Commun Transp Res

Research Article

Dynamically local-enhancement planner for large-scale autonomous driving

Nanshan Deng[1,†], Weitao Zhou[1,†,✉], Yifei He[1], Qian Cheng[1], Bo Zhang[2], Junze Wen[1], Chunyang Liu[2], Jianmo He[2], Xiang Sha[2], Zelin Qian[2], Kun Jiang[1], Mengmeng Yang[1], Zhong Cao[3], Diange Yang[1,✉]

Cite this article: https://doi.org/10.26599/COMMTR.2026.9640020

ABSTRACT: Current autonomous vehicles are typically deployed within limited geographic regions, while scalable operation across diverse locations is increasingly demanded. As deployment regions expand, planners must cope with heterogeneous traffic dynamics under fixed on-board computational and memory budgets, where adapting a single monolithic model through data aggregation or parameter expansion becomes inefficient and costly. This paper proposes the Dynamically Local-Enhancement (DLE) planner, which improves region-level adaptability without increasing policy capacity or performing full online policy optimization during deployment. Global driving competence is decoupled from region-specific adaptation through explicit region-conditioned representations. Long-term regional characteristics are distilled into map-level historical memory via a latent-variable encoder, while real-time interactions are modeled by a dual-layer traffic graph neural network that jointly captures vehicle interactions and road topology. The resulting region-conditioned representation is used to condition a shared reinforcement learning planner at inference time, where dynamic behavior arises from location-indexed retrieval and conditional forward inference rather than parameter growth. We evaluate DLE in multi-region closed-loop CARLA benchmarks. Under a fixed parameter budget, DLE consistently improves cross-regional adaptability and outperforms baselines in safety and comfort metrics. These results indicate that memory-based region-aware enhancement offers a practical paradigm for scaling autonomous driving planners under deployment constraints.

KEYWORDS: Autonomous Driving, Reinforcement Learning, Latent-variable, Region-conditioned Driving

1 Introduction

Autonomous driving systems have achieved remarkable progress in recent years, with companies such as Waymo demonstrating the ability to operate tens of thousands of miles with few or no disengagements in regulated trials (Xu et al., 2021). These achievements showcase the feasibility of autonomous mobility under carefully defined conditions. However, most deployments to date have been confined to specific regions or restricted road types. When systems are expanded to new cities or countries, substantial challenges arise due to shifts in local driving behaviors, in particular multi-agent interaction norms that directly affect planning decisions such as yielding, gap acceptance, and merging aggressiveness, even when road topology is similar (Deng et al., 2021b; Zhou et al., 2022). For example, Waymo's public safety reports highlight the extensive retraining and validation effort required before expanding into new metropolitan areas (Schwall et al., 2020). Similarly, Tesla's Full Self-Driving (FSD) beta, developed primarily on U.S. Road data, has faced difficulties adapting to Chinese and European contexts, where traffic norms and legal frameworks differ significantly.

These cases illustrate a broader issue: As the operational region of autonomous driving systems expands, the likelihood of performance degradation increases, and ensuring consistent, non-conflicting planning logic becomes progressively harder. A natural response is to continually increase the capacity of models so that they can cover a broader range of driving regions (Cao et al., 2023). Yet in practice, model size often grows rapidly with dataset scale (Ouyang et al., 2022), raising concerns about deployability. A similar issue arises when relying on a single model or rule-based system to handle all regions: new rules introduced for novel scenarios must be checked against existing ones to avoid contradictions, a task that becomes increasingly complex as the system grows (Deng et al., 2021a). These challenges highlight fundamental concerns about the adaptability of autonomous systems and motivate the need for approaches that can provide scalable and conflict-free adaptability across diverse driving regions.

In this work, we study cross-regional planning under a position-varying interaction distribution, where geographically distinct areas induce different multi-agent behavior modes. To cope with such regional variability, the community has explored a range of adaptation mechanisms. Although these approaches differ in form, many share a common strategy of adapting by modifying or expanding the policy itself, for example through fine-tuning/transfer learning (Diehl et al., 2025), parameter-efficient adapters(Hu et al., 2022), online test-time updates(Sima et al., 2025),

† Nanshan Deng and Weitao Zhou contributed equally to this work.
[1] School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China [2] Didi Chuxing, Beijing100095, China. [3] Department of Civil and Environmental Engineering, University of Michigan, Michigan 48109, USA.
✉ Corresponding author. E-mail: W. Zhou, zhouwt@mail.tsinghua.edu.cn; D. Yang, ydg@mail.tsinghua.edu.cn
Received: October 29, 2025; Revised: January 25, 2026; Accepted: March 20, 2026

or modular expert selection(Zhu et al., 2023).

Such parameter-centric adaptation has achieved notable success in many settings, but it also relies on an implicit assumption that regional differences can be addressed by adjusting a single global model toward a sequence of target domains. This assumption is often violated in real-world autonomous driving. Driving behaviors across geographic regions are not hierarchically related but coexist in parallel (Zhu et al., 2023). Differences in yielding norms, gap acceptance, merging aggressiveness, and broader social driving conventions form distinct interaction modes that are tied to location. Updating policy parameters to fit one region can therefore degrade performance in others, leading to instability, conflicts, and growing deployment complexity as the number of regions increases.

Our approach stems from the observation that driving competence can be divided into two parts. The first is a set of region-invariant skills such as collision avoidance, lane following, and basic interaction with surrounding vehicles. The second is adaptation to local interaction conventions, including typical yielding behavior, gap acceptance, merging aggressiveness, and lane usage norms. Human drivers naturally combine these two components. After mastering the general skill of driving, entering a new city mainly requires learning how other road users behave in that place. This perspective suggests that scalable cross-regional deployment should preserve a shared core driving policy while enabling location-indexed specialization in interaction behaviors, without continuously rewriting/expanding the policy parameters.

Motivated by this perspective, we proposed *Dynamic Local-Enhancement (DLE)* planner. DLE maintains a single shared global policy that captures universal driving skills, and augments the planning state with compact, location-indexed regional representations learned from historical interactions. By localizing adaptation in the state representation rather than in region-specific policy parameters or online gradient updates, DLE enables persistent region-aware behavior under a fixed policy capacity. The main contributions of this work are summarized as follows:

1) We formulate cross-regional planning as a position-varying problem under non-stationary interaction distributions and propose the Dynamic Local-Enhancement (DLE) framework to enable persistent region-aware behavior under a fixed policy capacity, without per-region policy copies or online fine-tuning.

2) We introduce a dual-layer graph representation that encodes interaction states together with road topology into compact, location-indexed regional features, enabling persistent storage and retrieval of regional driving patterns.

3) We develop a reinforcement learning-based enhancement objective that injects regional features into the planner's state representation to improve closed-loop safety and efficiency across regions while maintaining a single shared policy.

2 Related Works

2.1. Region-specific driving policy fine-tuning and adaptation

One mainstream approach for cross-regional autonomous driving planning is to fine-tune a base driving policy on region-specific data, so that its interaction and decision-making behavior adapts to local traffic patterns. Recent examples include LoRD (Diehl et al., 2025), which applies low-rank residual decoders to adapt planning policies from U.S. to Singapore driving while mitigating catastrophic forgetting. Similar ideas could in principle be implemented using lightweight parameter-efficient mechanisms such as adapters or low-rank updates, Align2Act (Jaisankar and Tandel, 2025) applies low-rank adaptation (LoRA) (Hu et al., 2022) to instruction-tuned driving models and demonstrates that lightweight parameter updates can adapt planning behavior across different driving scenarios in the nuPlan benchmark. Geo-aware conditioning Mixture-of-Experts (MoE) driving policy, as in AnyD (Zhu et al., 2023), further demonstrates that explicitly encoding geographic priors improves robustness in open-loop and closed-loop evaluation.

While effective, all these methods require maintaining multiple sets of parameters for different regions, leading to heavy memory overhead and complex version management in practice. Moreover, the effectiveness of fine-tuning relies on the assumption that adaptation involves specializing in a broadly trained model to a simpler sub-task. In contrast, differences between region A and region B in autonomous driving (e.g., opposite traffic sides, distinct lane structures, or divergent driving norms) are not hierarchical but parallel. As a result, naive fine-tuning may fail to capture the true distribution gap, risking conflicts between regions rather than seamless transfer..

2.2. Test-time/continual driving policy adaptation

Another line of work studies test-time and continual adaptation (TTA/TTT), where models are adjusted during deployment to cope with regional distribution shifts without offline retraining. In autonomous driving, Centaur (Sima et al., 2025) applies a cluster-entropy objective to adapt end-to-end planners, improving robustness under domain shift. More recently, model-based policy adaptation frameworks have been proposed to convert open-loop end-to-end driving agents into safer closed-loop planners via online adaptation mechanisms. (Lin et al., 2025)

Beyond autonomous driving, similar ideas have been explored in robotic navigation and planning. Methods such as TTA-Nav (Piriyajitakonkij et al., 2024) and (Xu et al., 2025) perform test-time adaptation of navigation and reinforcement learning policies to handle non-stationary environments, demonstrating the effectiveness of deployment-time adaptation at the policy level. Meanwhile, perception-oriented TTA methods, including AR-TTA (Sójka et al., 2023), DARTH (Segu et al., 2023), and TTA-DAME (Jeon et al., 2025) focus on adapting visual or state representations under environmental changes such as weather and lighting, indirectly benefiting downstream planning.

