, a strong constrained-attention imitation-learning baseline. The gains are especially pronounced on `test14-hard`, which contains more challenging scenarios, where NR-CLS and R-CLS change from 70.81 and 66.12 to 73.39 and 67.12, respectively. Ablation results show that combining all three-level components yields consistent performance across diverse scenarios, supporting the effectiveness of robust imitation learning for autonomous driving.

Keywords: Autonomous Driving, Imitation Learning, Robust Planning, Vision Language Model

1 Introduction

Robust motion planning remains a central challenge in autonomous driving [1, 2]. Imitation learning (IL) learns driving policies from expert demonstrations and has shown competitive performance on modern planning benchmarks [3, 4, 5]. This paradigm allows a planner to learn complex driving behavior directly from data without manually specifying every interaction rule. However, the training signal is typically collected under the logged expert state distribution, while deployment requires the planner to act on states induced by its own previous decisions [6, 7]. Small early deviations can lead the ego vehicle to states where the resulting observations, surrounding-agent configurations, and map-relative poses are rarely encountered in the demonstrations. This distribution shift is especially problematic in closed-loop driving, where prediction errors, representation errors, and trajectory-selection errors can feed back into subsequent decisions.

In Transformer-based IL planners, this shift can affect multiple stages of the planning pipeline. At the channel level, ego-state aggregation may collapse onto a small subset of channels, producing shortcut dependence on self-state cues [8, 9, 10, 11, 12]. At the scene level, the encoder may preserve broad map context while under-attending to an interaction-critical agent, or it may diffuse attention

Submitted to the 10th Conference on Robot Learning (CoRL 2026). Do not distribute.

across many external tokens without emphasizing the objects that constrain the ego decision [13, 14]. At the decision level, a multimodal decoder may assign high likelihood to a trajectory mode that remains close to the logged expert while still having unfavorable tail behavior under interaction uncertainty [15]. These effects arise over different variables, namely ego-state channels, external scene tokens, and candidate trajectory modes. They are therefore difficult to address with a single regularizer or final-stage selector.

This paper formulates robust imitation planning as a *three-level robustness* problem, spanning ego-state aggregation at the channel level, interaction-relevant scene-token weighting at the scene level, and supervision of candidate trajectory modes under interaction risk at the decision level. Based on this formulation, **TRIAD Planner** (**T**hree-level **R**obust **I**mitation learning for **A**utonomous **D**riving) combines constrained ego-state attention, offline VLM-guided scene supervision, and tail-risk-aware mode supervision within a feed-forward planner. The offline VLM-guided scene prior and tail-risk-aware mode targets are used only during training and are disabled during deployment.

TRIAD Planner is evaluated on the nuPlan benchmark [7] using open-loop score (OLS), non-reactive closed-loop score (NR-CLS), and reactive closed-loop score (R-CLS). The main contributions are as follows.

- **A unified three-level robustness framework for imitation-based planning.** Closed-loop imitation degradation in Transformer-based planners is addressed across channel, scene, and decision levels, corresponding to ego-state channel aggregation, external scene-token relevance, and multimodal trajectory-mode selection.
- **Channel- and scene-level robustness in attention-based planning.** Constrained ego-state attention is integrated with offline VLM-guided scene supervision, reducing shortcut dependence on ego-state channels while encouraging the ego query to retain interaction-critical agent and route-relevant map evidence without online VLM inference.
- **Decision-level robustness via tail-risk-aware mode supervision.** A clearance-based tail-risk score is incorporated into multimodal supervision to construct risk-aware hard and soft mode targets, discouraging trajectory modes with unfavorable tail behavior during training.
- **Robustness gains on nuPlan across random and hard splits.** Experiments on `test14-random` and `test14-hard` show consistent improvements in open-loop and closed-loop metrics over strong learning-based baselines, while preserving standard feed-forward inference without online VLM guidance.

2 Related Work

Autonomous driving planners span rule-based, learning-based, and hybrid formulations. Rule-based methods such as IDM [16] and PDM-Closed [17] encode motion and traffic priors, while learning-based planners such as UrbanDriver [18], GC-PGP [19], and PDM-Open [17] learn policies from driving data. Transformer-based imitation planners improve interaction modeling [11], although they can also suffer from ego-state shortcut dependence. PlanTF [12] mitigates this issue with a State Dropout Encoder, while CAR Planner [20] constrains ego-state attention concentration with an augmented-Lagrangian formulation. These methods motivate channel-level robustness, although they do not directly supervise external scene-token relevance.

