ViLo: LiDAR Localization with Vision-Language Priors

Minghang Zhu[1,2,*] **Jianshi Wu**[1,2,*] **Yuxin Guo**[1,2,*] **Wen Li**[3]
Penghui Shang[4] **Sheng Ao**[1,2] **Cheng Wang**[1,2,†]

[1]Fujian Key Laboratory of Urban Intelligent Sensing and Computing, Xiamen University, Xiamen, China
[2]Key Laboratory of Multimedia Trusted Perception and Efficient Computing, Ministry of Education of China, Xiamen University, Xiamen, China
[3]School of Engineering Mathematics and Technology, University of Bristol, Bristol, United Kingdom
[4]Research and development, Zhiyuan Research Institute

mihoo@stu.xmu.edu.cn wujianshi@stu.xmu.edu.cn 23020240157665@stu.xmu.edu.cn

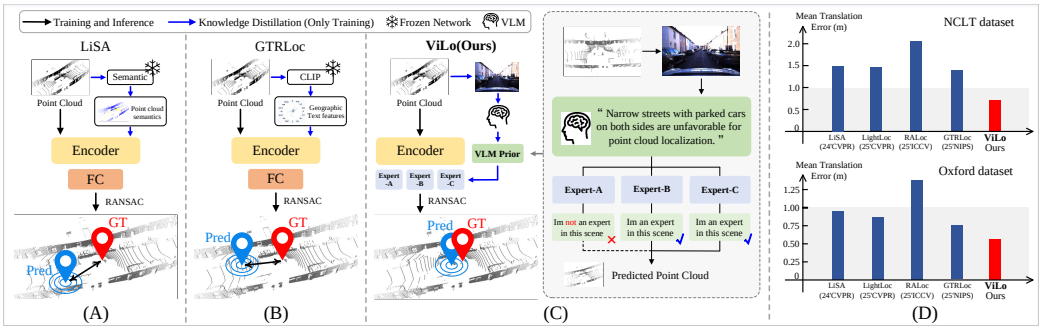


Figure 1: (A) LiSA and (B) GTRLoc rely on a single network to learn features indiscriminately, lacking scene-adaptive capability. (C) The proposed ViLo leverages VLM priors to guide multiple expert networks for more scene-adaptive learning. (D) Comparison of performance with current state-of-the-art methods on the Oxford and NCLT outdoor LiDAR localization benchmarks.

Abstract

LiDAR localization is a fundamental task in robotics and computer vision, aiming to estimate the global pose of point clouds. Although Scene Coordinate Regression (SCR) has demonstrated state-of-the-art performance in this field, standard SCR methods employ a monolithic network for uniform optimization across diverse scenes, which inevitably suffers from capacity interference in highly degraded scenes. Recent research on Vision-Language Models (VLMs) indicates that they can acquire rich scene understanding priors, providing the adaptive learning capabilities currently missing in localization networks. In this paper, we propose ViLo, the first framework to integrate VLM priors into SCR to fundamentally enhance localization robustness. Specifically, we design a temporally consistent key-frame querying mechanism to extract open-vocabulary degeneracy priors. We then construct a dynamic prior-guided Mixture-of-Experts (MoE) model for adaptive feature learning tailored to varying scenes. Finally, to avoid additional computational overhead during inference, ViLo internalizes the VLM's scene understanding into the 3D backbone via cross-modal distillation, enabling a lightweight, LiDAR-only deployment. Extensive experiments on the Oxford RobotCar and NCLT datasets demonstrate that ViLo significantly outperforms existing state-of-the-art methods. https://reckozmh.github.io/ViLo

[*]Equal contribution.
[†]Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

1 Introduction

LiDAR localization, which aims to estimate the 6-DoF global pose of point clouds, is a fundamental and critical task in robotics and autonomous driving [1, 2, 3, 4, 5]. Traditional map-based methods [6, 7] rely on dense matching between query frames and pre-built 3D maps, which often incurs prohibitive storage costs and high communication overhead [8, 9, 10]. To overcome these limitations, learning-based map-free regression methods have emerged [11, 12, 13, 14, 15]. These methods are generally categorized into Absolute Pose Regression (APR) and Scene Coordinate Regression (SCR). Unlike APR [16, 17], which directly regresses poses, SCR [18] predicts correspondences between the point cloud and world coordinates and subsequently solves for the pose via RANSAC registration [19, 20, 21]. Due to its explicit exploitation of geometric information, SCR has demonstrated superior performance and yielded numerous state-of-the-art results in large-scale outdoor environments [22]. For instance, LiSA [23] enhances performance by distilling semantic features into the SCR network to help the model understand deep point cloud semantics. GTR-Loc [24] leverages CLIP [25] to obtain geospatial text features, aiding the network in reducing the impact of visually similar geometric features. Additionally, RALoc [22] utilizes data augmentation and feature fusion to mitigate the influence of environmental structural changes.

Despite these impressive results, as shown in Fig. 1, standard SCR methods [14, 22, 23, 26, 24] face a fundamental bottleneck in large-scale outdoor scenarios: they typically employ a monolithic network for uniform optimization across significantly diverse environments [27, 28]. Real-world outdoor settings exhibit extreme structural diversity, ranging from "nominal environments" rich in structural features to "highly degraded scenes" characterized by severe occlusions, dynamic objects, or geometric homogeneity [29, 30, 31]. Forcing a monolithic network to perform uniform fitting across such divergent patterns inevitably leads to severe Capacity Interference, where the network struggles to find an effective balance between reliable geometric structures and environmental noise.

To fundamentally break through this limitation, we propose ViLo, the first SCR localization framework to deeply integrate high-level Vision-Language Model (VLM) priors with a Mixture-of-Experts (MoE) architecture [32, 33]. Specifically, we first introduce an efficient key-frame querying mechanism to obtain temporally consistent, open-vocabulary [34] VLM degeneracy priors. Subsequently, we construct a Scene-Adaptive MoE model. By combining hierarchical scale-aware experts with scene routing and adaptive gating mechanisms, ViLo dynamically schedules the expert network based on VLM scene priors [35, 36]. This approach effectively eliminates the capacity interference issues inherent in monolithic networks, enabling robust localization in complex and dynamic scenarios.

Finally, considering the prohibitive computational latency of invoking large VLMs or processing multi-modal images during real-time inference, ViLo adopts a Cross-modal Distillation strategy [37, 38, 39]. During training, the VLM's scene understanding and adaptive scheduling capabilities are internalized into a pure 3D LiDAR backbone. At inference time, the model entirely discards image inputs and the VLM, enabling a lightweight, real-time, zero-overhead, LiDAR-only deployment.

We evaluate ViLo on two large-scale outdoor benchmarks, including Oxford RobotCar [40] and NCLT [41]. Results demonstrate that ViLo significantly outperforms existing top-tier methods. Our primary contributions are summarized as follows:

- We propose ViLo, the first LiDAR localization framework to integrate VLM priors into SCR, fundamentally enhancing robustness and addressing scene-level capacity interference.
- We design an efficient key-frame querying mechanism to obtain stable, open-vocabulary degeneracy priors from the VLM, providing crucial guidance for adaptive optimization.
- We introduce a Scene-adaptive MOE that dynamically adapts to diverse environments via scene routing and adaptive gating, overcoming the limitations of monolithic uniform optimization.
- We establish a cross-modal distillation strategy to internalize VLM reasoning into the 3D backbone, enabling a real-time, LiDAR-only inference pipeline that surpasses state-of-the-art methods by 20% and 51% on Oxford and NCLT.

2 Related Work

LiDAR Localization. Traditional LiDAR localization methods typically aim to establish correspondences between query point clouds and pre-built 3D maps [3, 42, 43]. These methods primarily achieve localization through map matching. Although map matching can provide high accuracy, it suffers from prohibitive map storage costs and significant communication overhead. To address these limitations, researchers have proposed regression-based localization methods, which perform inference without relying on pre-built 3D maps.

Absolute Pose Regression (APR). Absolute Pose Regression methods [15, 17] directly regress 6-DoF poses from input point clouds. Pioneering research such as PointLoc [11] utilized PointNet++ [44] with a multi-layer perceptron (MLP) head to validate the feasibility of this approach, though it showed limitations in complex scenes. Subsequent studies have improved APR through novel network architectures, multi-modal fusion, or temporal constraints [16]. For example, DiffLoc [17] optimizes poses via an iterative diffusion process. Nevertheless, APR's reliance on global scene representations can hinder effective geometric encoding, which potentially limits localization accuracy.

Scene Coordinate Regression (SCR). Unlike APR, Scene Coordinate Regression methods learn to predict per-point world coordinates and determine the final pose through RANSAC [19]. SGLoc [14] first decoupled the localization task into point correspondence regression and pose estimation. To enhance training efficiency in large-scale scenes, LightLoc [26] constructs a pre-trained encoder that achieves optimal performance with just one hour of training. PELoc [18] utilizes data augmentation to achieve compatibility between low-cost rapid deployment and localization accuracy. Although SCR methods achieve higher accuracy by learning scene geometry, they perform poorly in highly degraded or visually similar scenes. To mitigate this issue, semantic-aware localization LiSA [23] distills a semantic segmentation model to help the network understand point cloud structures. The geometric-text-aware method GTRLoc [24] utilizes CLIP's pre-trained text features [25] to reduce feature ambiguity. RALoc [22] employs rotation augmentation and feature fusion strategies to reduce the impact of environmental structural noise. However, these methods primarily focus on feature-level enhancement and fundamentally rely on a monolithic network for uniform optimization across diverse scenes, which inevitably leads to capacity interference.

Vision-Language Models (VLM). Benefiting from the development of Large Language Models (LLMs) such as ChatGPT [45], LLaMA [46], and Gemini [47], the use of universal priors has become possible, stimulating research in the Vision-Language Model (VLM) field. Models like CLIP [25] and VL-Mamba [48] use large-scale pre-training to align textual and visual representations, facilitating the development of VLM-based localization tasks [49]. For instance, LP-SLAM [50] and TextSLAM [51] first achieved text-based place recognition, while Text2Pos [52] and Text2Loc [53] proposed using text for large-scale urban place recognition. The latest MSGNav [54] even enables training-free, zero-shot indoor navigation based on VLMs. However, the aforementioned methods use VLMs for feature matching or indoor navigation but lack the capability for adaptive architectural scheduling in diverse outdoor scenes. To this end, we propose ViLo, which leverages VLMs to obtain scene priors to dynamically guide adaptive localization for different experts across various scenes. Through a distillation strategy, VLM reasoning capabilities are internalized into the 3D backbone, enabling LiDAR-only localization without additional overhead during inference.