Beyond planning, AR-TTA (Sójka et al., 2023) evaluates TTA methods on benchmarks such as SHIFT and CLAD-C with a memory buffer for stability, DARTH (Segu et al., 2023) addresses domain shifts in multiple object tracking, and TTA-DAME (Jeon et al., 2025) tackles environmental variations such as weather and lighting.

Despite these advances, most TTA-based approaches rely on online parameter updates and provide only transient improvements tied to the current test trajectory. Each agent adapts independently, and no persistent, region-indexed memory is retained. This makes it difficult to accumulate and reuse local driving knowledge across repeated visits to the same geographic

region, limiting their scalability and long-term stability in large-scale, multi-region deployment.

2.3. Memory- and retrieval-augmented planning

Recent driving planners have begun to use memory and retrieval to address cross-domain and cross-region variability in closed-loop planning, where policies trained on one environment often fail to generalize to new traffic distributions. In autonomous driving, retrieval-augmented planners such as RAD (Wang et al., 2025), RealDrive (Ding et al., 2025) and Driving-RAG (Chang et al., 2025) query similar historical scenarios or trajectories to guide motion generation and decision making when the current region exhibits unfamiliar interaction patterns. In parallel, VLA-based planners such as ORION (Fu et al., 2025), FSDrive (Zeng et al., 2025), and AutoVLA (Zhou et al., 2025) incorporate long-horizon temporal or semantic memory(Tang et al., 2025; Wijaya et al., 2024) to stabilize planning under distribution shift across different environments.

While effective for handling rare or ambiguous situations, these approaches typically treat memory as either scene-level retrieval or short-term temporal context. They do not explicitly model persistent, location-indexed interaction patterns that recur across geographically distinct regions, which are critical for cross-regional planning where differences in yielding, gap acceptance, and merging norms are tied to place rather than to transient scene variations. As a result, retrieval-based or history-based memory alone is insufficient to provide stable and reusable regional specialization across deployments, motivating our region-aware, position-indexed memory formulation in DLE.

3 Problem Description

3.1. Preliminaries

Autonomous driving planning is commonly formulated as a Markov Decision Process (MDP) defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T} \rangle$,

where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, $\mathcal{R}: \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, and $\mathcal{T}: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$ is the transition kernel, i.e., $\mathcal{T}(s' \mid s, a) = \Pr(s_{t+1} = s' \mid s_t = s, a_t = a)$

The objective of a policy $\pi: \mathcal{S} \times \mathcal{A} \to [0,1]$ is to maximize the expected cumulative reward over a planning horizon H:

$$V_\pi(s) = E_\pi[\sum_{i=t}^{H+t} \gamma^{i-t} r_i \mid s_t = s] \quad (1)$$

where $\gamma \in [0,1]$ is the discount factor.

In large-scale deployment, the driving environment is not governed by a single stationary MDP, but by a position-indexed collection of MDPs $\{\mathcal{M}(g)\}$, where each region g induces distinct interaction patterns, yielding behaviors, and transition dynamics. Representing all regions with a single global MDP may fail to capture local traffic variations. Consequently, the global policy π trained on all data $\mathcal{D}$ of size n_σ may deviate from the locally optimal policy $\tilde{\pi}$ in specific regions, expressed as

$$\Delta_s V_\pi = |\Sigma_s V_\pi - \Sigma_s V_{\tilde{\pi}}|, \quad (2)$$

which measures the regional policy performance gap. This work aims to minimize $\Delta_s V_\pi$ without explicitly modeling n_σ.

3.2. Problem description

We study a region-conditioned planning problem for large-scale autonomous driving. Because traffic dynamics and interaction conventions vary across geographic locations, a single global policy may exhibit inconsistent performance when deployed across regions. Let $l \in \mathcal{L}$ denote a region/location index (e.g., a map segment or region ID). For each region l, the environment induces region-dependent dynamics and rewards, defining a region-wise value $V^{(l)}(\pi)$.

Our goal is to obtain region-adaptive behavior while using a single shared set of policy parameters and no on-vehicle online updates. Under these constraints, the policy may only condition on the region index l, and should perform no worse than a global baseline π_b that does not use region conditioning. Formally, letting $\pi(\cdot \mid s, l)$ denote a family of region-conditioned policies, we target, for all $l \in \mathcal{L}$,

$$V^{(l)}\big(\pi(\cdot \mid \cdot, l)\big) \geq V^{(l)}(\pi_b) \quad (3)$$

Equivalently, using the gap to a region-optimal policy $\pi^{*(l)}$,

$$\begin{aligned} &\Delta^{(l)} V(\pi) \coloneqq V^{(l)}\big(\pi^{*(l)}\big) - V^{(l)}(\pi), \\ &\text{s.t. } \Delta^{(l)} V\big(\pi(\cdot \mid \cdot, l)\big) \leq \Delta^{(l)} V(\pi_b). \end{aligned} \quad (4)$$

This formulation captures the key requirement of scalable deployment: improving cross-region consistency while keeping a single shared set of policy parameters and avoiding on-vehicle online adaptation.

3.3. Region-Related Driving Processes

Standard MDP formulations often assume region-invariant dynamics, i.e., the transition kernel $P(s' \mid s, a)$ does not depend on location. In real-world driving, however, road structure, traffic density, and interaction conventions vary across regions, so the same action may induce different next-state distributions. As a result, a policy optimized in one region can generalize poorly to another.

To capture such regional heterogeneity, we consider a family of region-varying MDPs. Let $l_t \in \mathcal{L}$ denote the region/location index at time t. The region-conditioned transition is defined as

$$\begin{aligned} &P^{(l)}(s' \mid s, a) \\ &\triangleq Pr(s_{t+1} = s' \mid s_t = s, a_t = a, l_t = l). \end{aligned} \quad (5)$$

Accordingly, the finite-horizon value function of policy π in region l is

$$V_\pi^{(l)}(s) = \mathbb{E}_\pi \left[\sum_{k=0}^{H} \gamma^k r_{t+k} \;\middle|\; s_t = s, l_t = l \right]. \quad (6)$$

Since different regions exhibit distinct transition dynamics, aggregating multi-region data while ignoring l may yield biased value estimates. On the other hand, naively concatenating raw GPS/absolute position into the state (e.g., $s \leftarrow [s, l]$) enlarges the state space and exacerbates data sparsity. We therefore focus on region-conditioned state representations, enabling region-adaptive decision making without explicitly expanding the high-dimensional positional state.

4 Method

4.1. Overview

Fig.1 illustrates the overall framework of the proposed region-conditioned planning approach, which consists of two decoupled stages: **offline regional memory construction** and **online region-conditioned planning**.

In the offline stage (left), historical driving data collected from large-scale deployments are grouped by geographic regions. For each region, trajectory segments are encoded by a variational representation model to extract region-dependent driving characteristics. The resulting latent distributions are aggregated into compact **map-level memories** $h(l)$, which summarize local traffic dynamics and interaction patterns. These memories are

stored in the HD map and updated only through offline or periodic processes.

In the online stage (right), at each time step t, the vehicle performs region-conditioned planning using real-time observations and retrieved map memory. Based on current perception and local map context, the planner constructs a **dual-layer traffic graph** composed of a vehicle-interaction layer and a road-topology layer. The region-specific memory $h(l_t)$is injected into the road-layer node features, allowing local driving characteristics to influence graph message passing.

The dual-layer graph is processed by a graph neural network to produce a fixed-dimensional representation x_t^{graph}, which jointly encodes real-time interactions and region-dependent context. This representation serves as the sole input to the driving policy, ensuring a consistent interface between training and deployment.

During training, policy optimization is guided by a reinforcement learning objective together with mutual-information-based regularization, which encourages the learned representation to preserve region-dependent cues predictive of future behaviors. At deployment time, the planner performs inference only, without any online parameter updates. When the vehicle enters a new region, adaptation is achieved by retrieving the corresponding map memory $h(l)$and recomputing the graph representation, enabling scalable and robust region-adaptive planning.

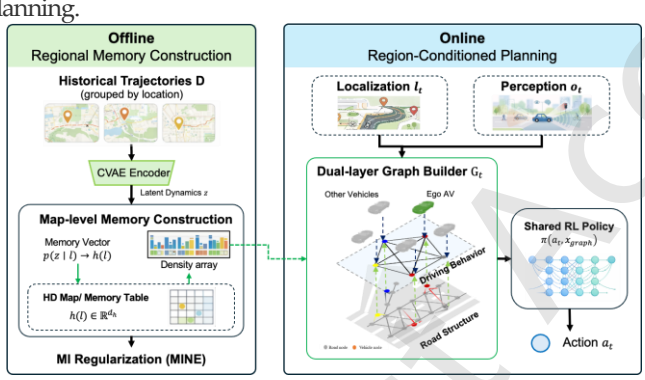


Fig. 1. Framework of the proposed DLE planner. *Left:* Offline construction of map-level regional memory $h(l)$ from historical driving data using a variational encoder. *Right:* Online planning via a dual-layer traffic graph, where region-specific memory is injected into the road-topology layer and propagated through a graph neural network to produce the policy input x_t^{graph}.