Language- and vision-language-based planners improve scene understanding by incorporating high-level reasoning. LLM-Assist [21] uses language-based reasoning for closed-loop planning on nuPlan, and recent VLM-based planners [22, 23] use multimodal inputs for scene reasoning and motion planning. These methods show the value of language-based context, while online large-model inference can increase planning latency. TRIAD Planner instead queries the VLM offline and uses the resulting scene prior only to supervise ego-query attention during training.

Tail-risk objectives have been studied in planning, control, and imitation learning. CVaR provides a standard upper-tail risk formulation [24], with applications in risk-constrained control [25] and adversarial imitation learning [26]. However, these formulations are not directly designed for

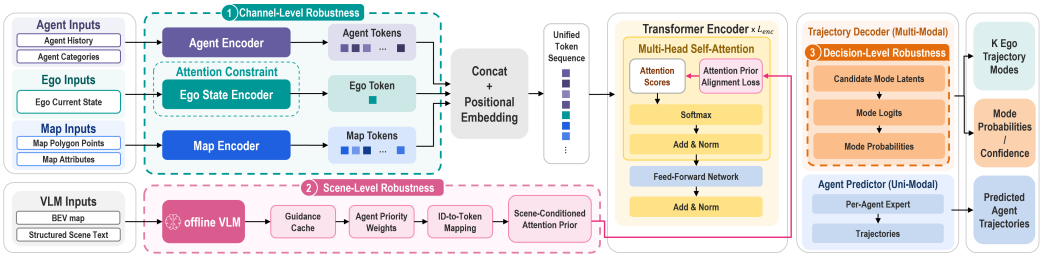


Figure 1: Overview of TRIAD Planner. Three robustness components are added to a base planner at the channel, scene, and decision levels while inference remains feed-forward.

multimodal imitation planners, where supervision must choose among candidate trajectory modes. TRIAD Planner applies a clearance-based tail-risk score to training-time mode supervision, constructing risk-aware hard and soft mode targets.

3 TRIAD Planner

3.1 Planner Overview

As shown in Figure 1, TRIAD Planner receives the current ego state, surrounding-agent histories, map polygon points and attributes, and outputs K candidate ego trajectories with mode logits. The base planner is augmented with three robustness components. The channel-level component constrains ego-state attention before scene-token fusion, the scene-level component supervises ego-query attention over external tokens, and the decision-level component reshapes multimodal supervision with a tail-risk score from predicted ego-agent clearance.

3.2 Base Architecture

Let $\mathbf{s}^{\text{ego}} \in \mathbb{R}^6$ denote the current ego state, represented by (x, y, ψ, v, a, s) for position, heading, velocity, acceleration, and steering angle. Let $\{\mathbf{H}_i\}_{i=1}^{A}$ and $\{\mathbf{P}_j\}_{j=1}^{M}$ denote the valid agent-history and map-polygon inputs, where A and M are the numbers of valid agent and map tokens after masking. Agent and map inputs are encoded as tokens in $\mathbb{R}^d$. The ego token is built by channel-wise projection followed by learned-query aggregation. A shared Transformer encoder processes all tokens. The final ego token is decoded into K candidate ego trajectories and mode logits.

3.3 Channel-Level Robustness Component via Constrained Ego-State Attention

Channel-level robustness is applied to the ego-state encoder. Each channel of $\mathbf{s}^{\text{ego}} \in \mathbb{R}^6$ is embedded independently, and a learned query attends to the six channel embeddings. Let $a_\theta^{(n)} \in \Delta^{C-1}$ denote the head-averaged ego-state attention weights for sample n, where $C = 6$. The sample-wise deviation from uniform attention and its batch average are

$$d(a_\theta^{(n)}) = \frac{1}{C}\sum_{i=1}^{C}\left|a_{\theta,i}^{(n)} - \frac{1}{C}\right|, \qquad D(\theta) = \frac{1}{N}\sum_{n=1}^{N} d(a_\theta^{(n)}).$$

The constraint $D(\theta) \le \delta$ uses the absolute-deviation form to discourage excessive concentration on a small subset of ego-state channels while preserving task-dependent variation across channels.