3 Methodology

ViLo relies on Scene Coordinate Regression (SCR) for LiDAR localization. The input is a query point cloud $\mathcal{P} = \{p_i\}_{i=1}^{N}$, where $p_i \in \mathbb{R}^3$ represents a 3D point in the local LiDAR coordinate frame. The output is a set of predicted world coordinates $\mathcal{P}' = \{p_i'\}_{i=1}^{N}$. The standard training loss for SCR is typically defined as L_1 distance between the predicted p_i' and the ground-truth p_i^*:

$$\mathcal{L}_{\text{loc}} = \frac{1}{N} \sum_{i=1}^{N} \|p_i' - p_i^*\|_1. \tag{1}$$

3.1 Motivation

The core idea of ViLo is to leverage VLM priors to assist the localization task. To illustrate our rationale, Fig. 2 presents: (A) the advantage of VLM reasoning over rigid semantics, and (B) the

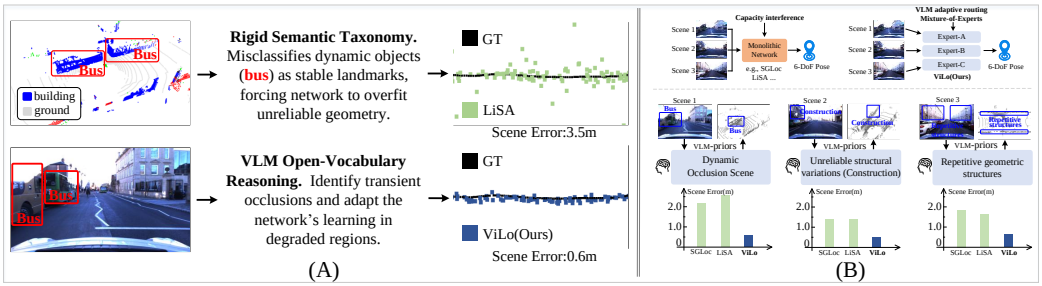


Figure 2: **Motivation.** Left(A): VLM priors are adopted to better perceive scene localizability compared to semantic-aware methods (LiSA). Right(B): Mixture-of-Experts (MOE) is utilized to resolve the capacity interference inherent in monolithic networks across various degraded scenes through VLM-guided dynamic expert scheduling.

necessity of the MoE architecture for diverse scenes. As shown in Fig. 2(A), compared to the semantic-aware method LiSA [23], which misidentifies dynamic buses as stable landmarks and forces the network to learn unreliable geometries, ViLo transfers 2D scene priors to the 3D backbone. This enables the network to perceive the localizability of the current environment and autonomously schedule feature learning for improved performance. Furthermore, Fig. 2(B) highlights the limitations of monolithic architectures. While existing methods like SGLoc [14] and LiSA [23] rely on feature-level fusion within a single network, they suffer from severe capacity interference during uniform optimization. In contrast, ViLo utilizes VLM priors to guide specialized learning across distinct expert networks, effectively resolving these conflicts. To this end, as shown in Fig. 3, we propose ViLo, a novel framework designed to efficiently integrate VLM priors into SCR-based localization.

3.2 VLM Priors

To ensure computational efficiency and stability, we first propose a temporally consistent key-frame querying mechanism. Subsequently, we leverage the VLM's Chain-of-Thought [55, 56] capability to explicitly reason about scene degeneracies from 2D images. Finally, we detail how these raw priors are extracted into optimization weights and scene semantic embeddings for downstream tasks.

Key-Frame Querying and Temporal Consistency Independent per-frame evaluation lacks temporal consistency and incurs prohibitive API costs. To balance evaluation stability and efficiency, we introduce a temporal consistency prompting mechanism by replacing exhaustive querying with spatial key-frames. A key-frame, denoted as $\mathcal{K}_i$, is selected when either of the following criteria is met: (1) the spatial displacement reaches the LiDAR sparse Field of View boundary, or (2) the visual similarity threshold between consecutive frames are satisfied (see **Appendix Sec.** A). When evaluating the current key-frame $\mathcal{K}_i$, we append the VLM results of the previous N key-frames ($\mathcal{K}_{i-1}$ to $\mathcal{K}_{i-N}$), which include the degradation score s and the degeneracy description $\mathcal{T}^{\text{desc}}$. This historical context enforces reasoning smoothness and suppresses evaluation fluctuations across continuous scenes. Intermediate non-key-frames efficiently inherit these priors via interpolation. This strategy reduces token overhead by $\sim 95\%$ while preserving robust scene-level guidance.

Open-Vocabulary Degeneracy Reasoning Traditional semantic methods rely on rigid classification taxonomies, failing to capture the complexity of real-world environments. We address this by guiding the VLM through a 2D-to-3D Chain-of-Thought to analyze outdoor structures from first principles. For a given key-frame $\mathcal{K}_i$, the VLM generates free-form, open-vocabulary outputs comprising three essential priors: (1) a localizability score $s_i \in [1, 10]$, where higher values indicate more severe degradation; (2) a degeneracy description $\mathcal{T}_i^{\text{desc}}$ for temporal prompting, such as "dynamic bus occlusion", "seasonal foliage variations", and "building construction"; and (3) a scene content analysis $\mathcal{T}_i^{\text{content}}$. Prompt details in **Appendix Sec.** B.

VLM Prior Extraction The VLM priors provide two key types of guidance for point cloud relocalization, as follows:

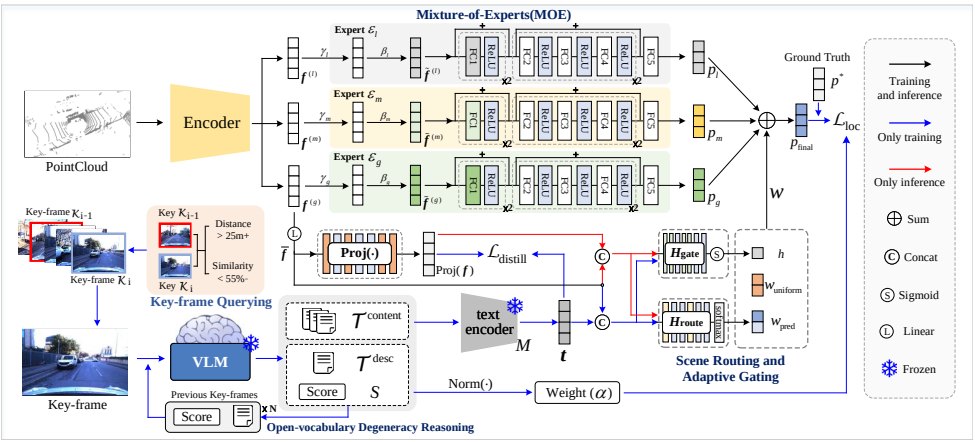


Figure 3: **The pipeline of ViLo.** We extract VLM priors via temporally consistent Key-frame Querying coupled with Open-vocabulary Degeneracy Reasoning. These priors are encoded by a frozen text encoder to extract deep semantic embeddings, which subsequently guide the Mixture-of-Experts via Scene Routing and Adaptive Gating to predict point clouds in the world coordinate system . Finally, the network is optimized by a localization loss ($\mathcal{L}_{\text{loc}}$) modulated by the prior information, alongside a cross-modal distillation loss ($\mathcal{L}_{\text{distill}}$) derived from the VLM's scene reasoning. Finally, LiDAR-only inference enables lightweight deployment.

1) Global Optimization Weight (α_i): The normalized evaluation score s_i serves as a global weight to modulate the overall training loss. It acts as a scene-adaptive curriculum learning strategy, preventing network gradients from being dominated by structural noise in highly degraded environments.

2) Scene Semantic Embedding ($\boldsymbol{t}_i$): We utilize a frozen, lightweight text encoder $M(\cdot)$ to extract a semantic embedding $\boldsymbol{t}_i = M(\mathcal{T}_i^{\text{content}})$ from the scene content analysis $\mathcal{T}_i^{\text{content}}$. This embedding provides the scene context necessary to guide the expert routing in MoE (Sec. 3.3)

3.3 Scene-Adaptive Mixture-of-Experts with Scene Routing

To overcome monolithic capacity interference, we introduce a Scene-Adaptive Mixture-of-Experts (MoE). It leverages VLM priors to dynamically route and gate scale-aware experts, tailoring optimization across varied scenes without conflicting geometric fitting.

Hierarchical Scale-Aware Tri-Experts To prevent expert homogenization, We deploy three parameter-independent experts ($\mathcal{E}_l, \mathcal{E}_m, \mathcal{E}_g$) within a residual architecture, each geometrically grounded at different depths of the 3D backbone [26]. Extracting features from distinct layers yields varying receptive fields to capture specific geometries: high-frequency local details $\boldsymbol{f}^{(l)}$, mid-range regional structures $\boldsymbol{f}^{(m)}$, and global contextual topologies $\boldsymbol{f}^{(g)}$. Furthermore, to explicitly amplify feature discrepancy and ensure specialization, we introduce learnable scaling ($\boldsymbol{\gamma}_k$) and shifting ($\boldsymbol{\beta}_k$) factors for each expert $k \in \{l, m, g\}$. The backbone feature $\boldsymbol{f}^{(k)}$ is expressed as:

$$\tilde{\boldsymbol{f}}^{(k)} = \boldsymbol{\gamma}_k \odot \boldsymbol{f}^{(k)} + \boldsymbol{\beta}_k, \tag{2}$$

where $\odot$ denotes element-wise multiplication. This mechanism enforces representation divergence, ensuring scale-specific specialization. Each expert then processes the modulated feature to output coordinate predictions $p_k = \mathcal{E}_k(\tilde{\boldsymbol{f}}^{(k)})$.

