LINK: Learning to Localize from Known to Unknown Scenes

Minghang Zhu[1,2,*] **Kaibo Jin**[1,2,*] **Zhijing Wang**[1,2] **Ziwei Shi**[1,2]
Wen Li[3] **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

mihoo@stu.xmu.edu.cn kaibooo@stu.xmu.edu.cn

Abstract

LiDAR relocalization based on scene coordinate regression (SCR) degrades sharply when test trajectories traverse route segments absent from training, a practical challenge in long-range autonomous-driving deployment, where complete route coverage is rarely achievable. We present LINK, a confidence-guided framework for relocalization under partial scene coverage. LINK predicts an absolute pose hypothesis and scene features with an SCR backbone, estimates pose reliability from temporal pose–feature sequences, and uses a decision policy to arbitrate between direct absolute hypotheses and relative motion propagated from reliable historical estimates, thereby preserving globally referenced localization across uncovered segments. We also introduce UrbanUnseen, a city-scale benchmark with native known-to-unknown scene transitions, together with controlled partial-coverage protocols on Quality-enhanced Oxford and NCLT. Experiments show that LINK substantially improves robustness when training coverage is incomplete, consistently outperforms representative map-free baselines under partial coverage, and remains effective in fully known scenes. Project page: `https://reckozmh.github.io/LINK/`.

1 Introduction

Reliable LiDAR relocalization is a fundamental capability for robots and autonomous vehicles operating in large-scale outdoor environments [44, 46, 50]. Existing approaches are commonly divided into map-based and map-free paradigms [17, 40, 43, 23, 41]. Map-based methods localize by matching the current scan against a pre-built 3-D map [29, 48], but storing, maintaining, and querying explicit maps and features introduces nontrivial deployment costs in long-range applications [8]. Map-free methods avoid explicit map access at test time and instead infer globally referenced pose hypotheses directly from LiDAR observations [47]. Among them, scene coordinate regression (SCR) pipelines [12, 32, 21, 23, 41] have shown promising accuracy and efficiency for outdoor relocalization.

However, existing map-free SCR methods still rely on an implicit assumption: the training trajectories sufficiently cover the scenes encountered during inference [15, 30, 24, 52]. Exhaustive route coverage is expensive and often impractical, so a test trajectory may traverse road segments absent from training [3, 7]. We refer to this setting as partial scene coverage, and denote training-covered and training-uncovered route segments as known and unknown scenes, respectively. Under known-to-unknown transitions, direct SCR may still output an absolute pose, but the prediction can become unreliable because the model must extrapolate beyond the scene distributions it has learned [4, 15,

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

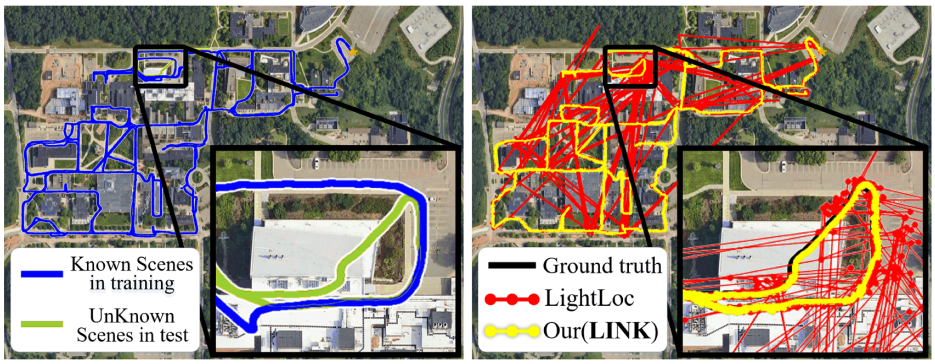


Figure 1: The impact of unknown scenes on LiDAR localization. Under a partially known NCLT protocol, some test segments are absent from training. LightLoc [23] produces frequent errors in unknown regions, whereas LINK maintains better trajectory consistency.

30, 51]. The central challenge is therefore not merely improving regression accuracy in known scenes, but maintaining globally referenced localization when coverage changes along a route. More generally, scene membership is only a proxy for pose quality: SCR hypotheses may also become unstable on nominally covered long routes. Fig. 1 illustrates the issue: under the partially known NCLT protocol, LightLoc [23] drifts severely after entering training-uncovered segments.

Addressing this challenge requires two coupled capabilities. First, the system must estimate whether the current absolute pose hypothesis remains trustworthy. Unreliable SCR hypotheses are difficult to identify from a single scan alone, since failures often appear as temporal instability rather than obvious per-frame appearance changes; temporal cues have therefore become important in localization [14, 53, 45, 13]. Second, once confidence degrades, the system should decide whether to accept direct SCR localization or propagate from a reliable historical pose. This is a sequential problem: switching depends on confidence history, reference-pose availability, and accumulated odometric drift. A practical solution should therefore combine temporal pose-reliability estimation with sequential mode selection.

To this end, we present LINK, a confidence-guided hybrid relocalization framework for map-free LiDAR relocalization under partial scene coverage. LINK uses an SCR backbone to produce absolute pose hypotheses and scene features, estimates pose reliability from temporal pose–feature consistency as a soft score rather than a hard known/unknown label, and uses a decision policy to choose between direct SCR localization and odometric propagation from reliable historical poses. In this way, LINK maintains a globally referenced trajectory across uncovered segments while suppressing unstable SCR hypotheses.

Existing evaluation protocols are also insufficient for assessing robustness under partial scene coverage. Public outdoor localization benchmarks [3, 7, 6, 16] do not explicitly separate training-covered and training-uncovered route segments. We therefore introduce UrbanUnseen, a city-scale LiDAR relocalization benchmark with route-level known/unknown transitions, and establish known/unknown protocols on Oxford [3] and NCLT [7] under random and manual partitions. Throughout the paper, Oxford denotes the Oxford Radar RobotCar benchmark with trajectory-aligned ground-truth poses following SGLoc [21]; Appendix A.1 provides the details. Together, these settings cover fully known routes, controlled partial coverage, and route-level known/unknown transitions.