The attention constraint is incorporated into the training objective as an augmented Lagrangian penalty.

$$\mathcal{L}_{\text{ALM}} = \mu[D(\theta) - \delta]_+ + \frac{\rho}{2}[D(\theta) - \delta]_+^2, \tag{1}$$

where $[z]_+ = \max(0, z)$, μ is the Lagrange multiplier, and ρ is the quadratic penalty coefficient. The multiplier μ is updated after each optimizer step.

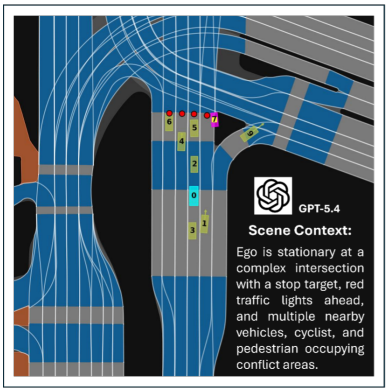
This regularizer acts only on ego-state aggregation before token fusion. It reduces shortcut dependence on self-state channels, while scene-level guidance supervises relevance over external tokens.

3.4 Scene-Level Robustness Component via VLM-Guided Token Relevance

Scene-level robustness addresses irrelevant attention weights assigned to external agent and map tokens. For training samples with valid VLM guidance, TRIAD Planner constructs an offline VLM-guided scene prior from scene-conditioned agent priority weights. This prior supervises the ego-query attention distribution and is excluded from the Transformer forward computation.

3.4.1 Offline Multimodal Guidance Generation

For each training frame, TRIAD Planner constructs an ego-centric BEV rendering and a structured scene description, as illustrated in Figure 2. The BEV rendering contains surrounding agents, identifiers, velocity cues, traffic-light states, and map elements, while the structured text summarizes ego dynamics, nearby agents, route context, and the ground-truth maneuver. A vision-language model processes this pair offline and returns a normalized priority distribution over visible non-ego agents. The priorities are stored before training and used only to supervise the ego-query attention distribution. Short VLM reasons are retained only for qualitative inspection.



ID	Weight	Reasoning
1	0.28	Closest moving vehicle behind-right in the ego lane corridor with moderate speed, posing the highest immediate collision risk.
3	0.16	Stationary vehicle directly ahead in the lane, relevant to forward blockage and stopping distance.
2	0.14	Stationary vehicle ahead in the same approach path, contributing to lane occupancy and limited clearance.
4	0.11	Vehicle farther ahead near the intersection connector with some motion, lower immediate but still relevant conflict risk.
5	0.1	Stationary vehicle near the upper intersection area, potentially affecting cross-traffic and turning space.
6	0.09	Stationary vehicle near the upper-left approach, relevant to intersection occupancy but farther from ego.
7	0.06	Slow bicycle near the upper-right connector, vulnerable road user with moderate conflict potential.
9	0.02	Far vehicle on the right-side connector moving away from the ego path, lowest immediate priority.

Figure 2: Offline VLM-guided scene-prior construction. The VLM is queried only during training to produce agent-level relevance scores.

3.4.2 Teacher Prior Construction and Attention Supervision

For guided samples, map polygons are scored with a route-aware geometric prior, and agent-map scores are normalized over valid external tokens to obtain $\mathbf{p}^{\text{scene}}$, defined over the union of valid agent and map tokens. Let $\mathcal{B}_{\text{guide}}$ denote the subset of minibatch samples with valid VLM guidance, and let $\widehat{\mathbf{p}}^{\text{scene}}$ denote the masked and normalized attention distribution of the ego query over valid agent and map keys. The scene-level guidance loss is

$$\mathcal{L}_{\text{scene}} = -\frac{1}{|\mathcal{B}_{\text{guide}}|} \sum_{n \in \mathcal{B}_{\text{guide}}} \sum_{u=1}^{A+M} p_{n,u}^{\text{scene}} \log \widehat{p}_{n,u}^{\text{scene}}. \tag{2}$$

This loss encourages the ego query to retain interaction-critical dynamic agents while preserving route-relevant map evidence.

3.5 Decision-Level Robustness Component via Tail-Risk-Aware Mode Supervision

Decision-level robustness is imposed on the multimodal trajectory set predicted by the decoder. For each mode k, a per-step clearance-risk sequence $z^{(k)} = (z_1^{(k)}, \ldots, z_{T_f}^{(k)})$ is computed from predicted ego-agent proximity, where T_f denotes the number of future planning steps. The risk is defined from the clearance deficit after accounting for the effective ego-agent safety radius and a safety margin.