Scene Routing and Adaptive Gating Determining the optimal expert requires holistic awareness of both the environmental semantics and the current geometric layout. Thus, instead of relying solely on the textual prior, we concatenate the global point cloud feature $\bar{\boldsymbol{f}}_i = \text{Linear}\left(\boldsymbol{f}_i^{(g)}\right)$ with the VLM semantic embedding $\boldsymbol{t}_i$ to form a cross-modal context vector $\boldsymbol{v}_i = [\bar{\boldsymbol{f}}_i, \boldsymbol{t}_i]$. A routing network $H_{\text{route}}(\cdot)$ then projects this fused context into categorical probabilities across the three experts:

$$\boldsymbol{w}_{\text{pred}} = \text{Softmax}\big(H_{\text{route}}(\boldsymbol{v}_i)\big). \tag{3}$$

Although the cross-modal context provides robust guidance, VLM features can occasionally be noisy or misaligned with the geometric reality. To guarantee training stability, we introduce an Adaptive Gating Mechanism. We train a confidence gate $h_i = \sigma\big(H_{\text{gate}}(\boldsymbol{v}_i)\big) \in [0, 1]$, where σ denotes the Sigmoid function. The final routing weight vector $\boldsymbol{w}_i = [w_l, w_m, w_g]$ interpolates between the dynamically predicted routing and an unbiased uniform prior:

$$\boldsymbol{w}_i = h_i \cdot \boldsymbol{w}_{\text{pred}} + (1 - h_i) \cdot \boldsymbol{w}_{\text{uniform}}, \tag{4}$$

where $\boldsymbol{w}_{\text{uniform}}$ denotes a uniform weight. When the cross-modal context is highly confident ($h_i \to 1$), the network strictly follows the specialized routing. Conversely, when the scene semantics are ambiguous ($h_i \to 0$), the routing gracefully degrades to a uniform distribution, preventing expert collapse. The final coordinate prediction is aggregated as:

$$p_{\text{final}} = \sum_{k \in \{l,m,g\}} w_k \cdot p_k. \tag{5}$$

The aggregated p_{final} is then used as p' in Eq. 1 to compute the localization loss $\mathcal{L}_{\text{loc}}$.

Cross-Modal Scene Semantic Distillation To internalize the VLM's scene reasoning into the 3D backbone, we project the global LiDAR feature $\bar{\boldsymbol{f}}_i$ into the text-aligned latent space via a projection head $\text{Proj}(\cdot)$. A distillation loss $\mathcal{L}_{\text{distill}}$, based on cosine similarity $\cos(\cdot)$, aligns the implicit features $\text{Proj}(\bar{\boldsymbol{f}}_i)$ with VLM scene semantic embedding $\boldsymbol{t}_i$:

$$\mathcal{L}_{\text{distill}} = 1 - \cos(\text{Proj}(\bar{\boldsymbol{f}}_i), \boldsymbol{t}_i). \tag{6}$$

3.4 Loss Function and LiDAR-only Inference

Training Loss The total training objective is defined as:

$$\mathcal{L}_{\text{total}} = \alpha \mathcal{L}_{\text{loc}} + \lambda \mathcal{L}_{\text{distill}}, \tag{7}$$

where α denotes the global optimization weight derived from the VLM priors. It modulates the overall loss to prevent network gradients from being dominated by structural noise in highly degraded environments, and λ is a balancing parameter empirically set to 0.1.

LiDAR-Only Inference During inference, ViLo operates strictly on LiDAR inputs. Since the VLM priors are successfully distilled into the 3D backbone via $\mathcal{L}_{\text{distill}}$, the projected global LiDAR feature $\text{Proj}(\bar{\boldsymbol{f}}_i)$ serves as an effective surrogate for the scene semantic embedding $\boldsymbol{t}_i$. Thus, at test time, the context vector is constructed exclusively from the point cloud as $\hat{\boldsymbol{v}}_i = [\bar{\boldsymbol{f}}_i, \text{Proj}(\bar{\boldsymbol{f}}_i)]$, from which both the scene routing probabilities and the adaptive gate are directly predicted:

$$\boldsymbol{w}_{\text{pred}} = \text{Softmax}\big(H_{\text{route}}(\hat{\boldsymbol{v}}_i)\big), \tag{8}$$

$$h_{\text{pred}} = \sigma\big(H_{\text{gate}}(\hat{\boldsymbol{v}}_i)\big). \tag{9}$$

The final routing weight $\boldsymbol{w}$ is then computed by interpolating $\boldsymbol{w}_{\text{pred}}$ with the uniform prior $\boldsymbol{w}_{\text{uniform}}$ using h_{pred}, exactly as formulated in Eq. 4.

4 Experiments

4.1 Setup

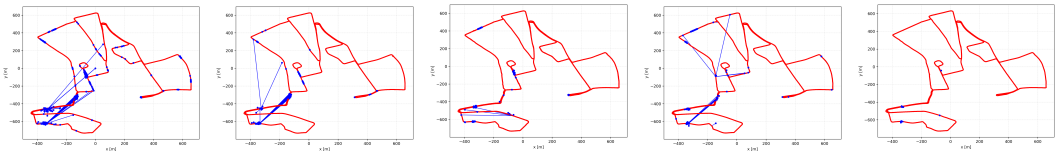
Datasets and Evaluation Metrics We evaluate on two outdoor LiDAR localization benchmarks: Quality-enhanced Oxford RobotCar [40] and NCLT [41], using mean position error (m) and mean orientation error (°) as metrics. The datasets are detailed as follows:

*Quality-enhanced **Oxford** RobotCar*: This urban driving dataset covers a 10 km route under various weather conditions, processed using the ground truth optimization method from SGLoc [14]. We use sequences 11-14-02-26, 14-12-05-52, 14-14-48-55, and 18-15-20-12 for training, and sequences 15-13-06-37, 17-13-26-39, 17-14-03-00, and 18-14-14-42 for testing.

NCLT: Collected at the University of Michigan North Campus, this dataset spans a 5.5 km route and captures seasonal variations. Ground truth is obtained by fusing GPS with SLAM optimization. Following prior localization settings [14, 23, 26], we use 2012-01-22, 2012-02-02, 2012-02-18, and 2012-05-11 for training, and 2012-02-12, 2012-02-19, 2012-03-31, and 2012-05-26 for testing.

Table 1: Mean position and orientation error ↓ (m, °) on the Quality-enhanced Oxford dataset. We highlight the **best** and second-best results.

	Method	15-13-06-37	17-13-26-39	17-14-03-00	18-14-14-42	Mean[m, °] ↓
APR	PosePN++ [12] (2022'PR)	4.54, 1.83	6.44, 1.78	4.89, 1.55	4.64, 1.61	5.13, 1.69
	STCLoc [16] (2022'TITS)	5.14, 1.27	6.12, 1.21	5.32, 1.08	4.76, 1.19	5.34, 1.19
	HypLiLoc [15] (2023'CVPR)	5.03, 1.46	4.31, 1.43	3.61, 1.11	2.61, 1.09	3.89, 1.27
SCR	SGLoc [14] (2023'CVPR)	1.79, 1.67	1.81, 1.76	1.33, 1.59	1.19, 1.39	1.53, 1.60
	LiSA [23] (2024'CVPR)	0.94, 1.10	1.17, 1.21	0.84, 1.15	0.85, 1.11	0.95, 1.14
	LightLoc [26] (2025'CVPR)	0.82, 1.12	0.85, 1.07	0.81, 1.11	0.82, 1.16	0.83, 1.12
	RALoc [22] (2025'ICCV)	1.52, 1.28	1.71, 1.27	1.37, 1.20	1.41, 1.23	1.51, 1.24
	GTRLoc [24](2025'NeurIPS)	0.77, 1.02	0.77, 1.01	0.67, 1.01	0.80, 1.07	0.75, 1.03
	ViLo(Ours)	**0.65**, **0.87**	**0.60**, **0.82**	**0.53**, **0.83**	**0.61**, **0.92**	**0.60**, **0.86**



SGLoc LiSA LightLoc RALoc **Ours(ViLo)**

Figure 4: Visualization on the Quality-enhanced Oxford (Seq. 17-13-26-39). Red and black points denote predictions and ground truth, respectively, localization errors >5 m are shown in blue points.

Baselines We compare with state-of-the-art large-scale LiDAR localization methods, including absolute pose regression (APR) methods PosePN++ [12], STCLoc [16], and HypLiLoc [15], as well as scene coordinate regression (SCR) methods SGLoc [14], LiSA [23], LightLoc [26], RALoc [22], and GTRLoc [24]. Among them, LightLoc, RALoc, and GTRLoc are the most recent methods.

Implementation ViLo is implemented in PyTorch with MinkowskiEngine for sparse convolutions. Voxel sizes are set to 0.25 m on Oxford and 0.3 m on NCLT. We use the AdamW optimizer [57] with a One-cycle learning rate schedule from 5e-4 to 5e-3, a batch size of 256, and train for 150 epochs. The VLM calls APIs of Gemini 2.5 pro [47], while a frozen Qwen3-Embedding-0.6B [58] extracts 256-dimensional VLM scene text features. All experiments are conducted on a single NVIDIA RTX 4090 GPU. More training details and network architecture ($H_{\text{route}}(\cdot)$, $H_{\text{gate}}(\cdot)$, $\text{Proj}(\cdot)$) implementation can be found in **Appendix Sec.** C.

4.2 Results on Oxford Dataset

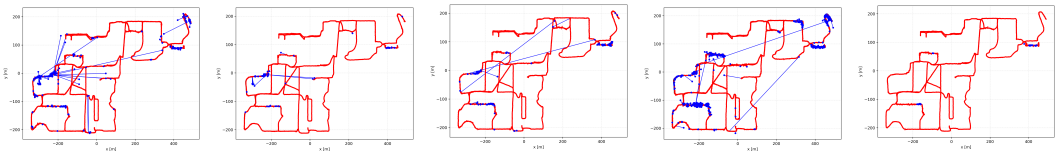
Tab. 1 summarizes the localization performance comparison on the Quality-enhanced Oxford benchmark. ViLo achieves consistent state-of-the-art performance across all individual test sequences, reducing the average translation and orientation errors by 20% (from 0.75 m to 0.60 m) and 17% (from $1.03°$ to $0.86°$) over the second-best method GTRLoc [24]. Notably, on the challenging sequence 17-13-26-39 characterized by severe dynamic occlusions, where semantic-aware LiSA degrades to 1.17 m, ViLo maintains robust accuracy at 0.60 m, and further achieves an optimal 0.53 m on sequence 17-14-03-00. Unlike monolithic SCR baselines that suffer from capacity interference or overfit unreliable transient geometry such as dynamic vehicles, ViLo leverages open-vocabulary VLM priors to dynamically schedule scale-aware experts via adaptive gating. Consequently, as visualized in Fig. 4, ViLo effectively eliminates catastrophic localization outliers (blue points, > 5 m) around sharp turns and occluded corridors, achieving superior trajectory smoothness and geometric fidelity.