The main contributions of this work are summarized as follows:

- We formulate map-free LiDAR relocalization under partial scene coverage and establish known/unknown evaluation protocols on Oxford and NCLT, including both random and manual partitions.
- We propose LINK, a confidence-guided hybrid relocalization framework that estimates pose reliability from temporal pose–feature consistency and adaptively switches between direct SCR localization and odometric propagation from reliable historical poses.
- We introduce UrbanUnseen and provide systematic evaluations on Oxford, NCLT, and UrbanUnseen, showing that LINK improves robustness under incomplete coverage while remaining competitive in fully known scenes.

2 Related Work

Relocalization paradigms. LiDAR relocalization methods can be broadly grouped into map-based and map-free paradigms. Map-based methods estimate the global pose by matching a query scan against a pre-built representation, typically retrieving candidate places and then refining the pose by registration. PointNetVLAD [33], MinkLoc3D [20], SOE-Net [38], and RING++ [40] learn or design global descriptors for place retrieval, while SpinNet [1], TEASER++ [42], and Buffer [2] improve geometric verification and registration. These methods can be accurate, but explicit map storage, maintenance, and search introduce deployment costs in long-range settings [8, 43]. Map-free relocalization avoids explicit map access by directly predicting globally referenced poses from LiDAR observations. APR methods such as PointLoc [36], STCLoc [45], HypLiLoc [35], NIDALoc [47], and DiffLoc [22] regress poses directly, while SCR methods build on scene-coordinate prediction [32, 12, 5]. Recent outdoor SCR systems, including SGLoc [21], LiSA [41], and LightLoc [23], improve accuracy and efficiency without explicit maps. However, these map-free methods are still mostly evaluated under the assumption that training and test routes share sufficient scene coverage, leaving their behavior under incomplete coverage less understood.

Reliability and hybrid localization. Prior work has long recognized that pose predictions need reliability estimates. Kendall and Cipolla [14] modeled predictive uncertainty for pose regression, KFNet [53] incorporated temporal filtering into relocalization, and STCLoc [45] showed that temporal consistency can improve LiDAR localization. Reliability cues also arise in retrieval and registration: Vidanapathirana et al. [34] re-ranked retrieved scans through spectral geometric verification, while Huang et al. [13] used consistency-aware modeling to improve registration reliability. These studies suggest that temporal instability and geometric inconsistency are informative cues for deciding whether a current pose estimate should be trusted. Practical systems also often combine absolute and relative localization. Voxelized GICP [19] provides efficient relative motion estimation, and graph-based optimization [37] improves long-horizon trajectory stability. However, such hybridization usually assumes either explicit maps or sufficiently reliable absolute updates. Under partial scene coverage, the key question becomes when a map-free absolute hypothesis should be accepted and when localization should instead be propagated from a reliable historical pose. This decision is not well captured by frame-wise confidence thresholding because its consequence depends on confidence history, reference-pose availability, and accumulated odometric drift. Existing work therefore provides useful ingredients for reliability estimation and trajectory stabilization, but reliability-guided absolute/relative mode selection as a sequential decision problem for map-free LiDAR relocalization remains less directly explored.

Benchmarks and protocols. Oxford [3, 21] and NCLT [7] remain the standard repeated-route benchmarks for outdoor map-free LiDAR relocalization. SGLoc [21], LiSA [41], DiffLoc [22], and LightLoc [23] all report their primary results on these datasets. Boreas [6] expands seasonal and spatial diversity, whereas MulRan [16] supports urban localization and place-recognition research. However, these benchmarks do not explicitly separate training-covered and training-uncovered route segments. As a result, robustness to partial scene coverage is usually tested only indirectly, for example by repartitioning existing trajectories. This distinction matters because a method may remain competitive in fully known scenes yet fail once the route enters unknown segments. A complementary evaluation setting is therefore needed to cover conventional fully known relocalization, controlled partial-coverage protocols, and native route-level known-to-unknown transitions.

3 Method

3.1 Overall Framework

Under partial scene coverage, a test trajectory may traverse both training-covered and training-uncovered route segments. Our objective is to output a globally referenced pose for every frame while reducing large localization errors once direct SCR localization becomes unreliable. The overall pipeline is shown in Fig. 2.

Given the current LiDAR scan $\mathbf{x}_t$, the SCR backbone predicts an absolute pose hypothesis $\mathbf{p}_t^{\mathrm{scr}} = (\mathbf{t}_t^{\mathrm{scr}}, \mathbf{q}_t^{\mathrm{scr}})$ and an intermediate feature map $\mathbf{F}_t \in \mathbb{R}^{N \times C}$, where N and C denote the number of feature tokens and the feature-channel dimension. We instantiate the backbone with LightLoc [23];

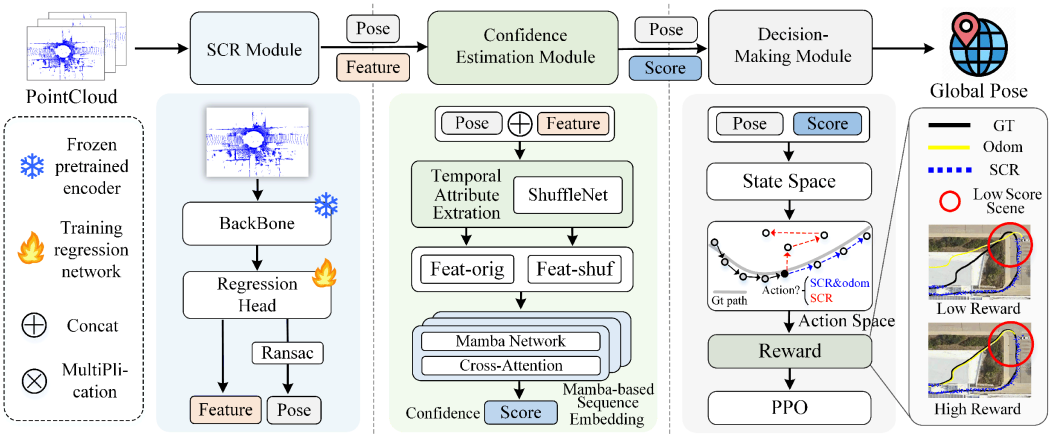


Figure 2: Overview of LINK. For each incoming LiDAR scan, the SCR backbone predicts an absolute pose hypothesis together with scene features. A temporal reliability estimator evaluates whether that hypothesis remains trustworthy, and the decision policy then chooses between direct SCR localization and odometric propagation from a reliable historical pose. The reward and PPO branches are used only during policy training.

training details are provided in Section 5.1. The predicted pose provides an efficient map-free absolute hypothesis, while the feature map is retained for temporal reliability estimation.