4.2. Map-level Historical Memory Encoding

4.2.1 Trajectory Dataset Construction

We collect historical traffic logs from connected vehicles, autonomous driving logs, or roadside perception. Each observed agent record can be represented as

$$v_{i,t} = \{\mathbf{id}_i, t, \mathbf{p}_{i,t}, \mathbf{v}_{i,t}, \mathbf{a}_{i,t}, \mathbf{F}_i\}, \tag{7}$$

where $\mathbf{p}_{i,t}$, $\mathbf{v}_{i,t}$, and $\mathbf{a}_{i,t}$are position, velocity, and acceleration in a global frame, and $\mathbf{F}_i$denotes static attributes, i denotes agent index.

To build training samples for a planning policy, we convert the multi-agent logs into an ego-centric local observation sequence. Specifically, for each ego vehicle at time t, we define its observation as:

$$o_t = f_{\text{obs}}\left(v_{i,t_{i\in\mathcal{N}_t}}, \mathcal{M}(l_t)\right) \tag{8}$$

where $\mathcal{N}_t$is the set of ego and nearby agents within a spatial range, $\mathcal{M}(l_t)$ denotes local map context around location l_t (e.g., lane geometry, connectivity), and $f_{\text{obs}}(\cdot)$ performs coordinate normalization (e.g., transforming agents into the ego Frenet frame and selecting a fixed number of nearest agents).

We then extract a compact planning-relevant feature vector:

$$x_t = f_{\text{seg}}(o_t), \tag{9}$$

where f_{seg} outputs the state used by the planner (e.g., relative longitudinal/lateral offsets, relative velocities, ego speed/acceleration, and optional lane-related attributes). Using a temporal horizon T_h, we form trajectory segments:

$$X_t = \{x_t, x_{t+1}, \dots, x_{t+T_h}\}. \tag{10}$$

The offline training set is

$$\mathcal{D}^{\text{traj}} = \{(X_k, x_k, l_k)\}_{k=1}^{N}, \mathcal{D}^{\text{traj}}(l) = \{(X_k, x_k) \mid l_k = l\}. \tag{11}$$

Here, k indexes trajectory snippet samples constructed offline, where each sample corresponds to a local trajectory segment extracted from long driving logs together with its initial state and region label.

4.2.2 Latent Dynamics Modeling via CVAE

To capture region-dependent traffic dynamics, we introduce a latent variable zand model the conditional transition distribution:

$$p(X \mid x) = \int p(X \mid x, z)\, p(z)\, dz. \tag{12}$$

We adopt a conditional variational autoencoder (CVAE) and approximate the posterior with an encoder $q_\psi(z \mid X, x)$ and a decoder $p_\omega(X \mid x, z)$. The standard variational objective is

$$\mathcal{L}_{\text{cvae}}(\psi, \omega) = \mathbb{E}_{q_\psi(z|X,x)}[-\log\, p_\omega(X|x,z)] + \text{KL}\left(q_\psi(z \mid X, x) \parallel p(z)\right). \tag{13}$$

After training, the encoder induces a location-dependent latent distribution:

$$p(z \mid l) = \mathbb{E}_{(X,x)\sim D^{traj}(l)}[q_\psi(z \mid X, x)] \tag{14}$$

4.2.3 Map Memory Interface

For each location index l, we summarize $p(z \mid l)$into a compact, storable map memory vector $h(l) \in \mathbb{R}^{d_h}$. Concretely, we compute a histogram-based density representation over latent dimensions (fixed bin number and range, L1-normalized), and store the key–value memory $\{l \mapsto h(l)\}$in the HD map.

During online driving, DLE does not run CVAE inference or sample z. It only retrieves the precomputed memory $h(l_t)$using the current location l_t. The memory can be updated offline periodically as new logs arrive.

4.3. Online Regional State Representation via Dual-layer Graph

At time t, DLE constructs a dual-layer graph $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ to encode real-time interactions together with map-level regional priors. The graph consists of three parts:

Vehicle-interaction layer $\mathcal{G}_t^v$**:** nodes represent the ego and nearby agents, with node features in Frenet coordinates (relative offsets, velocities, headings).

Road-topology layer $\mathcal{G}_t^r$ **:** nodes represent local lane segments/waypoints and their connectivity from the HD map. Importantly, each road node is augmented with the offline map memory $h(l)$, which summarizes region-specific driving dynamics for the corresponding location index l.

Cross-layer edges: connect each vehicle to its associated lane segment (e.g., nearest centerline projection) and adjacent lanes, enabling message passing to capture how road topology and regional priors modulate interactions.

4.3.1 Node Features and Memory Injection

Each vehicle node n_i^v is embedded via an MLP:

$$\mathbf{x}_i^v = \mathrm{MLP}_{\theta_v}(n_i^v). \tag{15}$$

For road nodes, we concatenate geometric/topological descriptors with the retrieved memory. Let n_j^r denote the geometric descriptor of the j-th lane segment/waypoint (e.g., relative position, curvature, heading, lane type) and let $l_j \in \mathcal{L}$ be its location index. The road node embedding is defined as:

$$\mathbf{x}_j^r = \mathrm{MLP}_{\theta_r}\left(\left[\, n_j^r - n_0^r \;\|\; h(l^j)\right]\right), \tag{16}$$

where n_0^r is a reference road node and $\|$ denotes concatenation. This design injects persistent regional driving priors directly into the road-topology layer as node attributes, avoiding any online latent inference or sampling.

4.3.2 Graph Message Passing and Graph-level Embedding

We adopt a GraphSAGE-style message passing. For node u at layer k, we aggregate neighborhood information as:

$$\begin{aligned} \mathbf{m}_u^{(k)} &= \sigma\left(\mathrm{MEAN}\mathbf{h}_v^{(k-1)} : v \in \mathcal{N}(u)\right), \\ \mathbf{h}_u^{(k)} &= \sigma\left(W^{(k)}[\mathbf{h}_u^{(k-1)} \parallel \mathbf{m}_u^{(k)}]\right), \end{aligned} \tag{17}$$

where $\mathcal{N}(u)$ denotes the neighborhood, $\sigma(\cdot)$ is a nonlinearity, and $W^{(k)}$ are learnable weights. Message passing is performed on the full dual-layer graph $\mathcal{G}_t$, so that information can propagate across vehicle nodes, road nodes, and cross-layer edges.

After K layers, we obtain a fixed-dimensional online regional embedding by pooling over vehicle-layer node embeddings:

$$x_t^{\mathrm{graph}} = \mathrm{Pool}(\{\mathrm{h}_i^{v,(K)}\}) \in \mathbb{R}^{d_g}. \tag{18}$$

Since road node features already include $h(l)$, the resulting x_t^{graph} encodes both real-time interactions and region-dependent priors in a unified representation.

4.3.3 Dynamic Local Enhancement Mechanism

DLE achieves dynamic local enhancement through memory injection and conditional inference. The offline map memory $h(l)$ learned in Sec. 4.2 is retrieved by location index and attached to road-topology nodes (Sec. 4.3.1). Through dual-layer message passing, this regional prior is propagated to vehicle nodes and summarized into the graph embedding

$$s_t := x_t^{\mathrm{graph}} \tag{19}$$

which serves as the RL state for decision making. Therefore, the "conditioning" on regional characteristics is already completed inside the dual-layer graph representation, and the deployed vehicle performs no online gradient update; the dynamics come from location-indexed memory retrieval and forward inference. During training, we further apply a trajectory–feature mutual information regularization (Sec. 4.5.2) to encourage the local representation to capture region-dependent interaction patterns.

4.4. Training objective & algorithm

4.4.1 Reinforcement Learning Objective

We formulate planning as a standard MDP with a stochastic policy $\pi_\theta(a \mid s_t, c_t)$ conditioned on the local context c_t, and optionally a value function $V_\xi(s_t, c_t)$ or action-value function $Q_\xi(s_t, a, c_t)$. The objective is to maximize the expected discounted return

$$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[\sum_{t=0}^{\infty} \gamma^t \; r_t], \tag{20}$$

where r_t denotes the environment reward.

We adopt a generic policy-improvement scheme that alternates between (i) collecting rollouts under the current policy and (ii) updating the policy parameters by maximizing a surrogate objective:

$$\theta \leftarrow \arg\max_\theta \mathbb{E}_t\left[\mathcal{S}\left(\theta; s_t, a_t, c_t, \hat{A}_t\right)\right], \tag{21}$$

where $\mathcal{S}(\cdot)$ denotes a stable surrogate function, and $\hat{A}_t$ is an advantage (or TD-error–based) estimator computed from V_ξ or Q_ξ. In our implementation, $\mathcal{S}(\cdot)$ can be instantiated by either value-based updates (e.g., Q-learning family) or policy-gradient updates (e.g., PPO family), while keeping the same DLE conditioning and representation modules unchanged.