4.3 Results on NCLT Dataset

Tab. 2 and Fig. 5 report the localization performance on the long-term NCLT dataset, which exhibits substantial seasonal variations. ViLo achieves an average translation error of 0.69 m and an orientation error of $2.28°$, outperforming the top baseline GTRLoc [24] by an unprecedented 51% margin (0.69 m versus 1.40 m). The superiority of ViLo is particularly pronounced on the cross-season sequence 2012-05-26, where all existing SCR baselines severely collapse with errors exceeding 3.0 m due to drastic foliage and structural changes, whereas ViLo maintains sub-meter precision at 0.99 m. As illustrated in the qualitative trajectory of sequence 2012-02-19 (Fig. 5), baseline models frequently produce false global matches across distant campus blocks, resulting in catastrophic cross-loop

Table 2: Mean position and orientation error ↓ (m, °) on the NCLT dataset. We highlight the **best** and second-best results.

	Method	2012-02-12	2012-02-19	2012-03-31	2012-05-26	Mean[m, °] ↓
APR	PosePN++ [12] (2022'PR)	4.97, 3.75	3.68, 2.65	4.35, 3.38	8.42, 4.30	5.36, 3.52
	STCLoc [16] (2022'TITS)	4.91, 4.34	3.25, 3.10	3.75, 4.04	7.53, 4.95	4.86, 4.11
	HypLiLoc [15] (2023'CVPR)	1.71, 3.56	1.68, 2.69	1.52, 2.90	2.29, 3.34	1.80, 3.12
SCR	SGLoc [14] (2023'CVPR)	1.20, 3.08	1.20, 3.05	1.12, 3.28	3.48, 4.43	1.75, 3.46
	LiSA [23] (2024'CVPR)	0.97, **2.23**	0.91, **2.09**	0.87, **2.21**	3.11, 2.72	1.47, 2.31
	LightLoc [26] (2025'CVPR)	0.98, 2.76	0.89, 2.51	0.86, 2.67	3.10, 3.26	1.46, 2.80
	RALoc [22] (2025'ICCV)	1.61, 4.71	1.61, 4.88	1.58, 4.71	3.51, 5.40	2.07, 4.92
	GTRLoc [24] (2025'NeurIPS)	0.95, 2.53	0.82, 2.45	0.82, 2.52	3.01, 2.99	1.40, 2.62
	ViLo(Ours)	**0.64**, 2.39	**0.55**, 2.15	**0.59**, 2.22	**0.99**, **2.37**	**0.69**, **2.28**



SGLoc LiSA LightLoc RALoc **Ours(ViLo)**

Figure 5: Visualization on the NCLT dataset (Seq. 2012-02-19). Red and black points denote predictions and ground truth, respectively, localization errors >5 m are shown in blue points.

Table 4: Ablation of Different VLM Priors on Oxford (w/o:without, w.:with).

Method	Mean Error ↓	
	Position(m)	Orientation(°)
ViLo (w/o VLM)	0.91	1.14
ViLo (w. Qwen2.5-7b)	0.93	1.17
ViLo (w. GPT-4o mini)	0.88	1.05
ViLo (w. Qwen 3.5 plus)	0.66	0.88
ViLo (w. Gemini 2.5 pro)	**0.60**	**0.86**

Table 5: Ablation on Scene Semantic Encoders (w/o:without, w.:with).

Method	Model Params	Position Error ↓	
		Oxford	NCLT
ViLo (w. Nomic)	0.15B	0.85	1.05
ViLo (w. BGE-M3)	0.5B	0.73	0.72
ViLo (w. Emb-gemma)	0.3B	0.66	0.74
ViLo (w. CLIP)	**0.12B**	0.77	0.85
ViLo (w. Qwen3-Emb)	0.6B	**0.60**	**0.69**

line jumps and clustered failures (blue points, > 5 m). In contrast, by distilling open-vocabulary degradation priors to dynamically activate scale-aware experts, ViLo effectively handles geometric ambiguity and environmental shifts, preserving uninterrupted trajectory continuity and eliminating severe relocalization errors.

4.4 Runtime and Model Size

Inference time is critical for localization. With LiDAR running at 20 Hz on Oxford and 10 Hz on NCLT, real-time inference requires under 50 ms and 100 ms per frame, respectively. As shown in Tab. 3, Although VLM priors are used in training via distillation, ViLo needs only point cloud input to the SCR module at inference. It runs at 33 ms (<50 ms) on Oxford and 53 ms (<100 ms) on NCLT, meeting real-time demands for autonomous driving.

Table 3: Runtime and Model Size Comparison on Oxford and NCLT.

Method	Model Size↓	Runtime↓	
		Oxford(<50ms)	NCLT(<100ms)
SGLoc [14]	116MB	38ms	75ms
LiSA [23]	116MB	38ms	75ms
LightLoc [26]	74MB	29ms	48ms
RALoc [22]	123MB	26ms	55ms
ViLo(Ours)	92MB	33ms	53ms

4.5 Ablation studies

4.5.1 Ablation on VLM Priors

Tab. 4 investigates the impact of employing different VLM priors on the Oxford benchmark. Advanced foundation models equipped with strong spatial reasoning capabilities, specifically Gemini 2.5 Pro (0.60 m, 0.86°) and Qwen 3.5 Plus (0.66 m, 0.88°), significantly surpass the variant without VLM (0.91 m, 1.14°), yielding an accuracy improvement of up to 34.1%. In contrast, lightweight models such as GPT-4o mini (0.88 m) and Qwen2.5-7b (0.93 m) fail to provide reliable 2D-to-3D degeneracy reasoning. Notably, Qwen2.5-7b causes slight performance degradation compared to the model

Table 6: Ablation of ViLO's main components on Oxford and NCLT.

VLM-w	VLM-t	MOE	Mean Error [m, °] ↓	
			Oxford	NCLT
			0.93,1.15	1.51,2.85
✓			0.85,1.13	1.34,2.66
		✓	0.80,1.02	1.27,2.60
	✓	✓	0.67,0.88	0.87,2.47
✓	✓	✓	0.60,0.86	0.69,2.28

Table 7: Ablation on the Number of Experts.

Number of Experts	Mean Error [m, °]↓	
	Oxford	NCLT
ViLo-(None)	0.93, 1.15	1.51, 2.85
ViLo-(2 Expert)	0.68, 0.90	0.98, 2.64
ViLo-(3 Expert)	0.60, 0.86	0.69, 2.28
ViLo-(4 Expert)	0.60, 0.83	0.67, 2.18

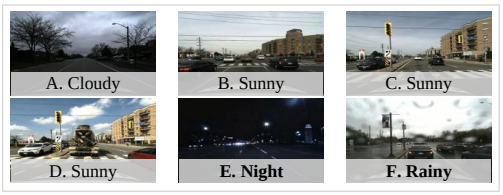


Figure 6: Boreas Dataset Scenes. We train on (A), (B), and (C), inference on (D), (E) and (F).

Table 8: ViLo LiDAR-only inference on Boreas Dataset.

Test Seq.	Different Conditions		
	D(Sun)	E(**Night**)	F(**Rain**)
Mean Error[m,°] ↓	2.33,2.68	2.63,2.48	2.71,2.46

without VLM priors, indicating that superficial visual descriptions or noisy degradation scores inject misleading supervisory signals during distillation and corrupt the expert routing logic. These comparisons demonstrate that high-capacity multimodal reasoning is indispensable for extracting accurate scene priors to guide dynamic MoE scheduling.

4.5.2 Ablation on Scene Semantic Encoders

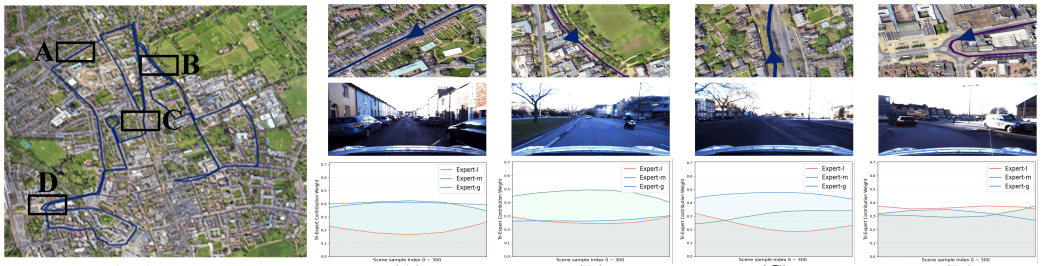
Tab. 5 evaluates various frozen text encoders for embedding the VLM scene analysis T_{content}, including Nomic [59], CLIP [25], Emb-gemma [60], BGE-M3 [61], and Qwen3-Emb [58]. Overall localization accuracy scales positively with the representational capacity of the semantic encoder, with our default Qwen3-Emb (0.6B) achieving the lowest errors of 0.60 m on Oxford and 0.69 m on NCLT. Among compact models, the lightweight CLIP (0.12B) outperforms the larger Nomic (0.15B) across both datasets, which is attributable to its vision-language pretraining that aligns textual descriptors more naturally with spatial scene semantics. In addition, Emb-gemma (0.3B) offers an appealing trade-off between capacity and accuracy, outperforming the 0.5B BGE-M3 on Oxford (0.66 m versus 0.73 m) and demonstrating competitive robustness on NCLT. These results verify that expressive textual embeddings are critical for providing discriminative scene context to guide the MoE routing mechanism.

4.5.3 Ablation on ViLo Main Components

Tab. 6 dissects the individual and cumulative contributions of ViLo's core designs: the Global Optimization Weight α (VLM-w), Scene Semantic Embedding $\boldsymbol{t}$ (VLM-t), and the Mixture-of-Experts (MOE) architecture. Introducing VLM-w alone reduces translation error from 0.93 m to 0.85 m on Oxford and from 1.51 m to 1.34 m on NCLT, confirming that curriculum weighting effectively mitigates the gradient corruption caused by degraded structures. Deploying an unguided MOE alone yields limited improvements (0.80 m and 1.27 m), as pure geometric features lack high-level contextual awareness to properly schedule experts. Crucially, conditioning MOE routing on VLM-t unleashes a remarkable performance boost, slashing the NCLT position error by 31.5% (1.27 m to 0.87 m) and the Oxford error to 0.67 m, which demonstrates that open-vocabulary semantic context is vital for steering scale-aware experts toward their specialized domains. Finally, combining all components achieves the optimal accuracy (0.60 m on Oxford and 0.69 m on NCLT), exhibiting strong synergy between loss-level noise suppression and architectural scene routing.