To emphasize rare unsafe frames rather than average trajectory risk, the temporal upper tail of $z^{(k)}$ is summarized using the mean excess above Value at Risk. Unlike an average risk score, this tail summary focuses on the magnitude of risk beyond a high-quantile threshold, making the supervision

sensitive to short but severe proximity events. Let $\eta_\alpha^{(k)} = \mathrm{VaR}_\alpha(z^{(k)})$ denote the empirical α-quantile of the risk sequence, where $\alpha \in (0,1)$ is the tail quantile level. The tail-risk score for mode k is

$$r_k^{\mathrm{tail}} = \frac{1}{(1-\alpha)T_f} \sum_{t=1}^{T_f} \left[z_t^{(k)} - \eta_\alpha^{(k)} \right]_+ . \tag{3}$$

Let $p^{\mathrm{mode}} = \mathrm{softmax}(\pi)$ denote the predicted mode distribution. During training, the supervised mode is selected by combining mode confidence and tail risk as

$$s_k = -\log p_k^{\mathrm{mode}} + \lambda_{\mathrm{tail}} r_k^{\mathrm{tail}}, \qquad k^* = \arg\min_k s_k. \tag{4}$$

Here, λ_{tail} controls the trade-off between mode confidence and tail risk. The selected mode is used for trajectory regression, keeping supervision anchored to the logged expert trajectory while discouraging high-risk modes.

A risk-aware soft target over modes is also constructed as

$$q_k^{\mathrm{mode}} = \frac{\mathrm{sg}(p_k^{\mathrm{mode}}) \exp(-\lambda_{\mathrm{tail}} \tilde{r}_k^{\mathrm{tail}})}{\sum_{j=1}^{K} \mathrm{sg}(p_j^{\mathrm{mode}}) \exp(-\lambda_{\mathrm{tail}} \tilde{r}_j^{\mathrm{tail}})}, \tag{5}$$

where $\tilde{r}_k^{\mathrm{tail}}$ is the within-sample normalized tail-risk score and $\mathrm{sg}(\cdot)$ denotes stop-gradient. The probability head is trained with

$$\mathcal{L}_{\mathrm{risk}} = \mathrm{KL}(\mathbf{q}^{\mathrm{mode}} \| \mathbf{p}^{\mathrm{mode}}).$$

This loss is used only to shape the training-time mode targets.

3.6 Training Objective and Inference Procedure

Training combines a base imitation objective with the three robustness terms. Using the risk-aware mode k^* from Eq. 4, the standard imitation objective combines trajectory regression, mode classification, and auxiliary agent prediction:

$$\mathcal{L}_{\mathrm{base}} = \mathcal{L}_{\mathrm{traj}} + \lambda_{\mathrm{mode}} \mathcal{L}_{\mathrm{mode}} + \lambda_{\mathrm{pred}} \mathcal{L}_{\mathrm{pred}}. \tag{6}$$

Here, $\mathcal{L}_{\mathrm{traj}}$ supervises the selected ego trajectory, $\mathcal{L}_{\mathrm{mode}}$ trains the mode logits with the selected mode target, and $\mathcal{L}_{\mathrm{pred}}$ supervises surrounding-agent prediction.

The final objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{base}} + \mathcal{L}_{\mathrm{ALM}} + \lambda_{\mathrm{scene}} \mathcal{L}_{\mathrm{scene}} + \lambda_{\mathrm{risk}} \mathcal{L}_{\mathrm{risk}}, \tag{7}$$

Here, λ_{risk} weights the risk-supervision loss, while λ_{tail} controls the risk sensitivity used to construct its targets. During training, offline VLM guidance is used only to construct $\mathbf{p}^{\mathrm{scene}}$ for $\mathcal{L}_{\mathrm{scene}}$. During inference, no VLM prior is queried, and the deployed planner selects the final trajectory from the learned mode distribution in a standard feed-forward pass.

4 Experiments

4.1 Experimental Setup

GPT-5.4-Mini was used to generate VLM guidance for 20% of the training set to limit offline processing cost, and guided samples were oversampled with probability 0.5 during training. The planners were trained for 25 epochs with batch size 64. Training used AdamW with an initial learning rate of $1\mathrm{e}{-3}$, weight decay $1\mathrm{e}{-4}$, and a cosine schedule with 3 warm-up epochs. State perturbation augmentation [27] was applied with probability 0.5. The current ego state was perturbed with bounded noise, and all coordinates were re-normalized in the perturbed ego frame.