A reliability estimator produces a soft score $c_t \in [0, 1]$ from a temporal pose–feature sequence. The decision policy then selects one of two localization modes: accepting the current SCR hypothesis, or propagating a stored reliable pose through relative odometry. The final output pose is:

$$\hat{\mathbf{p}}_t = \begin{cases} \mathbf{p}_t^{\mathrm{scr}}, & a_t = 0, \\ \mathbf{p}_{\tau_k}^{\mathrm{ref}} \oplus \Delta\mathbf{p}_{\tau_k \to t}^{\mathrm{odom}}, & a_t = k,\ k \in \{1, \ldots, K\}, \end{cases} \tag{1}$$

where a_t is the action selected by the policy, K is the number of candidate reference poses, $\mathbf{p}_{\tau_k}^{\mathrm{ref}}$ is the k-th candidate reference pose stored in a fixed-size buffer of recent high-confidence poses, $\Delta\mathbf{p}_{\tau_k \to t}^{\mathrm{odom}}$ is the odometric motion from time τ_k to the current frame t, and $\oplus$ denotes rigid-body pose composition. Thus, LINK remains map-free at inference while reducing exclusive reliance on SCR predictions in uncovered route segments.

3.2 Temporal Pose-Reliability Estimation

Unreliable SCR predictions under partial scene coverage are often exposed by temporal instability: pose hypotheses may fluctuate or drift even when individual scans remain locally plausible. We therefore estimate reliability from the ordered evolution of pose hypotheses and scene descriptors instead of from a single scan. Because this reliability score is evaluated online for every frame, the temporal module should capture pose–feature dynamics with limited inference overhead. We use shared Mamba blocks to model the ordered pose–feature sequence, followed by cross-attention fusion to compare the original stream with a lightweight perturbed companion stream.

For each frame, the intermediate feature map is compressed by a learnable operator $G(\cdot)$ and concatenated with the normalized SCR pose:

$$\begin{aligned} \mathbf{f}_t &= G(\mathbf{F}_t) \in \mathbb{R}^d, \\ \mathbf{z}_t &= [\tilde{\mathbf{p}}_t^{\mathrm{scr}}; \mathbf{f}_t] \in \mathbb{R}^{d+7}, \\ \mathbf{Z}_t^{\mathrm{orig}} &= [\mathbf{z}_{t-T+1}, \ldots, \mathbf{z}_t] \in \mathbb{R}^{T \times (d+7)}, \end{aligned} \tag{2}$$

where T is the temporal window length, d is the compressed descriptor dimension, and $\tilde{\mathbf{p}}_t^{\mathrm{scr}}$ denotes the normalized 7-D pose vector. We also construct a companion stream $\mathbf{Z}_t^{\mathrm{pert}} = \mathrm{Shuffle}(\mathbf{Z}_t^{\mathrm{orig}})$ using a lightweight feature-channel shuffle inspired by ShuffleNet [27]. This perturbation stream preserves frame order while disturbing feature channels, providing a consistency probe for the fusion module.

The original and perturbed streams are encoded by shared Mamba blocks [11] and fused by cross-attention. Denoting this encoder by $\Phi_{\mathrm{MCA}}(\cdot,\cdot)$, the reliability score is:

$$\begin{aligned} \mathbf{u}_t &= \Phi_{\mathrm{MCA}}\left(\mathbf{Z}_t^{\mathrm{orig}}, \mathbf{Z}_t^{\mathrm{pert}}\right), \\ c_t &= \sigma(\mathrm{MLP}\left(\mathrm{Pool}(\mathbf{u}_t)\right)), \end{aligned} \tag{3}$$

where $\mathrm{Pool}(\cdot)$ denotes temporal average pooling, $\mathrm{MLP}(\cdot)$ is a two-layer perceptron, and $\sigma(\cdot)$ is the sigmoid function. The output c_t is a soft pose-reliability score, not a hard known/unknown scene label. Implementation details are provided in Appendix B.2.

3.3 Confidence-Guided Localization Policy

Mode selection is formulated as a sequential decision problem because each action depends on recent confidence, available reference poses, and accumulated odometric drift. We use the state

$$\mathbf{s}_t = [c_t, c_{t-1}, \ldots, c_{t-T_h+1}, \boldsymbol{\psi}_t], \quad \boldsymbol{\psi}_t = [m_{1,t}, \delta_{1,t}, \ldots, m_{K,t}, \delta_{K,t}], \tag{4}$$

where T_h is the history length, $m_{k,t}$ marks whether the k-th candidate reference pose is valid, and $\delta_{k,t}$ is its normalized span to the current frame. The action space is $\mathcal{A} = \{0, 1, \ldots, K\}$: $a_t = 0$ accepts $\mathbf{p}_t^{\mathrm{scr}}$, while $a_t = k$ composes the k-th reference pose with odometry as in Eq. (1). For the action-conditioned pose $\hat{\mathbf{p}}_t(a_t) = (\hat{\mathbf{t}}_t(a_t), \hat{\mathbf{q}}_t(a_t))$ and ground truth $\mathbf{p}_t^* = (\mathbf{t}_t^*, \mathbf{q}_t^*)$, the reward is

$$\mathcal{R}(\mathbf{s}_t, a_t) = -\left(w_T \ell_t^T + w_R \ell_t^R + w_d \ell_t^d\right), \tag{5}$$

where $\ell_t^T = \|\hat{\mathbf{t}}_t(a_t) - \mathbf{t}_t^*\|_2$, $\ell_t^R = 2\arccos(|\langle \hat{\mathbf{q}}_t(a_t), \mathbf{q}_t^* \rangle|)$, and ℓ_t^d penalizes odometric span: $\ell_t^d = 0$ for $a_t = 0$ and $\ell_t^d = t - \tau_k$ when $a_t = k > 0$. The scalar weights w_T, w_R, and w_d balance immediate pose accuracy and accumulated drift; small-error cases are mapped to zero reward and large errors are clipped at -5. The policy is optimized with PPO [31] during training only; inference uses the learned policy without ground-truth poses or rewards.