4.5.4 Ablation on the Number of Experts

To determine the optimal model capacity for our Scene-Adaptive MoE, we investigate the impact of varying the expert count E in Tab. 7. The monolithic baseline ($E =$ None) exhibits the highest errors across both benchmarks (0.93 m and 1.51 m) due to severe capacity interference under uniform optimization. Introducing multi-expert specialization markedly enhances localization performance, with $E = 2$ substantially reducing errors to 0.68 m on Oxford and 0.98 m on NCLT. Setting $E = 3$ further boosts accuracy to 0.60 m and 0.69 m, which naturally aligns with our tri-scale geometric design across local, regional, and global features. While expanding to $E = 4$ yields marginal



Oxford Dataset (A) (B) (C) (D)

Figure 7: Visualization of MoE expert contribution weights across representative regions, demonstrating adaptive routing conditioned on VLM priors.

improvements on NCLT (0.67 m) and maintains identical translation on Oxford (0.60 m), it incurs noticeable parameter redundancy and computational overhead without offering substantial performance leaps. Therefore, $E = 3$ is adopted as the default configuration, achieving the ideal balance between representation capacity and architectural efficiency.

4.5.5 Ablation on Nighttime and Rainy Conditions

Because ViLo distills 2D VLM image priors into the 3D network during training, it relies solely on point clouds during inference. To verify its effectiveness under extreme conditions absent in Oxford and NCLT, we conduct experiments on the Boreas dataset [62]. As illustrated in Fig. 6, ViLo is trained on Boreas dataset sequences (A, B and C) and tested across varying conditions (D, E and F), including nighttime (E) and rainy (F) environments. As shown in Tab. 8, ViLo achieves comparable localization performance in both Night (2.63m, 2.48°) and Rainy (2.71m, 2.46°) conditions to that in Sunny (2.33m, 2.68°). This demonstrates the ViLo's robust environmental adaptability and validates the feasibility of leveraging VLM priors via MOE and distillation. Details in **Appendix Sec.** D.

4.5.6 Visualization of Mixture-of-Experts(MOE) Contributions

Fig. 7 visualizes the MoE contributions across four representative Oxford regions (A-D), demonstrating adaptive routing conditioned on VLM priors. In narrow streets with severe occlusions (A), the model relies on mid- and global-scale experts (Expert-m, g) for contextual constraints. Along park edges with unstructured vegetation (B), Expert-m dominates to handle uneven spatial density. Conversely, on open main roads lacking local features (C), the system shifts to Expert-g, leveraging macroscopic topology for stability. In highly complex intersections (D), expert weights reach a dynamic equilibrium, with Expert-l's contribution increasing to capture fine near-field details. Ultimately, ViLO dynamically allocates routing weights guided by these priors, effectively mitigating the performance degradation caused by capacity interference in a single network across diverse scenes. We also discuss feature distillation in **Appendix Sec.** E.

5 Conclusion

We present ViLo, the first LiDAR localization framework to integrate Vision-Language Model (VLM) priors into Scene Coordinate Regression. Guided by these priors, our Scene-Adaptive Mixture-of-Experts (MoE) effectively overcomes capacity interference in degraded scenes. Through cross-modal distillation, ViLo achieves lightweight, LiDAR-only inference without runtime overhead, establishing a new state-of-the-art on the Oxford and NCLT benchmarks.

6 Acknowledgements

This work was partially supported by the National Natural Science Foundation of China (No. 62501502), and the Fundamental Research Funds for the Central Universities (20720260165). The authors declare no competing interests.

References

[1] X. Chen, T. Läbe, A. Milioto, T. Röhling, J. Behley, and C. Stachniss, "Overlapnet: A siamese network for computing lidar scan similarity with applications to loop closing and localization," *Autonomous Robots*, vol. 46, no. 1, pp. 61–81, 2022.

[2] Z. Shi, X. Zhang, W. Xu, Y. Xia, Y. Zang, S. Shen, and C. Wang, "L2rsi: Cross-view lidar-based place recognition for large-scale urban scenes via remote sensing imagery," *Advances in Neural Information Processing Systems*, vol. 38, pp. 160 681–160 704, 2026.

[3] C. Shi, X. Chen, J. Xiao, B. Dai, and H. Lu, "Fast and accurate deep loop closing and relocalization for reliable lidar slam," *IEEE Transactions on Robotics*, vol. 40, pp. 2620–2640, 2024.

[4] H. Yin, X. Xu, S. Lu, X. Chen, R. Xiong, S. Shen, C. Stachniss, and Y. Wang, "A survey on global lidar localization: Challenges, advances and open problems," *International Journal of Computer Vision*, vol. 132, no. 8, pp. 3139–3171, 2024.

[5] T. Zhang, G. Wang, Y. Chen, H. Zhang, and J. Hu, "Multi-constellation-inspired single-shot global lidar localization," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 38, no. 9, 2024, pp. 10 404–10 412.

[6] J. Komorowski, "Minkloc3d: Point cloud based large-scale place recognition," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2021, pp. 1790–1799.

[7] D. Cattaneo, M. Vaghi, and A. Valada, "Lcdnet: Deep loop closure detection and point cloud registration for lidar slam," *IEEE Transactions on Robotics*, vol. 38, no. 4, pp. 2074–2093, 2022.

[8] K. Xiong, H. Xiang, Q. Xu, C. Wen, S. Shen, J. Li, and C. Wang, "Mining and transferring feature-geometry coherence for unsupervised point cloud registration," *Advances in Neural Information Processing Systems*, vol. 37, pp. 35 468–35 491, 2024.

[9] K. Du, H. Xu, H. Li, H. Qu, C.-W. Fu, and S. Liu, "Hybridreg: Robust 3d point cloud registration with hybrid motions," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 3, 2025, pp. 2789–2797.

[10] Q. Zhou, S. Agostinho, A. Ošep, and L. Leal-Taixé, "Is geometry enough for matching in visual localization?" in *European Conference on Computer Vision*. Springer, 2022, pp. 407–425.

[11] W. Wang, B. Wang, P. Zhao, C. Chen, R. Clark, B. Yang, A. Markham, and N. Trigoni, "Pointloc: Deep pose regressor for lidar point cloud localization," *IEEE Sensors Journal*, vol. 22, no. 1, pp. 959–968, 2021.

[12] S. Yu, C. Wang, C. Wen, M. Cheng, M. Liu, Z. Zhang, and X. Li, "Lidar-based localization using universal encoding and memory-aware regression," *Pattern Recognition*, vol. 128, p. 108685, 2022.

[13] J. Wu, M. Zhu, D. Liu, W. Li, S. Ao, S. Shen, C. Wen, and C. Wang, "Leader: Learning reliable local-to-global correspondences for lidar relocalization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2026, pp. 9932–9942.

[14] W. Li, S. Yu, C. Wang, G. Hu, S. Shen, and C. Wen, "Sgloc: Scene geometry encoding for outdoor lidar localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 9286–9295.

[15] S. Wang, Q. Kang, R. She, W. Wang, K. Zhao, Y. Song, and W. P. Tay, "Hypliloc: Towards effective lidar pose regression with hyperbolic fusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 5176–5185.

[16] S. Yu, C. Wang, Y. Lin, C. Wen, M. Cheng, and G. Hu, "Stcloc: Deep lidar localization with spatio-temporal constraints," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 1, pp. 489–500, 2022.

[17] W. Li, Y. Yang, S. Yu, G. Hu, C. Wen, M. Cheng, and C. Wang, "Diffloc: Diffusion model for outdoor lidar localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 045–15 054.

[18] Y. Chen, Q. Li, Y. Yang, W. Li, S. Ao, and C. Wang, "Unleashing the power of data generation in one-pass outdoor lidar localization," in *Proceedings of the 33rd ACM International Conference on Multimedia*, 2025, pp. 199–208.

[19] O. Chum and J. Matas, "Optimal randomized ransac," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1472–1482, 2008.

[20] S. Ao, Q. Hu, H. Wang, K. Xu, and Y. Guo, "Buffer: Balancing accuracy, efficiency, and generalizability in point cloud registration," in *CVPR*, 2023, pp. 1255–1264.

[21] S. Ao, Y. Guo, Q. Hu, B. Yang, A. Markham, and Z. Chen, "You only train once: Learning general and distinctive 3d local descriptors," *TPAMI*, vol. 45, no. 3, pp. 3949–3967, 2022.

[22] Y. Yang, W. Li, S. Ao, Q. Xu, S. Yu, Y. Guo, Y. Zhou, S. Shen, and C. Wang, "Raloc: Enhancing outdoor lidar localization via rotation awareness," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025, pp. 3304–3313.

[23] B. Yang, Z. Li, W. Li, Z. Cai, C. Wen, Y. Zang, M. Muller, and C. Wang, "Lisa: Lidar localization with semantic awareness," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 15 271–15 280.

[24] S. Yu, W. Li, X. Sun, Z. Yuan, X. Wang, S. Wang, R. She, and C. Wang, "Gtr-loc: Geospatial text regularization assisted outdoor lidar localization," *Advances in Neural Information Processing Systems*, vol. 38, pp. 136 468–136 490, 2026.

[25] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.

[26] W. Li, C. Liu, S. Yu, D. Liu, Y. Zhou, S. Shen, C. Wen, and C. Wang, "Lightloc: Learning outdoor lidar localization at light speed," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 6680–6689.

[27] W. Li, D. Wang, Z. Ding, A. Sohrabizadeh, Z. Qin, J. Cong, and Y. Sun, "Hierarchical mixture of experts: Generalizable learning for high-level synthesis," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 39, no. 17, 2025, pp. 18 476–18 484.

[28] P. Wang, Z. Fan, T. Chen, and Z. Wang, "Neural implicit dictionary learning via mixture-of-expert training," in *International Conference on Machine Learning*. PMLR, 2022, pp. 22 613–22 624.

[29] Y. Huang, C. Liu, M. Zhu, S. Ao, C. Wen, and C. Wang, "Difflo: Semantic-aware lidar odometry with diffusion-based refinement," in *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2025, pp. 17 050–17 059.

[30] C. Liu, W. Li, Y. Huang, M. Zhu, Y. Yang, D. Liu, S. Ao, and C. Wang, "Rcp-lo: A relative coordinate prediction framework for generalizable deep lidar odometry," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 9, 2026, pp. 7078–7086.