Evaluation follows the nuPlan benchmark on `test14-random` and `test14-hard`. Each split contains 261 scenarios across 14 scenario types. The reported metrics are open-loop score (OLS), non-reactive closed-loop score (NR-CLS), and reactive closed-loop score (R-CLS). OLS measures imitation quality against logged expert behavior. NR-CLS evaluates closed-loop rollout with replayed agents. R-CLS evaluates closed-loop rollout with reactive simulated agents.

Table 1: Performance comparison on the test14 benchmark.

Planners		Test14-random			Test14-hard		
Type	**Method**	OLS	NR-CLS	R-CLS	OLS	NR-CLS	R-CLS
Expert	Log-replay	100.0	94.03	75.86	100.0	85.96	68.80
Rule-based	IDM	34.15	70.39	72.42	20.07	56.15	62.26
	PDM-Closed	46.33	90.05	91.64	26.43	65.08	75.19
Hybrid	PDM-Hybrid	81.26	90.10	91.29	74.08	65.99	76.07
Learning-based	RasterModel	62.93	68.71	67.52	52.38	50.65	51.41
	UrbanDriver	82.45	63.57	60.92	76.90	51.67	49.06
	GC-PGP	80.89	61.57	54.88	75.40	47.16	41.39
	PDM-Open	84.14	52.81	57.23	79.06	33.51	35.83
	PlanTF	86.27	85.23	79.36	83.34	70.03	59.83
	Pluto (w/o post.)	87.15	86.60	77.96	81.89	70.28	56.70
	CAR Planner	87.67	85.91	78.31	86.31	70.81	66.12
	TRIAD Planner	89.72	87.65	82.55	87.13	73.39	67.12

Table 2: Scenario-level inference latency on 261 evaluation scenarios. Lower is better. Header abbreviations denote UrbanDriver, RasterModel, PDM-Open, PDM-Hybrid, and PDM-Closed.

Metric	CAR	**TRIAD**	PlanTF	Urban	Raster	PDM-O	PlanCNN	GCPGP	PDM-H	PDM-C	Pluto
Median (s)	**14.97**	**15.11**	15.15	17.09	18.07	19.05	20.37	21.76	32.65	34.32	45.39
Mean (s)	**15.00**	**15.18**	15.01	18.63	18.05	19.31	20.09	21.64	32.74	34.52	43.48
Std. (s)	2.45	**2.44**	2.43	6.27	6.31	6.86	6.53	6.89	8.44	9.09	13.89
Total (min)	**65.2**	**66.0**	65.3	81.1	78.5	84.0	87.4	94.1	142.4	150.2	189.1

4.2 Experimental Results

Overall benchmark results. Table 1 compares TRIAD Planner with rule-based, hybrid, and learning-based planners on `test14-random` and `test14-hard`. Among the evaluated learning-based planners, TRIAD Planner obtains the strongest performance across OLS, NR-CLS, and R-CLS. Compared with CAR Planner, which already includes the channel-level constraint, TRIAD Planner improves R-CLS from 78.31 to 82.55 on `test14-random` and from 66.12 to 67.12 on `test14-hard`, with consistent OLS and NR-CLS gains. This suggests that scene-level guidance and decision-level supervision improve closed-loop robustness beyond constrained ego-state aggregation.

Inference efficiency. Table 2 summarizes scenario-level inference latency on 261 evaluation scenarios. TRIAD Planner remains close to CAR Planner, which uses constrained ego-state attention, and PlanTF, which uses a State Dropout Encoder to mitigate ego-state shortcut learning, in median, mean, and variance. This indicates a lightweight and stable deployment profile.

Ablation analysis of the three levels. Figure 3 evaluates how the channel (C), scene (S), and decision (D) components contribute to the Base planner. Base denotes the vanilla Transformer planner, and Base+C corresponds to CAR Planner. Scene- and decision-level components affect the Base planner through different supervision targets, with S guiding external-token relevance and D reshaping risk-aware mode selection. The individual Base+S and Base+D variants improve several closed-loop scores over Base, suggesting that both external-token relevance and mode-level risk provide useful training signals.