3.4 Training Protocol

LINK is trained in stages. The SCR backbone is first trained on the known-scene subset of each target dataset. Following the LightLoc protocol [23], the SCR pose supervision loss is:

$$\mathcal{L}_{\mathrm{SCR}} = \frac{1}{M_{\mathrm{scr}}} \sum_{i=1}^{M_{\mathrm{scr}}} \|\mathbf{p}_i^{\mathrm{scr}} - \mathbf{p}_i^*\|_1, \tag{6}$$

where M_{scr} is the number of SCR training scans, $\mathbf{p}_i^{\mathrm{scr}}$ is the predicted 7-D pose, and $\mathbf{p}_i^*$ is the ground-truth pose. After the SCR backbone is frozen, the reliability estimator is trained with soft targets derived from frozen-SCR pose errors on reliability-training samples with ground-truth poses, so confidence measures localization reliability rather than scene membership. For each frame, we define the combined SCR pose error e_t and map it to a soft confidence target using a Gaussian kernel:

$$e_t = \ell_t^{T,\mathrm{scr}} + \beta \ell_t^{R,\mathrm{scr}}, \qquad \hat{c}_t = \exp\left(-\frac{e_t^2}{2\sigma_e^2}\right), \tag{7}$$

where $\ell_t^{T,\mathrm{scr}}$ and $\ell_t^{R,\mathrm{scr}}$ are the translation and rotation errors of the SCR hypothesis, β balances the two units, and σ_e is estimated from empirical SCR error statistics on the reliability-training samples. The reliability estimator is supervised by mean-squared regression:

$$\mathcal{L}_{\mathrm{CE}} = \frac{1}{M_{\mathrm{ce}}} \sum_{t=1}^{M_{\mathrm{ce}}} (c_t - \hat{c}_t)^2, \tag{8}$$

where M_{ce} is the number of reliability-training samples. Finally, with the SCR backbone and reliability estimator fixed, the decision policy is optimized with PPO using the reward in Eq. (5). The clipped PPO objective, generalized advantage estimation, reference-pose state construction, and odometry and inference-time state details are provided in Appendix B.3 and Appendix B.4.

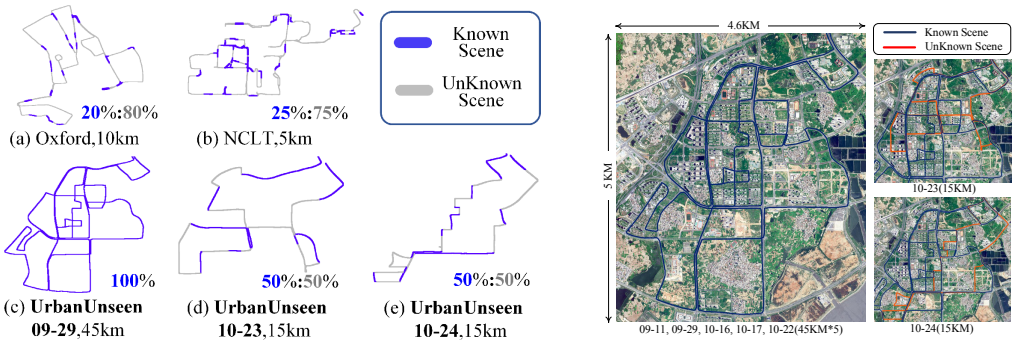


Figure 3: Dataset partition and UrbanUnseen benchmark visualization. Left: panels (a) and (b) show Oxford and NCLT random partitions (average known-scene coverage: 20.8% and 25.7%; in-panel percentages are approximate); panels (c)–(e) show UrbanUnseen partitions. Blue denotes known route segments retained for training; gray denotes unknown segments outside the training support. Right: the UrbanUnseen satellite-map overview. Full-coverage routes are used for training and fully known evaluation; routes 10-23 and 10-24 contain natural known-to-unknown transitions following the dark-blue/red legend.

4 UrbanUnseen and Evaluation Protocols

Evaluation labels and regimes. We operationalize partial scene coverage at the route level. A test frame is labeled as known if its ground-truth position lies inside the spatial support used for training under the chosen protocol, and as unknown otherwise. These labels are used only for protocol construction and diagnostic reporting; they are not provided to LINK or to any baseline during inference. We evaluate three regimes. The fully known regime follows the conventional repeated-route setting, where the test route is sufficiently covered by training data. The controlled partially known regime is constructed on Oxford [3, 21] and NCLT [7] by retaining only selected spatial regions as known during training while evaluating on the full test trajectories. The UrbanUnseen regime provides route-level known/unknown transitions that occur naturally in the collected routes.

UrbanUnseen. UrbanUnseen is a city-scale LiDAR relocalization benchmark designed to evaluate both repeated-route relocalization and route-level partial scene coverage. It covers approximately 23 km^2 and contains seven trajectories. Five full-coverage routes, 09-11, 09-29, 10-16, 10-17, and 10-22, span approximately 44–47 km each. Two additional routes, 10-23 and 10-24, are 15 km long and each contains approximately 50% unknown scenes. We use 09-11, 10-16, 10-17, and 10-22 for training, 09-29 for fully known evaluation, and 10-23 and 10-24 for route-level partially known evaluation. The right side of Figure 3 provides a satellite-map overview of these routes and their known/unknown splits. All experiments in this paper use the Ouster OS1-128 LiDAR stream, with LiDAR-frame poses obtained by temporal alignment and interpolation from the 100 Hz FDI-Sigma navigation system. Additional acquisition details, sensor availability, trajectory statistics, and split definitions are provided in Appendix A.2.

Controlled protocols on public benchmarks. We evaluate controlled partial coverage on Oxford [3, 21] and NCLT [7]. For Oxford, all experiments use the trajectory-aligned ground-truth poses described in Appendix A.1. Since Oxford and NCLT do not explicitly separate training-covered and training-uncovered route segments, we construct two controlled partial-coverage protocols. In the random protocol, world-coordinate centers are sampled along the training trajectories, and scans within a 30 m radius are retained as known. In the manual protocol, known regions are selected around road intersections to approximate a deployment setting where data collection may concentrate around traffic-interaction areas rather than uniformly covering the route. In both protocols, the initial route segment is included in the known support to provide a valid global localization start. The average known-scene ratios are 20.8% on Oxford and 25.7% on NCLT under the random protocol, and 31.4% on Oxford and 22.1% on NCLT under the manual protocol. Panels (a)–(e) of Figure 3 visualize the main known/unknown partitions used in these protocols.