[31] B. Zhou, Z. Zhang, Z. Song, J. Guo, T. Zhang, S. Song, Q. Wu, and H. Kong, "Generalizing unsupervised lidar motion estimation via patch-aware feature reweighting under snowy conditions," *Pattern Recognition*, p. 114630, 2026.

[32] Y. Zhang, R. Cai, T. Chen, G. Zhang, H. Zhang, P.-Y. Chen, S. Chang, Z. Wang, and S. Liu, "Robust mixture-of-expert training for convolutional neural networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 90–101.

[33] H. Xi, Z. Shao, D. A. Hensher, J. D. Nelson, H. Chen, and K. Wijayaratna, "A multi-task transformer with mixture-of-experts for personalized periodic predictions of individual travel behavior in multimodal public transport," *Transportation Research Part C: Emerging Technologies*, vol. 179, p. 105287, 2025.

[34] J. Wu, X. Li, S. Xu, H. Yuan, H. Ding, Y. Yang, X. Li, J. Zhang, Y. Tong, X. Jiang *et al.*, "Towards open vocabulary learning: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 46, no. 7, pp. 5092–5113, 2024.

[35] X. Liu, H. Zhang, Y.-m. Cheung, X. You, and Y. Y. Tang, "Efficient single image dehazing and denoising," *Computer Vision and Image Understanding*, vol. 162, no. C, p. 23–33, 2017.

[36] H. Xi, Y. Zhang, and Y. Zhang, "Detection of safety features of drivers based on image processing," in *18th COTA International Conference of Transportation Professionals*. American Society of Civil Engineers Reston, VA, 2018, pp. 2098–2109.

[37] S. Gupta, J. Hoffman, and J. Malik, "Cross modal distillation for supervision transfer," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2827–2836.

[38] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International journal of computer vision*, vol. 129, no. 6, pp. 1789–1819, 2021.

[39] X. Liu, Z. Hu, H. Ling, and Y.-M. Cheung, "Mtfh: A matrix tri-factorization hashing framework for efficient cross-modal retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 964–981, 2021.

[40] D. Barnes, M. Gadd, P. Murcutt, P. Newman, and I. Posner, "The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset," in *2020 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2020, pp. 6433–6438.

[41] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of michigan north campus long-term vision and lidar dataset," *The International Journal of Robotics Research*, vol. 35, no. 9, pp. 1023–1035, 2016.

[42] L. Luo, S. Zheng, Y. Li, Y. Fan, B. Yu, S.-Y. Cao, J. Li, and H.-L. Shen, "Bevplace: Learning lidar-based place recognition using bird's eye view images," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 8700–8709.

[43] L. Luo, S.-Y. Cao, X. Li, J. Xu, R. Ai, Z. Yu, and X. Chen, "Bevplace++: Fast, robust, and lightweight lidar global localization for unmanned ground vehicles," *IEEE Transactions on Robotics*, 2025.

[44] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[45] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[46] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models. arxiv 2023," *arXiv preprint arXiv:2302.13971*, vol. 10, 2023.

[47] G. Comanici, E. Bieber, M. Schaekermann, I. Pasupat, N. Sachdeva, I. Dhillon, M. Blistein, O. Ram, D. Zhang, E. Rosen *et al.*, "Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities," *arXiv preprint arXiv:2507.06261*, 2025.

[48] Y. Qiao, Z. Yu, L. Guo, S. Chen, Z. Zhao, M. Sun, Q. Wu, and J. Liu, "Vl-mamba: Exploring state space models for multimodal learning," *arXiv preprint arXiv:2403.13600*, 2024.

[49] S. Kang, Y. Liao, P. Wang, W. Liao, Q. Zhang, B. Busam, X. Chen, and Y. Liu, "Vlm-loc: Localization in point cloud maps via vision-language models," *arXiv preprint arXiv:2603.09826*, 2026.

[50] W. Zhang, Y. Guo, L. Niu, P. Li, C. Zhang, Z. Wan, J. Yan, F. U. D. Farrukh, and D. Zhang, "Lp-slam: Language-perceptive rgb-d slam system based on large language model," *arXiv preprint arXiv:2303.10089*, 2023.

[51] B. Li, D. Zou, Y. Huang, X. Niu, L. Pei, and W. Yu, "Textslam: Visual slam with semantic planar text features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 1, pp. 593–610, 2023.

[52] M. Kolmet, Q. Zhou, A. Ošep, and L. Leal-Taixé, "Text2pos: Text-to-point-cloud cross-modal localization," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 6687–6696.

[53] Y. Xia, L. Shi, Z. Ding, J. F. Henriques, and D. Cremers, "Text2loc: 3d point cloud localization from natural language," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 14 958–14 967.

[54] X. Huang, S. Zhao, Y. Wang, X. Lu, W. Zhang, R. Qu, W. Li, Y. Wang, and C. Wen, "Msgnav: Unleashing the power of multi-modal 3d scene graph for zero-shot embodied navigation," *arXiv preprint arXiv:2511.10376*, 2025.

[55] Q. Zhao, Y. Lu, M. J. Kim, Z. Fu, Z. Zhang, Y. Wu, Z. Li, Q. Ma, S. Han, C. Finn *et al.*, "Cot-vla: Visual chain-of-thought reasoning for vision-language-action models," in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 1702–1713.

[56] X. Zhao, C. Liu, R. Ji, Z. Zhang, M. Zhu, L. Song, Z. Ren, L. Qingliang, Y. Gao, Z. Du *et al.*, "Cot-vlnbench: A benchmark for visual chain-of-thought reasoning in vision-language-navigation robots," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, no. 43, 2026, pp. 36 573–36 581.

[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[58] Y. Zhang, M. Li, D. Long, X. Zhang, H. Lin, B. Yang, P. Xie, A. Yang, D. Liu, J. Lin *et al.*, "Qwen3 embedding: Advancing text embedding and reranking through foundation models," *arXiv preprint arXiv:2506.05176*, 2025.

[59] Z. Nussbaum, J. X. Morris, B. Duderstadt, and A. Mulyar, "Nomic embed: Training a reproducible long context text embedder," *arXiv preprint arXiv:2402.01613*, 2024.

[60] H. S. Vera, S. Dua, B. Zhang, D. Salz, R. Mullins, S. R. Panyam, S. Smoot, I. Naim, J. Zou, F. Chen *et al.*, "Embeddinggemma: Powerful and lightweight text representations," *arXiv preprint arXiv:2509.20354*, 2025.

[61] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu, "Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation," *arXiv preprint arXiv:2402.03216*, vol. 4, no. 5, 2024.

[62] K. Burnett, D. J. Yoon, Y. Wu, A. Z. Li, H. Zhang, S. Lu, J. Qian, W.-K. Tseng, A. Lambert, K. Y. Leung *et al.*, "Boreas: A multi-season autonomous driving dataset," *The International Journal of Robotics Research*, vol. 42, no. 1-2, pp. 33–42, 2023.

[63] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv preprint arXiv:2304.07193*, 2023.

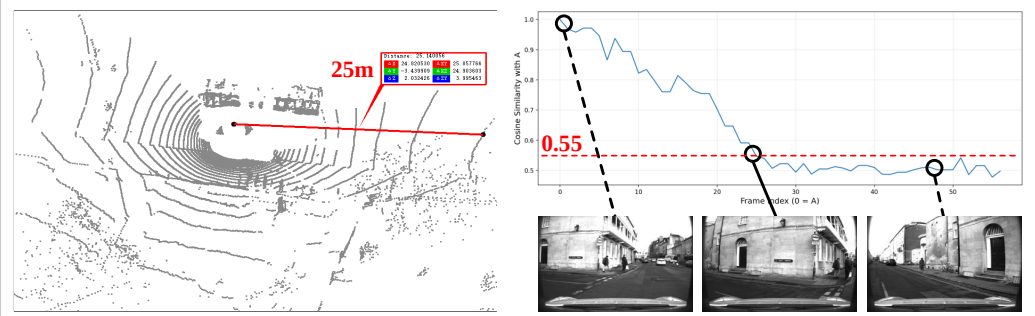
A Details of the Key-Frame Selection Mechanism

As outlined in the main manuscript, our temporal consistency querying mechanism replaces exhaustive per-frame VLM evaluation with a highly efficient spatial-visual key-frame strategy. A new key-frame $\mathcal{K}_i$ is triggered to capture updated VLM priors only when the sensor observation undergoes significant geometric or semantic shifts. The selection strictly depends on two complementary criteria: the LiDAR sparse Field of View (FoV) boundary and the visual similarity threshold. With the above keyframe selection strategy, the cost of invoking the VLM API is reduced by 95%.

A.1 Spatial Displacement and LiDAR Sparse FoV Boundary The point density of a LiDAR sensor inherently decays as the distance from the sensor center increases, leading to highly sparse and less reliable geometric structures at the periphery. To ensure sufficient yet non-redundant geometric overlap between consecutive evaluations, we define a spatial displacement threshold τ_{spatial} based on the effective dense radius of the LiDAR point cloud. Specifically, we calculate the radial distance from the sensor origin to the boundary where the point density drops below a reliable threshold for structural feature extraction. For the specific LiDAR sensors used in our benchmarks (Oxford and NCLT), as shown in Fig 8(A), we empirically measure this dense-to-sparse transition boundary at a distance of $d_{\text{sparse}} \approx 25.0$ meters. Consequently, a new key-frame is triggered when the accumulated translation from the previous key-frame $\mathcal{K}_{i-1}$ exceeds this boundary:

$$\Delta d = \|\mathbf{p}_t - \mathbf{p}_{\mathcal{K}_{i-1}}\|_2 \geq \tau_{\text{spatial}}, \tag{10}$$

where $\mathbf{p}_t$ and $\mathbf{p}_{\mathcal{K}_{i-1}}$ denote the positional coordinates of the current frame and the last key-frame, respectively, and we set $\tau_{\text{spatial}} = d_{\text{sparse}}$.