However, the Base+S+D variant suggests that combining scene and decision supervision alone may not consistently improve all metrics when channel-level aggregation remains unconstrained. This indicates that better scene relevance and risk-aware mode selection do not fully address shortcut dependence in ego-state aggregation. TRIAD combines C, S, and D and provides the most balanced profile across OLS, NR-CLS, and R-CLS on both `test14-random` and `test14-hard`. This trend suggests that the channel-level constraint provides a stabilizing anchor for integrating scene- and decision-level supervision, supporting the complementary roles of the three robustness components.

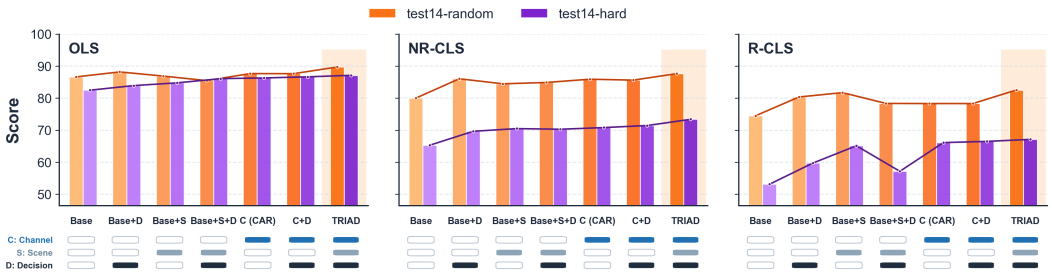
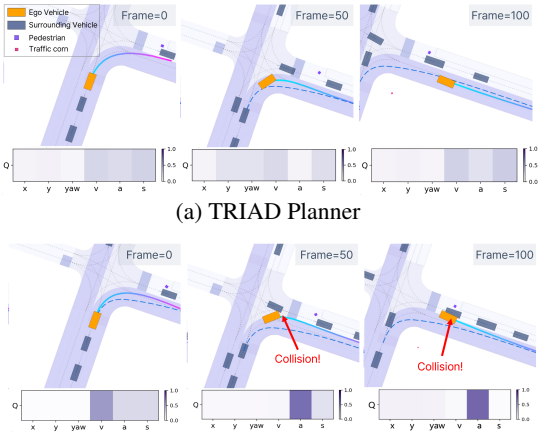


Figure 3: Ablation of channel, scene, and decision robustness components across `test14-random` and `test14-hard`. Bars show OLS, NR-CLS, and R-CLS, and the lower indicators denote the active components, with C, S, and D corresponding to channel, scene, and decision robustness.

Channel-level analysis. Figure 4 visualizes ego-state attention in the same right-turn scenario. At the beginning of the rollout, both planners attend to multiple ego-state channels. After the interaction begins, PlanTF places most of the ego-query attention on the acceleration channel, as shown in Figure 4(b). The planned trajectory then drifts toward the nearby vehicle and leads to a collision in this scenario, suggesting that excessive reliance on a single self-state channel can produce an unstable response. This pattern provides an interpretable failure signal. In contrast, TRIAD Planner maintains a broader allocation across dynamic ego-state channels, including velocity and acceleration. This broader allocation is consistent with the channel-level constraint and helps the resulting plan stay closer to the route centerline throughout the maneuver.



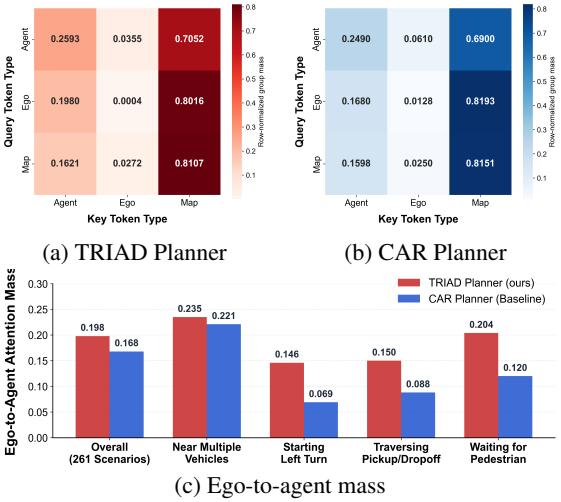
(a) TRIAD Planner

(b) PlanTF

Figure 4: Ego-state encoder attention in a right-turn scenario. (a) and (b) compare TRIAD Planner and PlanTF.