Exact public-benchmark sequence IDs, per-sequence known-scene ratios, and additional partition visualizations are provided in Appendices A.3 and A.4.

Table 1: Results on Oxford under fully known and random partial-coverage protocols.

Protocol	Method	15-13-06-37↓	17-13-26-39↓	17-14-03-00↓	18-14-14-42↓	Average [m/°]↓
Fully Known	SGLoc (CVPR'23) [21]	1.79/1.67	1.81/1.76	1.33/1.59	1.19/1.39	1.53/1.60
	DiffLoc (CVPR'24) [22]	2.03/**1.04**	1.78/**0.79**	2.05/**0.83**	1.56/**0.83**	1.86/**0.87**
	LiSA (CVPR'24) [41]	0.94/1.10	1.17/1.21	0.84/1.15	0.85/1.11	0.95/1.14
	LightLoc (CVPR'25) [23]	0.82/1.12	0.85/1.07	0.81/1.11	0.82/1.16	0.83/1.12
	LINK	**0.75**/1.12	**0.70**/1.05	**0.70**/1.08	**0.73**/1.14	**0.72**/1.10
Only 20.8% Scenes Known	SGLoc+FG (CVPR'23) [21]	7.03/4.16	7.41/4.53	7.30/4.21	7.09/4.11	7.21/4.25
	DiffLoc+FG (CVPR'24) [22]	7.23/4.19	6.86/3.84	6.90/4.33	7.01/3.97	7.00/4.08
	LiSA+FG (CVPR'24) [41]	6.44/3.81	7.28/4.20	7.35/4.10	6.78/4.00	6.96/4.03
	LightLoc+FG (CVPR'25) [23]	7.12/3.72	7.75/4.09	5.66/4.04	7.26/4.18	6.94/4.00
	LINK	**2.26/3.43**	**2.77/3.72**	**2.16/3.91**	**1.77/3.52**	**2.24/3.64**

Table 2: Results on NCLT under fully known and random partial-coverage protocols.

Protocol	Method	2012-02-12↓	2012-02-19↓	2012-03-31↓	2012-05-26↓	Average [m/°]↓
Fully Known	SGLoc (CVPR'23) [21]	1.20/3.08	1.20/3.05	1.12/3.28	3.81/4.74	1.83/3.54
	DiffLoc (CVPR'24) [22]	0.99/2.40	0.92/2.14	0.98/2.27	**1.88**/2.43	**1.19/2.31**
	LiSA (CVPR'24) [41]	0.97/**2.23**	0.91/**2.09**	0.87/**2.21**	3.30/2.84	1.51/2.34
	LightLoc (CVPR'25) [23]	0.98/2.76	0.89/2.51	0.86/2.67	3.86/3.51	1.64/2.86
	LINK	**0.75**/2.31	**0.66**/2.22	**0.76**/2.57	2.94/**2.39**	1.27/2.37
Only 25.7% Scenes Known	SGLoc+FG (CVPR'23) [21]	5.47/6.83	5.25/6.36	5.04/6.97	7.29/7.31	5.76/6.87
	DiffLoc+FG (CVPR'24) [22]	4.65/6.28	4.15/5.87	3.48/4.29	6.41/5.48	4.67/5.48
	LiSA+FG (CVPR'24) [41]	4.90/5.72	5.61/5.63	4.55/5.60	6.20/5.31	5.32/5.57
	LightLoc+FG (CVPR'25) [23]	3.61/5.84	3.62/3.82	3.55/4.38	6.36/5.69	4.29/4.93
	LINK	**2.27/2.85**	**2.22/2.84**	**2.42/3.27**	**2.66/3.17**	**2.39/3.03**

5 Experiments

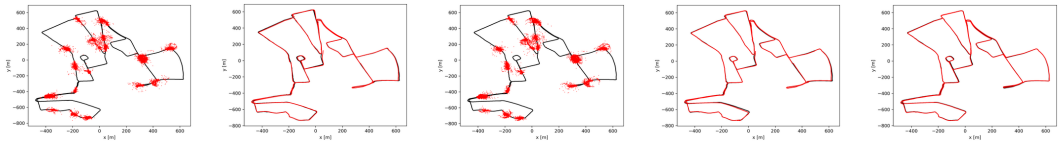
We evaluate LINK under conventional fully known relocalization, controlled partial coverage on public benchmarks, and route-level partial coverage in UrbanUnseen. The analysis then examines the contribution of reliability-aware switching, temporal reliability estimation, reward design, and the chosen backbone/odometry components.

5.1 Experimental Setup

Datasets, protocols, and metrics. Following Section 4, we evaluate LINK on Oxford [3, 21], NCLT [7], and UrbanUnseen under fully known, controlled partial-coverage, and route-level partial-coverage regimes. Known/unknown labels are used only for protocol construction and diagnostics, and are not provided to any method during inference. Unless otherwise stated, results are reported as translation / rotation error [m/°], where lower is better; **bold** and underlining denote the best and second-best values in each comparison column.

Baselines. We compare with representative map-free LiDAR relocalization methods: SGLoc [21], DiffLoc [22], LiSA [41], and LightLoc [23]. In partially known settings, we also evaluate stronger "+FG" variants, where each baseline is combined with the same voxelized GICP odometry [19] and factor-graph optimization [37].

Implementation details. Following the LightLoc setup, the scene-agnostic SCR encoder is frozen and only the regression head is retrained on the known-scene subset using AdamW with a one-cycle learning-rate schedule from 5×10^{-4} to 5×10^{-3} for 50 epochs and a batch size of 256. The SCR voxel size is 0.25 for Oxford, 0.3 for NCLT, and 0.5 for UrbanUnseen; odometry is estimated with voxelized GICP using a voxel size of 0.7. The reliability estimator uses a temporal window of $T = 10$ and three shared Mamba blocks with widths $\{128, 64, 32\}$. The policy uses ten actions, including direct SCR localization and propagation from candidate reference poses, and is optimized with PPO. All experiments are conducted on two NVIDIA RTX 4090 GPUs. Additional details and the PPO objective are provided in Appendix B.