4.4.2 Historical Information Feedback via Mutual Information Regularization

Here "history" refers to a short trajectory segment collected in a region,

$$Y_t = [s_t, a_t, s_{t+1}, \ldots, s_{t+H}], \tag{22}$$

which is used to regularize the learned local representation. This is not a control-theoretic feedback loop. Instead, it encourages the local representation to capture region-dependent interaction patterns that are predictive of future behaviors.

At each time step, we construct a dual-layer traffic graph and obtain an online representation x_t^{graph} from the GNN. Note that the map-level memory $h(l_t)$ is injected into the road-layer node features during graph construction and is propagated through message passing; hence x_t^{graph} already encodes both real-time interactions and region-specific context, without introducing an additional explicit fusion module.

We employ two mutual-information (MI) regularizes for representation learning. The first term maximizes the MI between the trajectory segment and the representation, $I(Y_t; x_t^{\mathrm{graph}})$, encouraging x_t^{graph} to preserve region-dependent cues that are predictive of future behaviors. The second term maximizes the MI between the current state and the representation, $I(s_t; x_t^{\mathrm{graph}})$, which helps prevent representation collapse and stabilizes training by retaining decision-relevant information about the current step.

Following Mutual Information Neural Estimation (MINE), we introduce critics $T_{w_l}(\cdot)$ and $T_{w_c}(\cdot)$, and estimate MI using the Donsker–Varadhan lower bound:

$$\begin{aligned} \hat{I}_w(A; B) \geq\; & \mathbb{E}_{p(A,B)}[T_w(A, B)] \\ & - \log\, \mathbb{E}_{p(A)p(B)}[\exp\,(T_w(A, B))] \end{aligned} \tag{23}$$

In practice, joint samples (A, B) are obtained from the same trajectory segment (or the same time step), while marginal samples are constructed by shuffling B within a minibatch to break correspondence. The MI regularization loss is defined as

$$\begin{aligned} \mathcal{L}_{\mathrm{MI}}(\theta_{\mathrm{rep}}, w_c, w_l) = & -\hat{I}_{w_c}(s_t; x_t^{\mathrm{graph}}) \\ & - \alpha\, \hat{I}_{w_l}(Y_t; x_t^{\mathrm{graph}}) \end{aligned} \tag{24}$$

where θ_{rep} denotes parameters of the representation modules producing x_t^{graph} (e.g., the dual-layer GNN and encoders), and the critics are updated to maximize the corresponding bounds.

4.4.3 Overall Objective

Let $\mathcal{S}(\cdot)$ denote a stable RL surrogate objective used for policy improvement (Sec. 4.5.1). The overall training objective combines RL optimization and representation regularization:

$$\min_\theta \mathcal{L}_{\mathrm{RL}}(\theta) + \lambda(t)\mathcal{L}_{\mathrm{MI}}\left(\theta_{\mathrm{rep}}, w_c, w_l\right) \tag{25}$$

where $\mathcal{L}_{\mathrm{RL}}(\theta)$ is the negative RL surrogate (e.g., $-\mathbb{E}_t[\mathcal{S}(\cdot)]$ or an equivalent minimization form), and $\lambda(t)$ is a time-dependent

weight.

Annealing for stability. We gradually decay $\lambda(t)$ to zero during training so that MI regularization mainly shapes early representation learning while avoiding late-stage oscillations.

4.4.4 Conditioning Dropout

To ensure robustness when map memory is missing, delayed, or uncertain, we apply memory dropout during training. With probability p (set to 0.5 in our experiments), we mask the memory-related components in road-layer node features (i.e., the injected $h(l_t)$ part) when building the dual-layer graph. This forces the planner to rely on interaction cues and general driving ability when regional memory is unavailable. During deployment, memory injection is enabled whenever $h(l_t)$ is available.

4.4.5 Training and Deployment Process

We present a unified offline training procedure that applies to both value-based and policy-gradient instantiations. The key is that the DLE representation and conditioning are identical, while the RL optimizer is specified by the choice of $\mathcal{S}(\cdot)$.

Algorithm 1 Offline Training of DLE Planner

Inputs: simulator or offline dataset; map memory table $\{h(l)\}$; memory dropout prob. p; MI weight schedule $\lambda(t)$.
Initialize: policy/planner parameters θ, representation parameters $\theta_{\text{rep}} \subseteq \theta$, MI critic parameters (w_c, w_l).
for each training iteration **do**

1. Sample experience. Obtain a batch $\mathcal{B}$ of transitions or short rollouts according to the chosen RL instantiation (on-policy or replay-based).

2. Online representation construction.
For each step $t \in \mathcal{B}$:

- Construct a dual-layer traffic graph G_t from current observations and local map context.
- Inject map memory $h(l_t)$ into road-layer node features and apply memory dropout with probability p(training only).
- Run the dual-layer GNN to compute the interaction-aware representation x_t^{graph}.

3. RL surrogate optimization. Use x_t^{graph} as the planner state and update θ by minimizing $\mathcal{L}_{\text{RL}}(\theta)$.

4. Mutual-information regularization.

- Form trajectory segments $Y_t = [s_t, a_t, \dots, s_{t+H}]$ and pair them with x_t^{graph}.
- Update MI critics (w_c, w_l) to maximize $\hat{I}_{w_c}(s_t; x_t^{graph})$ and $\hat{I}_{w_l}(Y_t; x_t^{graph})$.
- Update representation-related parameters θ_{rep} by minimizing $\lambda(t)\mathcal{L}_{\text{MI}}$.

end for

During deployment, DLE performs conditional forward inference only and does not update parameters.

Algorithm 2 Online Inference of DLE

Inputs: observation o_t, location index l_t, HD map memory $\{h(l)\}$, trained parameters θ.

1. Compute planner state $s_t = f_{\text{state}}(o_t)$.
2. Build the dual-layer graph G_t from $(o_t, \text{local map})$, inject $h(l_t)$ into road-layer node features.
3. Compute $x_t^{\text{graph}} = f_{\text{gnn}}(G_t)$.
4. Output action $a_t \sim \pi_\theta(\cdot \mid x_t^{\text{graph}})$ (or $a_t = \arg\max_a \pi_\theta(a \mid x_t^{\text{graph}})$).

5 Experiment

To demonstrate the effectiveness of the proposed Dynamically Local-Enhancement (DLE) planner, we design a two-part experimental evaluation: (1) verifying the extraction of region-specific driving features, (2) evaluates the planner's capability to enhance decision making in both simulated environments and real-world driving experiments.

5.1. Cross-regional Feature Extraction and Analysis

5.1.1 Objective

This experiment aims to evaluate the capability of the proposed framework to extract region-specific 305 driving features. The goal is to verify that as the amount of data increases, the implicit regional driving 306 features gradually converge and can effectively distinguish different regional driving patterns.

5.1.2 Experiment Setup and Dataset

To evaluate cross-regional driving characteristics, we use naturalistic trajectory datasets collected from diverse environments. Specifically, the HighD dataset (Krajewski et al., 2018) provides highway scenes from Germany, while the NGSIM dataset(Federal Highway Administration (FHWA), 2007) contains urban driving data from the United States, including US-101 (Los Angeles), I-80 (Emeryville), Peachtree Street (Atlanta), and Lankershim Boulevard (Los Angeles). All datasets provide vehicle trajectories sampled at 10~Hz with lane-level context, enabling analysis of regional driving styles. In this study, these datasets are used to analyze and validate region-dependent interaction patterns, while closed-loop planning performance is evaluated separately in CARLA under controlled and repeatable conditions.

The state space $x_{i,t}$ comprises the ego vehicle's relative position to the preceding vehicle, ego velocity, preceding vehicle velocity, and ego acceleration. These variables are organized into sequential representations of size 10×10, forming a 100-dimensional embedding. A global feature extractor is first trained on the combined dataset, after which region-specific latent maps are initialized and iteratively updated using 100,000 randomly sampled frames per region. This setup allows us to observe how regional features emerge and stabilize as additional data are incorporated.

To validate the representational quality of the extracted features, a hold-out subset from each region is used for scene recognition. A separability criterion is applied to measure how well the learned features discriminate between regions, thereby verifying their ability to capture distinctive local traffic patterns and support efficient spatial partitioning.

5.1.3 Feature Extraction Results

To quantify the distinctiveness of regional driving behaviors, we define the Degree of Distinction (DoD) as

$$\text{DoD} = \frac{\text{JSD}(z_x \parallel z_y)}{\max(\text{JS}_x, \text{JS}_y)}, \tag{26}$$

where $\text{JSD}(z_x \parallel z_y)$ denotes the Jensen–Shannon divergence between latent features of regions x and y.

The intra-region divergences JS_x and JS_y are computed by comparing each region's latent features with its own augmented feature distribution, i.e.,

$$\text{JS}_x = \text{JSD}(z_x \parallel z_x^a), \text{JS}_y = \text{JSD}(z_y \parallel z_y^a) \tag{27}$$

where z_x^a and z_y^a represent augmented samples from the same region. A higher DoD value thus indicates stronger dissimilarity between regional driving behaviors after accounting for their internal consistency.