Scene-level analysis. Figure 5 compares TRIAD Planner and CAR Planner by aggregating pre-decoder Transformer attention into `Agent`, `Ego`, and `Map` query-key groups.

The VLM-guided scene prior supervises the ego query over external agent and map tokens, so the analysis focuses on the `Ego` query row. Both planners remain map-oriented, reflecting the importance of lane and route geometry. TRIAD Planner increases ego-to-agent attention from 0.168 to 0.198 while maintaining dominant ego-to-map attention, which changes only slightly from 0.819 to 0.802. Ego-to-ego attention decreases from 0.013 to approximately zero, indicating a weaker ego-token self-loop after ego-state features have already been encoded before token fusion.



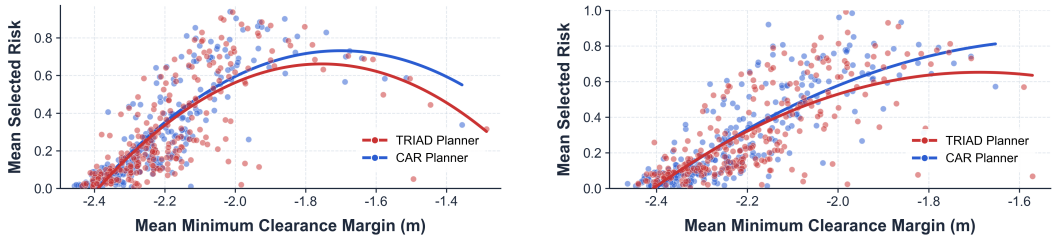
(a) TRIAD Planner

(b) CAR Planner

(c) Ego-to-agent mass

Figure 5: Scene-level attention analysis on `test14-random`. (a) and (b) show group attention maps, and (c) shows Ego→`Agent` mass

Figure 5(c) further shows that ego-to-agent attention changes are larger in actor-dependent contexts such as starting left turns, pickup and



(a) `test14-random` (b) `test14-hard`

Figure 6: Selected-risk trends over clearance margins on `test14-random` and `test14-hard`.

dropoff zones, and pedestrian-crossing scenes. These trends align with the scene-level objective, which encourages the ego query to retain interaction-critical agents and route-relevant map evidence.

Decision-level analysis. Figure 6 compares selected trajectory risk against the mean minimum clearance margin after subtracting the effective safety radius. More negative clearance values indicate tighter proximity, and lower risk values indicate lower selected-mode tail risk under comparable proximity.

On both splits, selected risk increases from low-risk cruising to denser interaction regimes, while the fitted TRIAD Planner trend remains below the CAR Planner trend over most of the observed clearance range. TRIAD Planner shows a lower fitted risk trend than CAR Planner even when the selected trajectories have similar minimum clearance to surrounding agents. This indicates that the reduction is not simply due to choosing trajectories farther from surrounding agents. Instead, training-time decision-level supervision shifts the learned mode distribution toward lower-tail-risk candidates under comparable proximity.

5 Conclusion

This paper presented TRIAD Planner, a three-level robustness framework for imitation-learning-based autonomous driving. TRIAD Planner combines channel-level constrained ego-state attention, scene-level offline VLM-guided supervision, and decision-level tail-risk-aware mode supervision within a feed-forward planner. Since the scene- and decision-level components are used only during training, deployment requires no online VLM query or additional inference-time module.

Experiments on nuPlan test14 show improved planning quality and closed-loop performance over strong learning-based planners. Ablation results further show that combining all three-level components yields consistent performance across diverse scenarios, supporting the complementary roles of channel-, scene-, and decision-level robustness in imitation-based planning.

One limitation is that the VLM-guided scene prior depends on the quality of offline VLM priorities and the completeness of the BEV-text scene description. Incorrect or incomplete VLM priorities may provide noisy supervision for rare or ambiguous interactions. Future work will study more reliable scene-prior construction and evaluate the three-level formulation beyond nuPlan, including additional datasets and closed-loop simulators with different maps, traffic patterns, and interaction styles.

References

[1] A. Tampuu, T. Matiisen, M. Semikin, D. Fishman, and N. Muhammad. A survey of end-to-end driving: Architectures and training methods. *IEEE Transactions on Neural Networks and Learning Systems*, 33(4):1364–1384, 2020.