(A) Spatial Displacement (B) Visual Similarity

Figure 8: Visualization of Key-Frame Selection Mechanism

A.2 Visual Similarity Model and Threshold As shown in Fig 8(B), while the spatial threshold guarantees geometric coverage, outdoor environments frequently experience sudden semantic shifts (e.g., transient dynamic occlusions or turning intersections) that occur within a short spatial displacement. To capture these critical degeneracies, we incorporate a visual similarity criterion. We utilize a pre-trained DINOv2 [63] model to extract global image embeddings for each incoming camera frame. The visual similarity between the current frame I_t and the last key-frame $I_{\mathcal{K}_{i-1}}$ is computed using cosine similarity:

$$S_{\cos}(I_t, I_{\mathcal{K}_{i-1}}) = \frac{\mathbf{f}_t \cdot \mathbf{f}_{\mathcal{K}_{i-1}}}{\|\mathbf{f}_t\| \|\mathbf{f}_{\mathcal{K}_{i-1}}\|}, \tag{11}$$

where $\mathbf{f}$ represents the extracted feature vector. When a sudden scene transition occurs, the cosine similarity drops rapidly. A new key-frame is designated if $S_{\cos} < \tau_{\text{vis}}$, where τ_{vis} is an empirically defined threshold set to 0.55.

A.3 Unified Key-Frame Selection Mechanism The final designation of a new key-frame $\mathcal{K}_i$ is governed by a unified triggering mechanism that evaluates the spatial and visual criteria using a logical OR operation. Specifically, a new frame at time t is selected as the next key-frame if either the spatial distance limit is reached or a sudden visual shift is detected. This joint condition is formulated as:

$$\text{Trigger}(\mathcal{K}_i) = \begin{cases} \text{True}, & \text{if } \Delta d \geq \tau_{\text{spatial}} \vee S_{\cos}(I_t, I_{\mathcal{K}_{i-1}}) < \tau_{\text{vis}} \\ \text{False}, & \text{otherwise} \end{cases} \tag{12}$$

========== **Prompt** ==========

```
def create_prompt(image_name, previous_results=None):
    """Create prompt"""
base_prompt = """You are a top-tier expert in autonomous driving and LiDAR-based outdoor global relocalization (specifically, Scene Coordinate Regression methods). Your task is to evaluate ego-centric driving images and deduce whether the corresponding 3D point cloud of the current scene is UNFAVORABLE for point cloud relocalization.

[Evaluation Criteria]
Point cloud relocalization relies heavily on close-range, rich, and static geometric structures. Please score the "degree of unsuitability for localization" on a scale of 1 to 10. Higher scores indicate worse conditions for localization, representing more severe scene degradation.

Unfavorable Factors (High Scores: 6-10):
- Dynamic & Transient Interference : The field of view is dominated by a large number of parked vehicles, or large vehicles at close range (e.g., buses, trucks) that severely occlude structural lines of sight.
- Geometric Degeneracy: Highly repetitive and uniform architectural structures (e.g., featureless long walls), or extremely narrow, constrained road scenes lacking distinct landmarks.
- Temporal Scene Alterations: Areas with active construction, road maintenance, or other activities that significantly alter the permanent physical structure of the scene.

Favorable Factors (Low Scores: 1-3):
- Open field of view with rich, reliable static geometric features at close range, such as unique building facades, clear walls, fences, tree trunks, or pole-like objects (e.g., streetlights, traffic signs), with minimal occlusion.

[Historical Memory Reference]
To ensure temporal consistency, please refer to the evaluation results of the previous frames (if provided). If the current scene is physically continuous with the previous frames without abrupt spatial changes, maintain the smoothness and consistency of your evaluation logic."""

    if previous_results:
        base_prompt += "\n\nPrevious frames results:"
        for i, prev_result in enumerate(previous_results, 1):
            frame_info = {
                "Degeneracy_Short_Text": prev_result.get("Degeneracy_Short_Text", "N/A"),
                "Total_Score": prev_result.get("Total_Score", "N/A") }
            base_prompt += f"\nFrame -{i}: {json.dumps(frame_info, ensure_ascii=False)}"
    base_prompt += f"""

[Strict Output Format]
You MUST output ONLY a valid JSON object. Do NOT include any Markdown formatting (such as ```json), and do NOT add any conversational or explanatory filler text. Your output must strictly adhere to the following JSON structure:

{{
  "Image_Name": "{image_name}",
  "Brief_Spatial_Analysis": "[Analyze the static geometry, occlusions, and repetitive/construction degradation factors in 1-2 sentences]",
  "Degeneracy_Short_Text": "[Summarize the degradation type in a short phrase, e.g., Massive parked vehicle occlusion, Large bus occlusion, Repetitive uniform buildings, Narrow featureless road, Ideal static structure]",
  "Total_Score": [An integer between 1 and 10]
}}"""

    return base_prompt
```

Figure 9: The prompt for VLM-based open-vocabulary degraded scene reasoning.

This dual-criteria strategy creates a highly robust and synergistic approach to temporal consistency. The spatial threshold ($\tau_{\text{spatial}} = 25\text{m}$) acts as a dependable baseline, guaranteeing that the VLM priors are updated at maximum distance intervals to maintain sufficient geometric overlap as the vehicle traverses nominal, straight paths. Conversely, relying solely on distance is insufficient for highly dynamic outdoor environments. The visual similarity threshold ($\tau_{\text{vis}} = 0.55$) functions as a crucial asynchronous early-trigger mechanism. As demonstrated by the sharp drop in the cosine similarity curve in Fig 8(B), if the vehicle turns a sharp corner or suddenly encounters a massive dynamic occlusion, the visual context changes drastically long before the 25-meter spatial limit is reached. In such cases, the visual criterion immediately triggers a new key-frame. This ensures the VLM can rapidly assess the newly introduced degeneracies and provide updated guidance to the 3D localization network exactly when it is most needed. Together, these two complementary criteria achieve an approximately 95% reduction in VLM API invocation costs by eliminating redundant evaluations for stable scenes, while remaining highly sensitive to critical environmental shifts.

B Details of VLM Prompting and Degeneracy Reasoning

This section provides the implementation details of our Vision-Language Model (VLM) prompting strategy, which serves as the core of the proposed Open-Vocabulary Degeneracy Reasoning. The prompt is designed to translate 2D visual context into actionable 3D localization priors.

Design Philosophy of the Reasoning Prompt To ensure the VLM effectively assesses the "localizability" of a scene, the prompt incorporates a specialized persona and a multi-step reasoning logic. As shown in the template below, the VLM is assigned the role of a top-tier expert in LiDAR-based relocalization. The reasoning process is guided by explicit criteria that distinguish between favorable static structures (e.g., unique facades, poles) and unfavorable degradation factors. Crucially, the prompt categorizes scene degeneracies into three dimensions: (1) dynamic and transient occlusions, (2) geometric degeneracy such as repetitive structures, and (3) temporal alterations like road construc-

tion. These categories guide the model to perform a 2D-to-3D Chain-of-Thought (CoT) analysis, identifying factors that would compromise the accuracy of Scene Coordinate Regression.

Temporal Consistency and Historical Memory To prevent abrupt fluctuations in evaluation scores across continuous sequences, we implement a *Historical Memory Reference* mechanism within the prompt. When querying for the current key-frame $\mathcal{K}_i$, the evaluation results of the previous N key-frames (including their degradation descriptions and scores) are dynamically injected into the input. The VLM is explicitly instructed to maintain temporal smoothness and logical consistency unless a significant spatial or semantic shift is detected. This design ensures that the generated priors provide stable and robust guidance for the downstream Mixture-of-Experts (MoE) scheduling.

Output Structure and Downstream Integration To enable automated parsing and integration into our LiDAR-only backbone, we enforce a strict JSON output format. The output contains three essential fields that map directly to the variables in our methodology: `Brief_Spatial_Analysis` ($\mathcal{T}_i^{\text{content}}$), `Degeneracy_Short_Text` ($\mathcal{T}_i^{\text{desc}}$), and `Total_Score` ($s_i$). The complete prompt template used in our experiments is shown in Fig. 9.

C More Implementation

In this section, we provide further technical details regarding the training configuration, data augmentation, and specific neural network architectures utilized in ViLo.

C.1 Training Reproducibility and Seeds To ensure the reproducibility of our experimental results, we fix the random seed to 2089 for all training sessions across the Oxford and NCLT datasets. This includes the initialization of all network weights, data shuffling, and stochastic augmentation processes.

C.2 Data Augmentation (Training Only) During the training phase, we apply several data augmentation techniques to the input LiDAR point clouds to enhance the model's geometric invariance. These operations are strictly disabled during inference to ensure deterministic localization.

- **Translation**: Random translation is applied with a probability $p = 0.5$ within the range $[-1m, 1m]$ along the x and y axes only, simulating varying vehicle lateral positions.
- **Rotation**: Random yaw rotation is applied with $p = 0.5$ within the range $[-10°, 10°]$ to account for orientation drift.
- **Feature Pooling**: We employ `MinkowskiAvgPooling` with a kernel size of 8 and a stride of 8 to downsample the high-dimensional sparse feature maps, effectively increasing the receptive field while maintaining computational efficiency.

C.3 Cross-modal Distillation Projector As illustrated in Fig. 10, the projection head $\text{Proj}(\cdot)$ is explicitly designed to bridge the complex modality gap between the 3D LiDAR geometric space and the VLM's text-aligned latent space. The network takes the global LiDAR feature $\boldsymbol{f} \in \mathbb{R}^{512}$ as input. It is initially projected into a 256-dimensional space via a linear layer followed by a Normalization layer, yielding the base representation $\bar{\boldsymbol{f}} \in \mathbb{R}^{256}$. To learn the highly non-linear mapping required for cross-modal semantic alignment, $\bar{\boldsymbol{f}}$ is subsequently processed by a deep network comprising 3 stacked residual blocks. Each block features an inverted bottleneck architecture: $\text{Linear}(256, 512) \rightarrow \text{ReLU} \rightarrow \text{Linear}(512, 256) \rightarrow \text{Dropout}$. A residual skip connection adds the block's input to the Dropout output, which is then passed through another Normalization layer. This deep, residual design prevents vanishing gradients and ensures stable optimization when calculating the cosine similarity-based distillation loss $\mathcal{L}_{\text{distill}}$ against the 256-dimensional VLM scene semantic embedding $\boldsymbol{t}_i$.

C.4 Architectural Details of Routing and Gating Modules The Scene-Adaptive MoE relies on specialized sub-networks to manage expert distribution and VLM prior integration.