In this experiment, each region is treated as an aggregated unit to emphasize cross-regional variation, rather than fine-grained road-level indices used later for policy enhancement. Table 1 summarizes the results. The DoD values are low for intra-region comparisons (e.g., 0.1 for HighD, 0.3 for I-80), reflecting the stability of features within a region, but substantially higher for cross-region pairs (e.g., 9.5 between HighD and I-80), confirming significant distributional differences. Consistently, values above~1 indicate heterogeneous driving styles, validating that the extracted latent features capture meaningful regional distinctions that can serve as a basis for subsequent policy adaptation.

Table 1. DOD across different regions

Target Region	highD	US-101	I-80	PSA	LBL
highD	0.1	/	/	/	/
US-101-LosAngeles	41.8	0.1	/	/	/
I-80-Emeryville	99.5	110.9	0.3	/	/
Atlanta (PSA)	62.8	63.1	29.6	0.5	/
Lankershim(LBL)	138.1	141.4	60.3	27.6	0.6

Fig.2 illustrates how the driving environment is progressively partitioned from a unified region into finer road-level segments. With limited data, all areas are treated uniformly, as the available features lack sufficient variation for differentiation. As more data are incorporated, the separability index between regions surpasses 1, allowing the environment to be divided into distinct characteristic zones. The resulting feature distributions exhibit clear boundaries across regions, confirming that the learned representations effectively capture heterogeneous driving patterns.

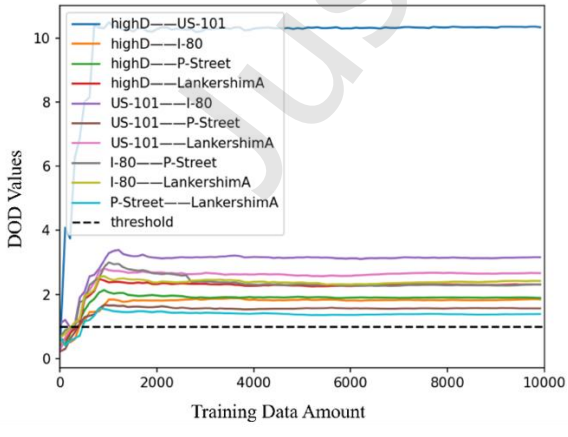


Fig. 2 Growth of regional feature separability (DoD) with increasing data volume.

5.2. Performance of Dynamic Local-Enhancement (DLE) Planner

5.2.1 Region-related Test Scenarios Design

To evaluate cross-location generalization under controlled yet realistic traffic conditions, we design a structured scenario generation protocol in the CARLA simulator. Experiments are conducted in Town03, where four representative locations are selected, as illustrated in Fig.3 : highway, roundabout, unprotected left turn, and merge-in.

For each location, we define a task-specific route segment that isolates the dominant interaction pattern of interest. Surrounding traffic is generated by instantiating a variable number of background vehicles, whose initial positions and velocities are randomized across rollouts to induce diverse traffic configurations. This initialization-level randomization ensures sufficient variability in local interactions while preserving the underlying road topology.

All background vehicles are controlled by a rule-based autonomous driving stack. Each vehicle follows a reference path computed by graph-based route search on the HD map and executes standard driving behaviors, including lane following, obstacle avoidance, yielding at intersections, and traffic light compliance. Importantly, while the initial conditions are randomized, the driving style parameters of background vehicles are location-dependent and fixed within each location. These parameters include, but are not limited to, speed limit compliance, lane-change aggressiveness, and collision-checking thresholds and probabilities. This design allows us to systematically evaluate whether a planner can leverage region-dependent structure to improve performance, while remaining robust to stochastic variations in traffic density and initial conditions.

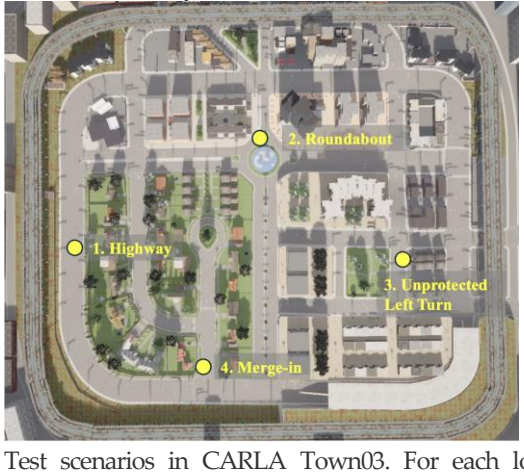


Fig. 3 Test scenarios in CARLA Town03. For each location, background vehicles are randomly initialized in number and state, while following rule-based driving policies with location-specific, fixed behavior parameters, resulting in distinct yet consistent interaction patterns across locations.

5.2.2 Baselines

We consider a set of baseline planners designed to examine whether cross-regional performance can be achieved through data mixing, capacity scaling, or online adaptation, without explicit region-aware enhancement.

Global PPO baselines. Following recent scalable RL planners 1, we implement two global PPO baselines.

(1) Small-capacity global PPO (SM) is trained jointly on data from all locations, with the same parameter budget as DLE. This baseline evaluates whether simple data aggregation across regions is sufficient for cross-regional generalization under a fixed model capacity.

(2) Capacity-expanded global PPO (GM) increases model capacity by introducing location experts and scales its parameter count proportionally with the number of locations. This baseline tests whether cross-regional performance can be recovered by brute-force capacity scaling.

Mixture-of-Experts (MoE). We further include a Mixture-of-Experts baseline that conditionally activates location-dependent experts, inspired by recent region-aware planners such as AnyD (Zhu et al., 2023). This baseline represents an explicit specialization strategy that balances global generalization with region-specific policy components.

Test-Time Adaptation (TTA). To compare against online adaptation strategies, we implement a test-time adaptation baseline following Centaur (Sima et al., 2025), which updates the planner during deployment using on-policy signals under the same observation and action interfaces. This baseline evaluates whether online optimization alone can effectively handle regional variation.

All baselines are implemented in our CARLA closed-loop benchmark and evaluated under identical scenario distributions using NuPlan-style metrics. This comparison allows us to distinguish the effects of data mixing, model capacity expansion, and online adaptation from the proposed region-aware enhancement mechanism.

5.2.3 DLE Planner Settings

The state space includes the positions, speeds, and orientations of the seven nearest vehicles in the lane coordinate system. The action set is discrete, consisting of acceleration, deceleration, lane change, and hold.

Both training and test scenarios share the same reward function composed of terminal rewards and step-wise shaping terms. The terminal reward assigns a large penalty for collision ($r_{\text{coll}} = -50$), a positive reward for successfully reaching the goal ($r_{\text{goal}} = +50$), and a penalty for stuck situations ($r_{\text{stuck}} = -10$), each of which immediately terminates the episode. During normal execution, the step-wise reward consists of three components: a progress term proportional to the reduction in distance to the goal,

$$r_{\text{prog}} = 0.5(d_{t-1} - d_t), \tag{28}$$

a bounded speed reward that linearly maps the ego speed to the range $[-0.1, 0.2]$ using a predefined maximum speed, and a constant time penalty $r_{\text{time}} = -0.1$ to discourage unnecessarily long trajectories. This reward design encourages safe, goal-directed, and efficient driving while enforcing strong terminal constraints on collisions and task completion.

5.2.4 Performance Metrics

We evaluate planner performance using a set of complementary metrics that capture safety, efficiency, and driving quality.

Reward. The cumulative reward reflects overall task performance under the unified reward formulation used during training, aggregating progress, safety, and comfort-related terms (scaled to $[0, 100]$for consistent comparison across methods).

Non-Collision Rate. The non-collision rate measures the percentage of episodes completed without any collision, serving as a primary safety indicator for closed-loop driving evaluation.

Progress. Progress is defined as the traveled distance toward the route endpoint. It quantifies the planner's ability to complete the driving task under dynamic traffic conditions and is particularly informative when episodes terminate early due to safety violations.

Average Time-to-Collision (TTC). Average TTC measures the mean time-to-collision over the episode, computed from relative states of surrounding vehicles. Higher values indicate safer interaction margins and more conservative decision-making.

Speed. The speed metric reports the average driving speed (scaled to $[0, 100]$ for consistent comparison across methods), reflecting traffic efficiency.

All metrics are computed over N randomized rollouts per location.

5.2.5 Results and Analysis

Table 2 reports the closed-loop planning performance in CARLA across multiple driving scenarios using NuPlan-style metrics. We compare the proposed DLE planner against a set of representative baselines, including global data-mixed policies, region-specific local policies, capacity-scaled mixture-of-experts, and test-time adaptation methods, under comparable parameter budgets. The parameter budget is normalized such that a standard PPO planner corresponds to 1× model capacity.

Global and Local PPO Baselines. Global PPO (4× parameters, trained on data mixed from all four scenarios) achieves a solid but not optimal overall performance. This indicates that increasing capacity and training on aggregated data can provide reasonable coverage, yet it still struggles to consistently capture heterogeneous interaction conventions across scenarios, leaving room for improvement in both stability and efficiency.