(a) LiSA (177 m, 42.3°) (b) LiSA† (7.2 m, 4.2°) (c) LightLoc (175 m, 49.5°) (d) LightLoc† (7.7 m, 4.0°) (e) **LINK** (2.7 m, 3.7°)

Figure 4: Oxford trajectory comparison on 17-13-26-39 under random partial coverage. † denotes factor-graph optimization; black and red lines denote ground truth and predictions.

Table 3: Results on UrbanUnseen under fully known and route-level partial-coverage protocols.

Method	**Fully Known**	**Approx. 50% Scenes Known**	
	09-29 [m/°]↓	**10-23 [m/°]↓**	**10-24 [m/°]↓**
SGLoc+FG (CVPR'23) [21]	8.15/5.72	14.53/9.42	14.50/8.36
DiffLoc+FG (CVPR'24) [22]	7.76/5.51	12.13/7.69	12.61/7.77
LiSA+FG (CVPR'24) [41]	7.38/5.29	13.97/7.32	12.88/7.98
LightLoc+FG (CVPR'25) [23]	5.85/4.64	10.43/6.94	9.58/7.17
LINK	**5.47/4.23**	**7.24/6.15**	**7.52/6.42**

Table 4: Policy ablation on Oxford under random partial coverage.

Policy	**Acc. [%]↑**	**Recall [%]↑**	**F1 [%]↑**	**Average [m]↓**
Threshold-based [49]	39.5	22.6	36.9	6.72
Rule-based [26]	62.5	52.0	68.4	4.38
MLP-based [9]	72.6	84.4	83.1	3.46
RL-based	**94.3**	**98.8**	**96.3**	**2.24**

Table 5: Temporal pose-reliability estimator ablation on NCLT under random partial coverage.

Variant	**Temp.**	**Mamba**	**Cross-Attn.**	**Error [m/°]↓**
Temporal only	✓			5.56/5.29
+ Mamba	✓	✓		4.28/4.37
+ Cross-attn.	✓		✓	4.40/4.33
Full	✓	✓	✓	**2.39/3.03**

5.2 Main Results and Analysis

Results on Oxford. Oxford provides a controlled setting in which most test frames lie outside the retained training support. In the fully known protocol, LINK remains comparable to the leading relocalization baselines. Under 20.8% known-scene coverage, the lowest-error +FG baseline averages 6.94 m / 4.00°, whereas LINK averages 2.24 m / 3.64° in Table 1. The trajectory in Figure 4 helps explain this gap: factor-graph optimization reduces local discontinuities, but it cannot recover global consistency once the underlying SCR hypotheses move outside the covered scene distribution. Under the manual protocol, where known regions are concentrated around intersections, LINK averages 1.37 m / 2.48°; full per-sequence results are provided in Appendix C.1.

Results on NCLT. NCLT tests the same controlled partial-coverage question on a campus-scale route with different geometry and data collection conditions. The fully known results remain competitive, with LINK giving the best translation error on three of the four test sequences in Table 2. Under 25.7% known-scene coverage, the average error decreases from 4.29 m / 4.93° for LightLoc+FG to 2.39 m / 3.03° for LINK. This consistency with Oxford supports the same interpretation: once direct SCR predictions leave the covered scene distribution, post-hoc smoothing is less effective than selecting when to use regression and when to propagate from reliable references. Under the manual protocol, LINK averages 3.19 m / 3.63° on NCLT, with full per-sequence results in Appendix C.2.

Results on UrbanUnseen. Unlike the controlled Oxford and NCLT protocols, UrbanUnseen evaluates natural route-level known-to-unknown transitions. Absolute errors are larger than on Oxford and NCLT, but the same failure mode appears: stabilized SCR baselines still drift when the route contains genuinely unseen segments, whereas LINK improves over LightLoc+FG on both partially known routes and also gives the lowest error on the fully known 09-29 route in Table 3. The remaining errors on 10-23 and 10-24 make clear that route-level partial-coverage relocalization is harder than the controlled protocols, but the relative gap suggests that reliability-guided switching is not merely exploiting a favorable random repartition. Full trajectory-level diagnostics on UrbanUnseen are provided in Appendix D.1, and additional confidence visualizations are provided in Appendix D.2.

Table 6: Policy reward ablation on Oxford under random partial coverage.

Reward	Trans.	Rot.	Penalty	Error [m/°]↓
Translation only	✓			3.11/4.06
+ Rotation	✓	✓		2.71/3.85
+ Odometry penalty	✓		✓	2.55/3.80
Full reward	✓	✓	✓	**2.24/3.64**

Table 7: Odometry ablation under partial scene coverage. We report the average error [m/°] ↓.

Method	Dataset			Real time
	Oxford↓	NCLT↓	UrbanUnseen↓	
LINK (PIN-SLAM [28])	2.88/4.26	3.71/4.50	10.69/7.35	×
LINK (Fast-LIO [39])	2.56/3.88	3.24/3.97	9.35/6.85	✓
LINK (GICP [19])	**2.24/3.64**	**2.39/3.03**	**7.38/6.28**	✓

Table 8: Backbone ablation under fully known and partially known coverage. Cells report average translation / rotation error [m/°]; lower is better.

Dataset	Protocol	LINK w/ SGLoc [21]	LINK w/ DiffLoc [22]	LINK w/ LiSA [41]	LINK w/ LightLoc [23]
Oxford	**Fully Known**	1.31/1.38	1.65/**0.83**	0.88/1.03	**0.72**/1.10
	20.8% Known	2.70/3.89	2.55/3.70	2.37/3.81	**2.24/3.64**
NCLT	**Fully Known**	1.62/3.29	**1.14/2.10**	1.43/2.31	1.27/2.37
	25.7% Known	4.20/5.55	3.92/4.68	3.97/4.30	**2.39/3.03**
UrbanUnseen	**Fully Known**	7.48/5.37	6.55/4.58	6.72/4.66	**5.47/4.23**
	Approx. 50% Known	13.62/8.60	12.13/7.54	11.39/7.77	**7.38/6.28**

Table 9: Inference runtime and model-size comparison on Oxford and NCLT.