Router Network Structure: As illustrated in Fig. 11, our router network architecture takes the VLM prior $\boldsymbol{t}_i \in \mathbb{R}^{256}$ and the point cloud feature $\bar{\boldsymbol{f}}_i \in \mathbb{R}^{256}$ as inputs. These two modality vectors are explicitly concatenated to form a 512-dimensional joint representation. Subsequently, this fused

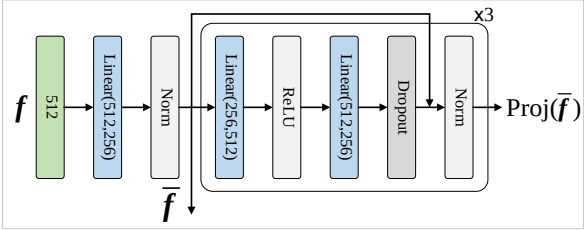


Figure 10: Projector Network Structure.

feature is fed into a Multi-Layer Perceptron designed with progressively decreasing dimensions. The network layers are interleaved with ReLU activation functions and a Dropout layer with a rate of 0.1 to prevent model overfitting. Finally, the expert routing probabilities are computed via a Softmax function.

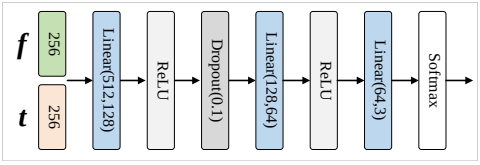


Figure 11: Router Network Structure.

Adaptive Gate Structure: As illustrated in Fig. 12, our adaptive gate dynamically controls the influence of the VLM prior based on the current environmental context. Similar to the router, the gate effectively incorporates both geometric and semantic information by explicitly concatenating the point cloud feature $\bar{\boldsymbol{f}}_i \in \mathbb{R}^{256}$ and the VLM prior $\boldsymbol{t}_i \in \mathbb{R}^{256}$ into a 512-dimensional joint representation. This fused vector is then processed by an MLP bottleneck. It first passes through a linear projection to a 64-dimensional hidden space, followed by a ReLU activation. Finally, a linear layer coupled with a Sigmoid activation function computes the final scalar gating weight bounded within $(0, 1)$.

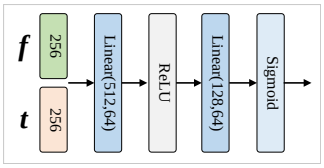


Figure 12: Gating Network Structure.

D Details of the Boreas Dataset and Extreme Condition Evaluation

To comprehensively evaluate the robustness of our VLM-based 2D-to-3D prior knowledge distillation under adverse environmental conditions, we leverage the Boreas dataset for our ablation study on nighttime and rainy conditions.

Boreas Dataset Characteristics The Boreas dataset is uniquely structured to test long-term environmental adaptability. It consists of over 350 km of driving data collected by traversing a repeated route over the course of an entire year. This systematic collection captures stark seasonal variations and severe weather conditions, including rain, snow, and nighttime illumination drops. The dataset features a high-quality sensor configuration, notably a 128-beam Velodyne Alpha-Prime LiDAR that provides highly dense structural measurements, a 5MP camera for visual reference, and centimeter-accurate post-processed GNSS/IMU data for reliable ground truth poses.

Training and Evaluation Protocol Because ViLo operates strictly as a LiDAR-only framework during inference, evaluating it on the Boreas dataset serves as a rigorous stress test. It validates



Figure 13: Boreas Dataset Scenes. We train under (A) cloudy, (B) sunny, and (C) sunny conditions, and perform inference under (D) sunny, (E) night, and (F) rainy conditions.

whether the 3D backbone has successfully internalized the VLM's semantic priors and can function robustly without any runtime dependency on visual inputs which natively fail at night or when LiDAR point clouds are corrupted by precipitation. To set up this evaluation, we carefully partitioned the Boreas sequences to create a stark domain gap between the training and inference sequences (As shown in Fig. 13):

- **Training Sequences:** The model is trained exclusively on daytime sequences featuring benign weather. This includes a cloudy sequence (A: 2020-11-26-13-58) and two sunny sequences (B: 2021-03-23-12-43 and C: 2021-03-30-14-23).
- **Inference Sequences:** Inference is performed on entirely distinct sequences to test true generalization. We evaluate on a baseline sunny scenario (D: 2021-04-13-14-49), a pitch-dark nighttime scenario (E: 2021-09-14-20-00), and a rainy scenario (F: 2021-04-29-15-55).

By achieving comparable metric localization accuracy across these highly divergent testing splits, we demonstrate that the Scene-Adaptive MoE effectively leverages the distilled priors to dynamically route features, ensuring robust global relocalization entirely independent of ambient illumination or precipitation interference during deployment.

E Verification of Cross-Modal Prior Distillation

A critical contribution of ViLo is the internalization of VLM scene reasoning into the 3D LiDAR backbone via the distillation projector $\text{Proj}(\cdot)$. To empirically validate that the projected LiDAR feature $\text{Proj}(\boldsymbol{f}_i)$ successfully learns and retains the VLM prior text feature $\boldsymbol{t}_i$ without relying on images during inference, we conduct both qualitative and quantitative feature alignment analyses on randomly sampled key-frames from the test set.

E.1 t-SNE Feature Space Alignment To visually inspect the latent space geometry, we employ t-SNE to project the 256-dimensional distilled LiDAR features $\text{Proj}(\boldsymbol{f})$ and the corresponding VLM prior features $\boldsymbol{t}$ into a 2D space, as illustrated in Fig. 14. The blue triangles represent the distilled 3D features, while the orange circles denote the target VLM scene embeddings. The thin gray lines connect features extracted from the exact same key-frame. As observed, the distribution of $\text{Proj}(\boldsymbol{f})$ heavily overlaps with the distribution of $\boldsymbol{t}$, forming tightly coupled clusters. The remarkably short connection distances between corresponding modality pairs demonstrate that the cross-modal projection head effectively bridges the modality gap. This confirms that the LiDAR-only backbone successfully constructs a latent representation that highly mimics the semantic scene comprehension of the VLM.

E.2 Cosine Similarity Statistics To quantitatively measure the effectiveness of this semantic alignment, we compute the cosine similarity between the distilled features $\text{Proj}(\boldsymbol{f})$ and the VLM priors $\boldsymbol{t}$ across the sampled key-frames. As shown in the statistical bar chart in Fig. 15, the vast majority of samples maintain a highly stable similarity score, achieving an impressive overall mean

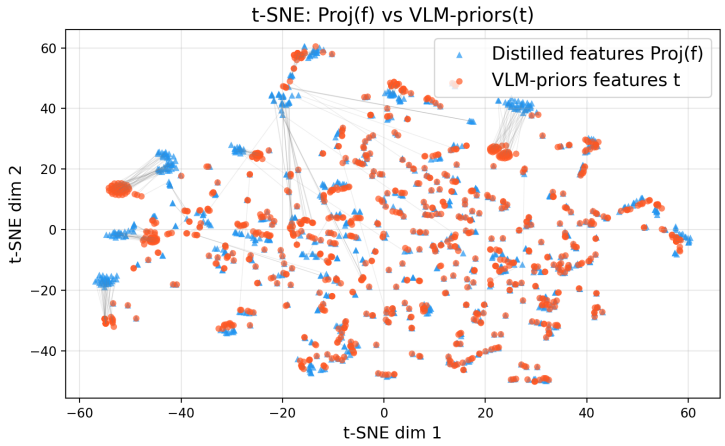


Figure 14: The t-SNE visualization results of the distilled features proj(f) and the VLM priors t.

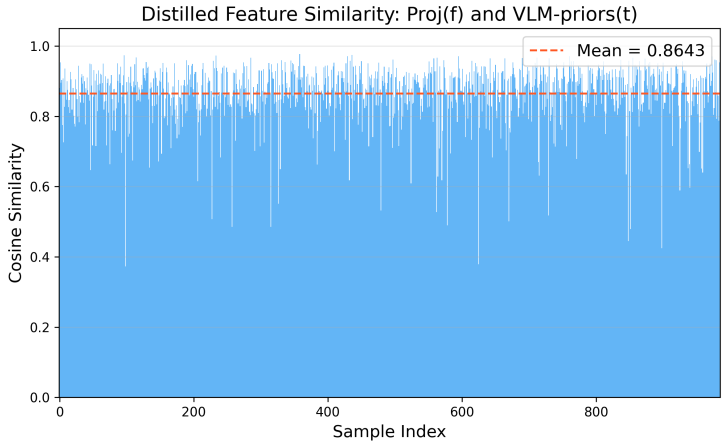


Figure 15: Feature similarity between distilled features proj(f) and VLM priors.

cosine similarity of 0.86. In high-dimensional latent spaces (256-dim), achieving a mean cosine similarity approaching 0.9 is exceptionally challenging and signifies a robust semantic mapping. This statistical evidence strongly supports our claim that the distillation loss $\mathcal{L}_{\text{distill}}$ enables the 3D network to consistently and reliably output VLM-equivalent degeneracy priors across diverse outdoor scenarios, fully supporting the downstream Scene-Adaptive MoE routing.

F Limitations

Since the reliability priors during the training phase heavily rely on the VLM's visual perception, the performance ceiling of ViLo is inherently constrained by the visual quality of the initial training trajectory, potentially limiting the network's generalization in specific long-tail scenarios where signal distortion of the teacher model occurs under nighttime or extreme illumination-degraded conditions.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly reflect the contributions and scope of this paper.

   Guidelines:

   - The answer [NA] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [NA] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In Appendix F, we discuss the limitations of our work.

   Guidelines:

   - The answer [NA] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: Our work does not include theoretical results.

Guidelines:

- The answer [NA] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We have disclosed the experimental protocols, dataset splits, implementation details, and hyperparameters needed to reproduce the main results.

   Guidelines:

   - The answer [NA] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: the code be open-sourced upon acceptance of the paper.

Guidelines:

- The answer [NA] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main details are shown in Section 4 and Appendix C.

   Guidelines:

   - The answer [NA] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Following common practice among outdoor LiDAR relocalization benchmarks, we do not report error bars; per-sequence results are provided for the main comparisons.

   Guidelines:

   - The answer [NA] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Details of the computational resources and runtime-memory statistics are provided in Sections 4.4 , with complementary storage details in Appendix 4.

   Guidelines:

   - The answer [NA] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our work conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [NA] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No] , they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of this work.

    Guidelines:

    - The answer [NA] means that there is no societal impact of the work performed.
    - If the authors answer [NA] or [No] , they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [NA] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original papers that produce the codes and datasets.

    Guidelines:

    - The answer [NA] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: We will be open-sourced upon acceptance of the paper.

    Guidelines:

    - The answer [NA] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [NA] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [NA] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [NA] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.