The Local PPO baselines policies use a compact budget (1×) and reveal a clear specialization–generalization trade-off. When trained on a single scenario, local policies can be highly specialized in that domain, but they often exhibit degraded safety and progress when evaluated across the full benchmark distribution, such as Safety 39.21% and Progress 52.45% in the roundabout-trained policy. Training a Local PPO on mixed data ("All") does not resolve this issue either: while its reward (69.88) remains comparable, it suffers from noticeably lower safety (82.84%) and progress (87.70%) than the Global PPO, suggesting that simply mixing data is insufficient for small-capacity planners to balance global generalization and region-specific behaviors.

Mixture-of-Experts and Test-Time Adaptation Baselines. The Mixture-of-Experts (MoE) baseline (4× parameters) combines scenario-specialized experts and achieves the highest reward (79.58) and the best progress (93.68%) among all methods. However, this gain comes with explicit capacity scaling and does not translate into the best safety-related outcomes: its safety (89.71%) is below both Global PPO and DLE, and its TTC (33.07 s) is also lower than DLE, indicating less conservative interaction handling despite strong task completion.

The Test-Time Adaptation (TTA) baselines apply online updates during evaluation (50 consecutive runs per scenario). Starting from the Global PPO, TTA yields Reward 69.06 with Safety 89.22% and Progress 92.24%, which is not consistently better than the non-adaptive Global PPO in this benchmark. When initialized from a Local PPO, TTA can raise reward to 75.34, but its performance is unstable across metrics, e.g., Safety drops to 69.61% and Progress to 78.65%, reflecting the difficulty of achieving reliable adaptation

under limited online experience and without strong region-aware structure.

Performance of DLE. Under a compact budget comparable to a single PPO policy, DLE achieves the most balanced cross-scenario performance. Compared to Global PPO, DLE improves reward by +5.40 and increases TTC by +3.81 s, while also improving safety (+1.45%) and progress (+1.09%). Compared to MoE, DLE attains similar completion quality (Progress 93.51% vs. 93.68%) while delivering higher Safety (+2.43%) and substantially higher TTC (+8.94 s) with far fewer parameters, indicating more cautious and stable interaction decisions under a deployment-friendly model size. Compared to TTA, DLE provides consistent gains without relying on online parameter updates during deployment-time evaluation.

Overall, these results support that explicit region-conditioned enhancement (via retrieved memory and structured interaction encoding) is a more effective and deployable mechanism for balancing global generalization and localized adaptation than either brute-force capacity scaling, naive data mixing, or online test-time adaptation.

Table 2. Test Results of closed-loop evaluation in CARLA

Policy	Training Scenario	Params	Reward ↑	Safety	Progress	TTC ↑ (s)	Speed ↑
Global PPO	All	4	71.67	**90.69**	92.42	38.2	32.84
Local PPO	All	1	69.88	82.84	87.70	**44.65**	35.64
	Hwy	1	68.16	75.98	84.17	29.08	50.63
	Round	1	51.46	39.21	52.45	22.31	**85.64**
	Left	1	54.96	46.08	58.1	20.94	65.48
	Straight	1	68.86	69.61	78.53	32.68	62.54
Mix-of-Expert (MOE)	All	4	**79.58**	89.71	**93.68**	33.07	55.11
Test-time Adaptation (TTA)	All (Global)	4	69.06	89.22	92.24	32.56	41.60
	All (Local)	1	75.34	69.61	78.65	43.21	64.14
DLE (Ours)	All	1+	**77.07**	**92.14**	**93.51**	**42.01**	**40.97**

5.2.6 Ablation Study

As shown in Table.3, To examine the contribution of representation regularization and latent regional encoding, we conduct an ablation study with three variants under identical training and evaluation settings: (i) the full DLE model, (ii) DLE without mutual-information regularization, and (iii) DLE without mutual-information regularization and without latent regional encoding.

Removing the MI regularization results in a consistent performance drop. Compared to the full DLE model, DLE w/o MI reduces Reward from 77.07 → 73.66 (−3.41), with noticeable degradations in Safety (92.14 → 89.33), Progress (93.51 → 91.90), TTC (44.37 → 41.92 s), and Speed (40.97 → 37.38). This indicates that MI regularization helps stabilize and align the learned region-conditioned representation with region-specific interaction dynamics.

When MI regularization is removed and the latent environmental encoding (LE) is further disabled, performance deteriorates further. DLE w/o MI, LE decreases Reward to 70.62 and TTC to 39.01 s, with a substantial reduction in speed (31.93). Safety slightly rebounds compared to the MI-only ablation (90.20 vs. 89.33), but remains below the full model (92.14), suggesting that removing LE weakens the efficiency and temporal stability of planning even if it does not monotonically affect a single safety score.

Overall, the ablation results show that MI regularization and latent encoding contribute complementary benefits, and the full DLE model yields the best trade-off among safety, progress, temporal margin, and efficiency under a compact parameter budget.

Table 3. Ablation Study Results for DLE Planner

Policy	Reward	Safety	Progress	TTC	Speed
DLE	**77.07**	**92.14**	**93.51**	**44.37**	**40.97**
DLE w/o MI	73.66	89.33	91.90	41.92	37.38
DLE w/o MI, LE	70.62	90.20	92.50	39.01	31.93

5.3. Real-World Case Study

5.3.1 Vehicle platform

The test platform is an autonomous vehicle equipped with full perception sensors and steer-by-wire control, as shown in Fig. 4. The onboard computer integrates an Intel Xeon CPU and an NVIDIA GTX GPU for real-time processing. The perception suite includes cameras, a 40-line LiDAR, millimeter-wave radar, an inertial navigation unit, and GPS.

Fig.4 Test vehicle and its sensor configuration

In the real-world experiments, a high-precision map was used. The map follows the OpenDRIVE format and the WGS84 (World Geodetic System) coordinate system, providing global path generation, road topology, and connection rules for each road segment.

5.3.2 Test Scenarios and Baseline

The real-world test route (Fig.5) is located in an urban industrial district and consists of two nested driving loops designed to capture diverse interaction patterns under controlled and repeatable conditions. The inner loop is a 1.0-km circuit formed by Taihe 3rd Street, Boxing 2nd Road, and Boxing 3rd Road, covering one-way curved streets, an industrial-park entrance with frequent vehicle in-and-out traffic, two signalized intersections, a multi-lane arterial road, and roadside parking zones.

The outer loop extends the inner route by continuing along Boxing 3rd Road through a major merge point, then passing Taihe 1st Street and Bolin Road before returning to the start, with a total length of 2.1 km. Together, the two loops provide a compact yet diverse closed-loop testbed that includes intersections, merging, parking, and varying traffic densities, enabling systematic evaluation of region-dependent interaction behaviors and planning performance.

Fig.5 Real-world test route in an urban industrial area. The inner loop (shown in green) is a 1.0-km circuit covering intersections, an industrial-park entrance, and roadside parking zones, while the outer loop (shown in yellow) extends the route to 2.1 km by including major merging segments and arterial roads.

During testing, the autonomous vehicle continuously navigated the route under natural traffic conditions. Environmental data, including surrounding vehicles and pedestrians, were captured by onboard sensors and roadside monitoring systems without manual disengagements, ensuring that the recorded behaviors reflect genuine traffic dynamics.

The baseline policy, similar to LM_{12}, uses the same training data as the DLE planner but without leveraging region-specific or geographical features. This setup reflects common industrial learning-based planners, such as end-to-end (E2E) systems.

5.3.3 Regional Feature Extraction and Analysis

To analyze region-dependent driving behaviors in real-world deployment, we perform unsupervised clustering on the learned regional feature representations extracted by the DLE framework. For each ego-vehicle trajectory segment, we compute its regional latent feature $z$and corresponding fused planning representation and apply t-SNE for dimensionality reduction followed by k-means clustering (with $k = 4$) to group segments with similar interaction patterns.

Fig.6. visualizes the resulting clusters along the test route. Most road segments belong to a dominant cluster (cluster 2, 3), indicating stable and homogeneous interaction patterns. In contrast, several localized zones (highlighted by red circles) exhibit distinct clusters (cluster1,4), revealing spatially concentrated behavioral deviations. These regions correspond to areas where surrounding traffic exhibits different interaction dynamics, such as frequent merging, stopping, or obstruction. Therefore, two representative segments (BJ1–BJ2) from these anomalous regions were selected for detailed case analysis.

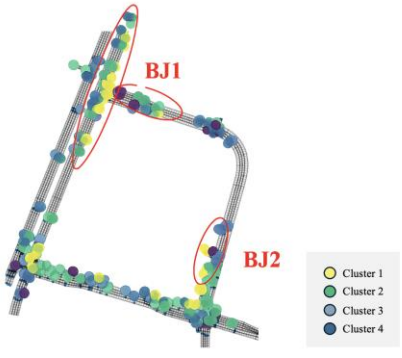


Fig. 6 Clustering of regional driving characteristics. Most road segments belong to dominant clusters (Clusters 2 and 3), indicating stable interaction patterns, while red-circled zones show distinct clusters (Clusters 1 and 4), revealing localized behavioral deviations.

Segment BJ1 (Industrial Park Entrance). As shown in Fig.7, segment BJ1 is located near the main entrance of an industrial park where vehicles frequently merge into the main lane from side roads. This region is dominated by the purple and blue clusters, which exhibit high deceleration frequency and short time-to-collision (TTC) events. These clusters correspond to repeated braking and yielding behaviors triggered by cross-traffic from the park entrance.