Method	Reliability [ms]↓	Policy [ms]↓	Runtime [ms]↓		GPU↓	Model↓
			Oxford	NCLT		
SGLoc+FG [21]	–	–	38	75	2.4 GB	116 MB
LiSA+FG [41]	–	–	38	75	2.4 GB	116 MB
LightLoc+FG [23]	–	–	29	45	2.1 GB	70 MB
LINK	8	11	48	64	2.5 GB	86 MB

5.3 Ablation and Deployment Analysis

Ablation of reliability-aware decision policy. To evaluate whether LINK benefits from sequential mode selection rather than frame-wise confidence rejection, we compare the RL policy with threshold-based [49], rule-based [26], and MLP-based [9] alternatives. As shown in Table 4, the RL policy achieves the highest oracle-action agreement and reduces the Oxford translation error from 3.46 m for the strongest non-RL policy to 2.24 m. This indicates that using temporal context and reference-pose availability is more effective than treating confidence as an isolated per-frame signal.

Ablation of temporal reliability estimator. To examine what information is needed for robust switching, we ablate the temporal pose-reliability estimator. As shown in Table 5, the full estimator achieves the lowest NCLT error when temporal modeling and pose–feature cross-attention are used together. This result suggests that reliability estimation benefits from modeling how pose hypotheses and scene features evolve jointly over time.

Ablation of policy reward design. We further ablate the policy reward to test whether the policy should optimize long-horizon pose consistency rather than instantaneous translation alone. As shown in Table 6, adding rotation and odometric-span penalties to translation supervision reduces the Oxford error from 3.11 m / 4.06° to 2.24 m / 3.64°. These terms help penalize decisions that are locally plausible but drift-prone under long odometric propagation.

Ablation of odometry source. To separate the switching mechanism from the relative-motion source, we evaluate different odometry modules. As shown in Table 7, GICP provides the best accuracy–runtime trade-off among PIN-SLAM [28], Fast-LIO [39], and voxelized GICP [19] across Oxford, NCLT, and UrbanUnseen. We therefore use GICP as the default odometry source in LINK.

Ablation of SCR backbone. We also evaluate whether the mode-selection framework depends on a particular SCR backbone. As shown in Table 8, LINK transfers across SGLoc, DiffLoc, LiSA, and LightLoc backbones, while LightLoc gives the strongest partial-coverage results among the tested variants. We therefore use LightLoc as the default SCR backbone.

Deployment cost. Finally, we compare deployment cost using runtime and model-size statistics on Oxford and NCLT. As shown in Table 9, the temporal pose-reliability estimator and learned decision policy add 8 ms and 11 ms, respectively, while LINK preserves real-time inference on both datasets.

Since LINK does not introduce explicit map storage, its deployment cost mainly comes from learned reliability estimation and policy inference rather than a return to map-based retrieval.

6 Conclusion

We study map-free LiDAR relocalization under partial scene coverage, where test routes may deviate from training data. We propose LINK, a confidence-guided hybrid framework that evaluates pose reliability from SCR sequences to adaptively switch between direct localization and odometric propagation. To benchmark this, we introduce UrbanUnseen and controlled protocols on Oxford/NCLT datasets. Results demonstrate that LINK significantly enhances robustness in unseen areas while maintaining competitive accuracy in known scenes, highlighting the value of reliability-aware mode selection for practical deployment.

Acknowledgements

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

References

[1] Sheng Ao, Qingyong Hu, Bo Yang, Andrew Markham, and Yulan Guo. SpinNet: Learning a general surface descriptor for 3D point cloud registration. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 11753–11762, June 2021.

[2] Sheng Ao, Qingyong Hu, Hanyun Wang, Kai Xu, and Yulan Guo. Buffer: Balancing accuracy, efficiency, and generalizability in point cloud registration. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 1255–1264, 2023.

[3] Dan Barnes, Matthew Gadd, Paul Murcutt, Paul Newman, and Ingmar Posner. The Oxford radar robotcar dataset: A radar extension to the Oxford robotcar dataset. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 6433–6438, 2020.

[4] Eric Brachmann and Carsten Rother. Visual camera re-localization from RGB and RGB-D images using DSAC. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(9):5847–5865, 2021.

[5] Eric Brachmann, Tommaso Cavallari, and Victor Adrian Prisacariu. Accelerated coordinate encoding: Learning to relocalize in minutes using RGB and poses. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 5044–5053, 2023.

[6] Keenan Burnett, David J Yoon, Yuchen Wu, Andrew Z Li, Haowei Zhang, Shichen Lu, Jingxing Qian, Wei-Kang Tseng, Andrew Lambert, Keith YK Leung, et al. Boreas: A multi-season autonomous driving dataset. *Int. J. Robot. Res.*, 42(1-2):33–42, 2023.

[7] Nicholas Carlevaris-Bianco, Arash K Ushani, and Ryan M Eustice. University of Michigan north campus long-term vision and LiDAR dataset. *Int. J. Robot. Res.*, 35(9):1023–1035, 2016.

[8] Xieyuanli Chen, Ignacio Vizzo, Thomas Läbe, Jens Behley, and Cyrill Stachniss. Range image-based LiDAR localization for autonomous vehicles. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 5802–5808, 2021.

[9] Owen Claxton, Connor Malone, Helen Carson, Jason J. Ford, Gabe Bolton, Iman Shames, and Michael Milford. Improving Visual Place Recognition based robot navigation by verifying localization estimates. *IEEE Robot. Autom. Lett.*, 9(12):11098–11105, 2024. doi: 10.1109/LRA.2024.3483045.

[10] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.*, 32(11):1231–1237, 2013.

[11] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

[12] Peiyu Guan, Zhiqiang Cao, Junzhi Yu, Chao Zhou, and Min Tan. Scene coordinate regression network with global context-guided spatial feature transformation for visual relocalization. *IEEE Robot. Autom. Lett.*, 6(3):5737–5744, 2021.

[13] Renlang Huang, Yufan Tang, Jiming Chen, and Liang Li. A consistency-aware spot-guided transformer for versatile and hierarchical point cloud registration. In *Adv. Neural Inf. Process. Syst.*, 2024.

[14] Alex Kendall and Roberto Cipolla. Modelling uncertainty in deep learning for camera relocalization. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 4762–4769, 2016.