[2] K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, and V. H. C. De Albuquerque. Deep learning for safe autonomous driving: Current challenges and future directions. *IEEE Transactions on Intelligent Transportation Systems*, 22(7):4316–4336, 2020.

[3] D. A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in Neural Information Processing Systems*, 1, 1988.

[4] A. Amini, I. Gilitschenski, J. Phillips, J. Moseyko, R. Banerjee, S. Karaman, and D. Rus. Learning robust control policies for end-to-end autonomous driving from data-driven simulation. *IEEE Robotics and Automation Letters*, 5(2):1143–1150, 2020. doi:10.1109/LRA.2020.2966414.

[5] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi. A survey of imitation learning: Algorithms, recent developments, and challenges. *IEEE Transactions on Cybernetics*, 2024.

[6] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[7] H. Caesar, J. Kabzan, K. S. Tan, W. K. Fong, E. Wolff, A. Lang, L. Fletcher, O. Beijbom, and S. Omari. nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021.

[8] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[9] C.-C. Chuang, D. Yang, C. Wen, and Y. Gao. Resolving copycat problems in visual imitation learning via residual action prediction. In *European Conference on Computer Vision*, pages 392–409. Springer, 2022.

[10] C. Wen, J. Lin, T. Darrell, D. Jayaraman, and Y. Gao. Fighting copycat agents in behavioral cloning from observation histories. *Advances in Neural Information Processing Systems*, 33: 2564–2575, 2020.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[12] J. Cheng, Y. Chen, X. Mei, B. Yang, B. Li, and M. Liu. Rethinking imitation-based planners for autonomous driving. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 14123–14130. IEEE, 2024.

[13] Z. Huang, Z. Sheng, Y. Qu, J. You, and S. Chen. Vlm-rl: A unified vision language models and reinforcement learning framework for safe autonomous driving. *Transportation Research Part C: Emerging Technologies*, 180:105321, 2025.

[14] Y. Chen, Z.-h. Ding, Z. Wang, Y. Wang, L. Zhang, and S. Liu. Asynchronous large language model enhanced planner for autonomous driving. In *European Conference on Computer Vision*, pages 22–38. Springer, 2024.

[15] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022.

[16] M. Treiber, A. Hennecke, and D. Helbing. Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62(2):1805–1824, 2000.

[17] D. Dauner, M. Hallgarten, A. Geiger, and K. Chitta. Parting with misconceptions about learning-based vehicle motion planning. In *Conference on Robot Learning*, pages 1268–1281. PMLR, 2023.

[18] O. Scheel, L. Bergamini, M. Wolczyk, B. Osiński, and P. Ondruska. Urban driver: Learning to drive from real-world demonstrations using policy gradients. In *Conference on Robot Learning*, pages 718–728. PMLR, 2022.

[19] M. Hallgarten, M. Stoll, and A. Zell. From prediction to planning with goal conditioned lane graph traversals. In *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, pages 951–958. IEEE, 2023.

[20] J. Kim and K. Choi. Car planner: Constrained-attention-based robust imitation learning for autonomous driving. *Authorea Preprints*, 2025.

[21] S. P. Sharan, F. Pittaluga, V. Kumar B. G., and M. Chandraker. LLM-assist: Enhancing closed-loop planning with language-based reasoning. *arXiv preprint arXiv:2401.00125*, 2023.

[22] Y. Zheng, Z. Xing, Q. Zhang, B. Jin, P. Li, Y. Zheng, Z. Xia, Y. Chen, and D. Zhao. Planagent: A multi-modal large language agent for closed-loop vehicle motion planning. *IEEE Transactions on Cognitive and Developmental Systems*, 2026.

[23] Z. Tang, S. Zhang, J. Deng, C. Wang, G. You, Y. Huang, X. Lin, and Y. Zhang. Vlmplanner: Integrating visual language models with motion planning. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 5040–5049, 2025.

[24] R. T. Rockafellar and S. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk*, 2:21–42, 2000.

[25] A. Hakobyan, G. C. Kim, and I. Yang. Risk-aware motion planning and control using CVaR-constrained optimization. *IEEE Robotics and Automation Letters*, 4(4):3924–3931, 2019.

[26] A. Santara, A. Naik, B. Ravindran, D. Das, D. Mudigere, S. Avancha, and B. Kaul. RAIL: Risk-averse imitation learning. *arXiv preprint arXiv:1707.06658*, 2017.

[27] M. Bansal, A. Krizhevsky, and A. Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.