In contrast, a nearby low-traffic entrance in the same road section belongs to the dominant green cluster and shows significantly fewer interaction events. This contrast demonstrates that the extracted regional features capture dynamic interaction patterns rather than static road geometry.

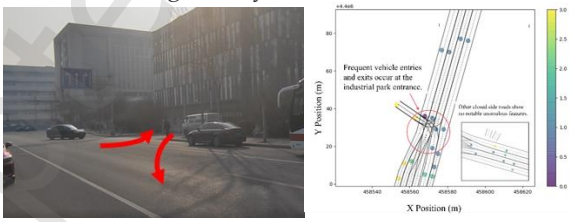


Fig. 7 Regional driving characteristics of segment BJ1.

Segment BJ2 (Illegal Parking and Lane Blockage). Segment BJ2, illustrated in Fig.8, corresponds to a road section near social and commercial venues where illegal roadside parking frequently blocks one or more lanes. This segment is primarily assigned to the purple cluster, which is associated with low average speed, high stop frequency, and prolonged lateral deviation due to lane changes.

These cluster characteristics are consistent with on-site observations of delivery vehicles temporarily occupying driving lanes, forcing ego vehicles to slow down or re-route. The clustering thus correctly identifies a localized behavioral regime dominated by recurrent obstructions.

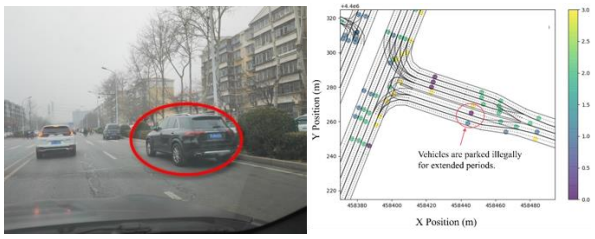


Fig.8 Regional driving characteristics of segment BJ2.

Overall, the clustering-based analysis reveals that the learned regional features systematically partition the driving environment into distinct interaction regimes, corresponding to merging-dominated zones, obstruction-heavy areas, and stable free-flow regions. These regions align with real-world operational challenges for planning, such as aggressive merging, frequent braking, and lane blockage.

Importantly, these high-variance clusters coincide with locations where the baseline planner exhibits degraded performance and where the DLE-enhanced planner achieves the largest improvements in safety and comfort metrics (as reported in Section

4.3.4). This demonstrates that the extracted regional features are not merely descriptive but provide actionable signals for region-aware planning and adaptive policy enhancement.

5.3.4 Planning Results and Analysis

Segment BJ1. Fig.9 (a) compares the autonomous vehicle's driving behavior under two driving policies in a cut-in scenario (corresponding to region BJ1). In the baseline policy, the ego vehicle maintains 25km/h until the cut-in occurs at 6--7s, then brakes sharply between 9--11s with a deceleration of 2.1m/ s^2, reducing speed to 13.4km/h. Although the event is safely passed, the delayed reaction causes discomfort and increases risk.

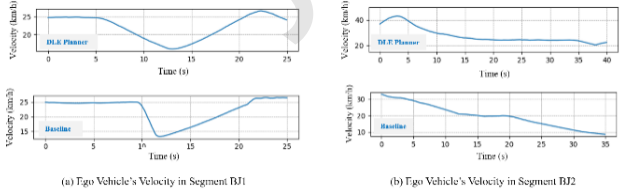
In contrast, the proposed DLE planner enhanced with local information anticipates the cut-in, gradually decelerating to 15km/h by 6s with mild braking (0.3m/s^2), then re-accelerating after the conflict. This shows earlier perception of interaction intent and smoother, safer control.

Segment BJ2. Fig 9.(b) shows results in a section with frequent illegal parking (region BJ2). In the baseline policy, the ego vehicle decelerates before the curve and initiates a late left lane change at 15, leading to a safety-driver takeover at 20s due to insufficient lateral clearance.

Under the proposed DLE planner, the system anticipates right-lane obstruction and initiates an early lane change about 15m before the curve, avoiding the conflict zone entirely and maintaining stable speed without disengagements.

Table 4 summarizes the overall planning performance in real-world testing. Compared with the baseline planner, the proposed DLE planner achieves a 14.8% increase in average driving speed (from 27 km/h to 31 km/h), while reducing disengagement count by 85.7% (from 7 to 1), indicating substantially improved operational stability. In addition, the progress per disengagement increases from 0.606 km to 4.215 km, reflecting a significantly longer uninterrupted driving distance.

Beyond efficiency and reliability, DLE also improves driving comfort. The peak longitudinal deceleration is reduced from 1.47 m/s² to 0.52 m/s², and the peak steering angle decreases from 162° to 97°, demonstrating smoother braking and less aggressive steering behavior under comparable traffic conditions. Overall, these results confirm that integrating region-aware driving characteristics enables the DLE planner to simultaneously improve safety, efficiency, and comfort in real-world deployment.



(a) Ego Vehicle's Velocity in Segment BJ1 (b) Ego Vehicle's Velocity in Segment BJ2

Fig. 9 Comparison of planning results in cut-in scenario

Table 4. Planning Performance in Real-world Testing

Metrics	Baseline	DLE Planner
Average speed(km/h)	27	**31**
Disengagement count	7	**1**
Peak longitudinal deceleration (m/s2)	1.47	**0.52**
Peak steering angle(°)	162	**97**
Progress/per Disengagement (km)	0.606	**4.215**

6 Conclusion

This paper addressed the challenge of scaling autonomous driving planners across regions with heterogeneous traffic dynamics under fixed on-board computational constraints. We proposed the Dynamically Local-Enhancement (DLE) planner, which achieves region-adaptive behavior through conditional inference rather than parameter scaling or online policy optimization.DLE decouples global driving competence from region-specific adaptation by conditioning a shared reinforcement learning planner on region-aware representations. Long-term regional characteristics are distilled into map-level memory, while real-time interactions are captured by a dual-layer traffic graph that jointly models vehicle interactions and road topology. This design enables dynamic adaptation via location-indexed retrieval without increasing policy capacity.

Extensive closed-loop experiments in CARLA and real-world testing demonstrate that DLE consistently outperforms global, local, mixture-of-experts, and test-time adaptation baselines under comparable parameter budgets, achieving improved safety, progress, and driving stability. These results suggest that memory-based region-aware enhancement provides a practical and effective paradigm for scaling autonomous driving planners to large and diverse deployment environments.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) (52402501, 52472449, 52394264) and the Postdoctoral Fellowship Program of China Postdoctoral Science Foundation (GZC20240823).

Author contributions

Nanshan Deng and Weitao Zhou contributed equally to this work.

Nanshan Deng and Weitao Zhou: Conceptualization, Methodology, Formal analysis, and Writing – original draft.

Yifei He, Qian Cheng, and Junze Wen: Investigation, Data curation, and experimental support.

Bo Zhang, Chunyang Liu, Jianmo He, Xiang Sha, and Zelin Qian: Real-world experiment support, data collection, and system deployment.

Kun Jiang and Mengmeng Yang: Validation, technical discussion, and Writing – review & editing.

Zhong Cao: Methodological discussion and Writing – review & editing.

Diange Yang: Supervision, Project administration, and Writing – review & editing.

All authors reviewed and approved the final manuscript.

Replication and Data Sharing

The source codes and replication package are available on ETS data at Data DOI: https://doi.org/10.26599/ETSD.2026.9190007

Declaration of competing interest

The authors have no competing interests to declare that are relevant to the content of this article.

References

Cao, Z., Jiang, K., Zhou, W., Xu, S., Peng, H., Yang, D., 2023. Continuous improvement of self-driving cars using dynamic confidence-aware reinforcement learning. Nat. Mach. Intell. 5, 145–158.

Chang, C., Ge, J., Guo, J., Guo, Z., Jiang, B., Li, L., 2025. Driving-RAG: Driving Scenarios Embedding, Search, and RAG Applications.

Deng, N., Cao, Z., Zhou, W., Jiang, K., Yang, D., 2021a. Adapt the Driving Policy to Local Traffic before Entering the New Area, in: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC). IEEE, pp. 798–803.

Deng, N., Jiang, K., Cao, Z., Zhou, W., Yang, D., 2021b. Decision-Oriented Driving Scenario Recognition Based on Unsupervised Learning, in: CICTP 2021. pp. 564–573.

Diehl, C., Karkus, P., Veer, S., Pavone, M., Bertram, T., 2025. Lord: Adapting differentiable driving policies to distribution shifts, in: 2025 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 7036–7043.

Ding, W., Veer, S., Chen, Y., Cao, Y., Xiao, C., Pavone, M., 2025. RealDrive: Retrieval-Augmented Driving with Diffusion Models.

Federal Highway Administration (FHWA), 2007. Next Generation Simulation (NGSIM) Vehicle Trajectories and Supporting Data.

Fu, H., Zhang, Diankun, Zhao, Z., Cui, J., Liang, D., Zhang, C., Zhang, Dingyuan, Xie, H., Wang, B., Bai, X., 2025. ORION: A Holistic End-to-End Autonomous Driving Framework by Vision-Language Instructed Action Generation.

Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., others, 2022. Lora: Low-rank adaptation of large language models. ICLR 1, 3.

Jaisankar, K., Tandel, S., 2025. Align2Act: Instruction-Tuned Models for Human-Aligned Autonomous Driving. https://doi.org/10.48550/arXiv.2510.10503

Jeon, D., Kim, T., Cho, S., Seo, M., Choi, J., 2025. TTA-DAME: Test-Time Adaptation with Domain Augmentation and Model Ensemble for Dynamic Driving Conditions. ArXiv Prepr. ArXiv250812690.

Krajewski, R., Bock, J., Kloeker, L., Eckstein, L., 2018. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC). IEEE, pp. 2118–2125.

Lin, H., Zhang, Y., Ding, W., Wu, J., Zhao, D., 2025. Model-Based Policy Adaptation for Closed-Loop End-to-End Autonomous Driving. https://doi.org/10.48550/arXiv.2511.21584

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., others, 2022. Training language models to follow instructions with human feedback. Adv. Neural Inf. Process. Syst. 35, 27730–27744.

Piriyajitakonkij, M., Sun, M., Zhang, M., Pan, W., 2024. TTA-Nav: Test-time Adaptive Reconstruction for Point-Goal Navigation under Visual Corruptions.

Schwall, M., Daniel, T., Victor, T., Favaro, F., Hohnhold, H., 2020. Waymo public road safety performance data. ArXiv Prepr. ArXiv201100038.

Segu, M., Schiele, B., Yu, F., 2023. Darth: Holistic test-time adaptation for multiple object tracking, in: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9717–9727.

Sima, C., Chitta, K., Yu, Z., Lan, S., Luo, P., Geiger, A., Li, H., Alvarez, J.M., 2025. Centaur: Robust end-to-end autonomous driving with test-time training. ArXiv Prepr. ArXiv250311650.

Sójka, D., Cygert, S., Twardowski, B., Trzciński, T., 2023. Ar-tta: A simple method for real-world continual test-time adaptation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3491–3495.

Tang, X., Yang, M., Wen, T., Jia, P., Cui, L., Luo, M., Sheng, K., Zhang, B., Jiang, K., Yang, D., 2025. PriorFusion: Unified integration of priors for robust road perception in autonomous driving. Commun. Transp. Res. 5, 100229. https://doi.org/10.1016/j.commtr.2025.100229

Wang, Y., Liu, Q., Jiang, Z., Wang, T., Jiao, J., Chu, H., Gao, B., Chen, H., 2025. RAD: Retrieval-Augmented Decision-Making of Meta-Actions with Vision-Language Models in Autonomous Driving.

Wijaya, B., Yang, M., Jiang, K., Zhang, W., Yang, D., 2024. Exploring the application of blockchain technology in crowdsource autonomous driving map updating. Commun. Transp. Res. 4, 100140. https://doi.org/10.1016/j.commtr.2024.100140

Xu, S., Zidek, R., Cao, Z., Lu, P., Wang, X., Li, B., Peng, H., 2021. System and experiments of model-driven motion planning and control for autonomous vehicles. IEEE Trans. Syst. Man Cybern. Syst. 52, 5975–5988.

Xu, S., ZihaoLian, Tan, M., Liu, L., Zhang, Z., Zhao, P., 2025. Test-time Adapted Reinforcement Learning with Action Entropy Regularization, in: Forty-Second International Conference on Machine Learning.

Zeng, S., Chang, X., Xie, M., Liu, X., Bai, Y., Pan, Z., Xu, M., Wei, X., Guo, N., 2025. FutureSightDrive: Thinking Visually with Spatio-Temporal CoT for Autonomous Driving.

Zhou, W., Cao, Z., Deng, N., Liu, X., Jiang, K., Yang, D., 2022. Dynamically conservative self-driving planner for long-tail cases. IEEE Trans. Intell. Transp. Syst. 24, 3476–3488.

Zhou, Z., Cai, T., Zhao, S.Z., Zhang, Y., Huang, Z., Zhou, B., Ma, J., 2025. AutoVLA: A Vision-Language-Action Model for End-to-End Autonomous Driving with Adaptive Reasoning and Reinforcement Fine-Tuning.

Zhu, R., Huang, P., Ohn-Bar, E., Saligrama, V., 2023. Learning to drive anywhere. ArXiv Prepr. ArXiv230912295.

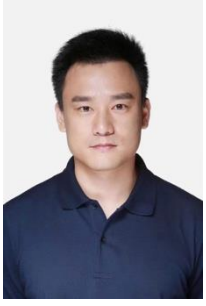
Author biography

Nanshan Deng received the B.S degree and Ph.D. degrees in automotive engineering from Tsinghua University, Beijing, China, in 2018 and in 2025. He is currently a Post-Doctoral Researcher with Tsinghua University. His research interests include autonomous vehicle, transfer reinforcement learning, and meta learning.

Weitao Zhou received received the B.S. and M.S. degrees in automotive engineering from Beihang University, and Ph.D. degrees in automotive engineering from Tsinghua University, Beijing, China, in 2023. He is currently a Post-Doctoral Researcher with Tsinghua University. His research interests include autonomous driving, reinforcement learning, and open-world learning.

Yifei He received the B.S. degree in automation from Tsinghua University, Beijing, China, in 2024. He is currently a Master Student with Tsinghua University. His research interests include autonomous driving.

Qian Cheng received the bachelor's degree from the Department of Automotive Engineering, Beijing Institute of Technology, Beijing, China, in 2016, and the master's degree from the Department of Mechanical Engineering, Karlsruhe Institute of Technology, Karlsruhe, Germany, in 2019. He is currently pursuing the Ph.D. degree with the School of Vehicle and Mobility. His research interests include autonomous vehicles, environment modeling, trajectory prediction, and uncertainty analysis.



Bo Zhang Zhang Bo received the B.S. degree from the College of International Software, Wuhan University in 2005, and the M.S. degree from the Institute of Software, Chinese Academy of Sciences in 2009. He is mainly engaged in research on human-computer interaction and artificial intelligence.

Junze Wen received the B.S. degree in mechanical engineering from Tsinghua University, Beijing, China, in

2022, where he is currently pursuing the Ph.D. degree with the School of Vehicle and Mobility. His research interests include perception uncertainty, environment cognition, and understanding of autonomous driving.

Chunyang Liu is currently a principal engineer at the Autonomous Driving Department of Didi Chuxing Technology. Dr. Liu received his B.S. from Shanghai Jiao Tong University and Ph.D. degree from the University of Technology Sydney, Australia. His research interests include autonomous driving, machine learning and data mining.

Jianmo He received the B.S. degree in automatic engineering from Beihang University and M.S. in electrical and systems engineering from Washington University in St. Louis in 2017. He is currently a staff engineer in Didi Autonomous driving. His research interests include autonomous driving and large language models.

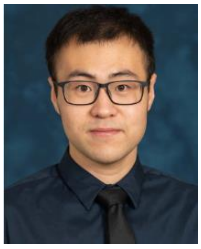
Xiang Sha received his B.S. and M.S. degrees in Electronic and Information Engineering from Huazhong University of Science and Technology, Wuhan, China, in 2019.
He is currently a staff algorithm engineer at DiDi Chuxing.
His research interests primarily focus on low-speed driving scene understanding and unstuck strategies for autonomous driving..

Zelin Qian received the B.S. degree from the School of Transportation and Logistics, Southwest Jiaotong University, Chengdu, China, in 2022 and the M.S. degrees in automotive engineering from Tsinghua University He is currently an engineer in DiDi. His research interests include intelligent decisionmaking and motion planning of autonomous driving.

Kun Jiang received the B.S. degree in mechanical and automation engineering from Shanghai Jiao Tong University, Shanghai, China, in 2011, and the master's degree in mechatronics system and the Ph.D. degree in information and systems technologies from the University of Technology of Compiègne (UTC), Compiègne, France, in 2013 and 2016, respectively. He is currently an Associate Research Professor with Tsinghua University, Beijing, China.

His research interests include autonomous vehicles, high-precision digital maps, and sensor fusion.

Mengmeng Yang received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2018.She is currently an Associate Research Professor with Tsinghua University, Beijing, China. Her research interests include autonomous vehicles, high-precision digital maps, and sensor fusion.



Zhong Cao received the B.S.and Ph.D. degrees in automotive engineering from Tsinghua University in 2015 and 2020, respectively. He is currently an research scientist with University of Michigan City. His research interests include autonomous vehicle, trustworthy reinforcement learning, long life learning, and HD map.

Diange Yang received the B.S. and Ph.D. degrees in automotive engineering from Tsinghua University, Beijing, China, in 1996 and 2001, respectively. He serves as the Director of automotive engineering at Tsinghua University. He is currently a Professor with the Department of Automotive Engineering, Tsinghua University. His research interests include intelligent transport systems, vehicle electronics, and vehicle noise measurement. Dr. Yang attended in "Ten Thousand Talent Program" in 2016. He also received the Second Prize from the National Technology Invention Rewards of China in 2010 and the Award for Distinguished Young Science and Technology Talent of the China Automobile Industry in 2011.