[15] Alex Kendall, Matthew Grimes, and Roberto Cipolla. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pages 2938–2946, 2015.

[16] Giseop Kim, Yeong Sang Park, Younghun Cho, Jinyong Jeong, and Ayoung Kim. MulRan: Multimodal range dataset for urban place recognition. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 6246–6253, 2020.

[17] Giseop Kim, Sunwook Choi, and Ayoung Kim. Scan Context++: Structural place recognition robust to rotation and lateral variations in urban environments. *IEEE Trans. Robot.*, 38(3): 1856–1874, 2021.

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] Kenji Koide, Masashi Yokozuka, Shuji Oishi, and Atsuhiko Banno. Voxelized GICP for fast and accurate 3D point cloud registration. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 11054–11059, 2021.

[20] Jacek Komorowski. MinkLoc3D: Point cloud based large-scale place recognition. In *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, pages 1790–1799, 2021.

[21] Wen Li, Shangshu Yu, Cheng Wang, Guosheng Hu, Siqi Shen, and Chenglu Wen. SGLoc: Scene geometry encoding for outdoor LiDAR localization. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 9286–9295, 2023.

[22] Wen Li, Yuyang Yang, Shangshu Yu, Guosheng Hu, Chenglu Wen, Ming Cheng, and Cheng Wang. DiffLoc: Diffusion model for outdoor LiDAR localization. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 15045–15054, 2024.

[23] Wen Li, Chen Liu, Shangshu Yu, Dunqiang Liu, Yin Zhou, Siqi Shen, Chenglu Wen, and Cheng Wang. LightLoc: Learning outdoor LiDAR localization at light speed. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 6680–6689, 2025.

[24] Zijun Li, Zhipeng Cai, Bochun Yang, Xuelun Shen, Siqi Shen, Xiaoliang Fan, Michael Paulitsch, and Cheng Wang. ConDo: Continual domain expansion for absolute pose regression. In *Proc. AAAI Conf. Artif. Intell.*, volume 39, pages 14628–14636, 2025.

[25] Lun Luo, Si-Yuan Cao, Xiaorui Li, Jintao Xu, Rui Ai, Zhu Yu, and Xieyuanli Chen. BEV-Place++: Fast, robust, and lightweight LiDAR global localization for autonomous ground vehicles. *IEEE Trans. Robot.*, 41:4479–4498, 2025. doi: 10.1109/TRO.2025.3585385.

[26] Xubo Luo, Xue Wan, Yixing Gao, Yaolin Tian, Wei Zhang, and Leizheng Shu. JointLoc: A real-time visual localization framework for planetary UAVs based on joint relative and absolute pose estimation. In *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pages 3348–3355, 2024. doi: 10.1109/IROS58592.2024.10802040.

[27] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. ShuffleNet V2: Practical guidelines for efficient CNN architecture design. In *Proc. Eur. Conf. Comput. Vis.*, pages 116–131, 2018.

[28] Yue Pan, Xingguang Zhong, Louis Wiesmann, Thorbjörn Posewsky, Jens Behley, and Cyrill Stachniss. PIN-SLAM: LiDAR SLAM using a point-based implicit neural representation for achieving global map consistency. *IEEE Trans. Robot.*, 40:4045–4064, 2024.

[29] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. GeoTransformer: Geometric transformer for fast and robust point cloud registration. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 11143–11152, 2022.

[30] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of CNN-based absolute camera pose regression. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 3302–3312, 2019.

[31] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[32] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in RGB-D images. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 2930–2937, 2013.

[33] Mikaela Angelina Uy and Gim Hee Lee. PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 4470–4479, 2018.

[34] Kavisha Vidanapathirana, Peyman Moghadam, Sridha Sridharan, and Clinton Fookes. Spectral geometric verification: Re-ranking point cloud retrieval for metric localization. *IEEE Robot. Autom. Lett.*, 8(5):2494–2501, 2023.

[35] Sijie Wang, Qiyu Kang, Rui She, Wei Wang, Kai Zhao, Yang Song, and Wee Peng Tay. HypLiLoc: Towards effective LiDAR pose regression with hyperbolic fusion. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 5176–5185, 2023.

[36] Wei Wang, Bing Wang, Peijun Zhao, Changhao Chen, Ronald Clark, Bo Yang, Andrew Markham, and Niki Trigoni. PointLoc: Deep pose regressor for LiDAR point cloud localization. *IEEE Sens. J.*, 22(1):959–968, 2021.

[37] Weisong Wen and Li-Ta Hsu. Towards robust GNSS positioning and real-time kinematic using factor graph optimization. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 5884–5890, 2021.

[38] Yan Xia, Yusheng Xu, Shuang Li, Rui Wang, Juan Du, Daniel Cremers, and Uwe Stilla. SOE-Net: A self-attention and orientation encoding network for point cloud based place recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 11348–11357, 2021.

[39] Wei Xu and Fu Zhang. Fast-LIO: A fast, robust LiDAR-inertial odometry package by tightly-coupled iterated Kalman filter. *IEEE Robot. Autom. Lett.*, 6(2):3317–3324, 2021.

[40] Xuecheng Xu, Sha Lu, Jun Wu, Haojian Lu, Qiuguo Zhu, Yiyi Liao, Rong Xiong, and Yue Wang. RING++: Roto-translation-invariant gram for global localization on a sparse scan map. *IEEE Trans. Robot.*, 39(6):4616–4635, 2023.

[41] Bochun Yang, Zijun Li, Wen Li, Zhipeng Cai, Chenglu Wen, Yu Zang, Matthias Muller, and Cheng Wang. LiSA: LiDAR localization with semantic awareness. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 15271–15280, 2024.

[42] Heng Yang, Jingnan Shi, and Luca Carlone. TEASER: Fast and certifiable point cloud registration. *IEEE Trans. Robot.*, 37(2):314–333, 2020.

[43] Yuyang Yang, Wen Li, Sheng Ao, Qingshan Xu, Shangshu Yu, Yu Guo, Yin Zhou, Siqi Shen, and Cheng Wang. RALoc: Enhancing outdoor LiDAR localization via rotation awareness. In *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, pages 3304–3313, 2025.

[44] Huan Yin, Xuecheng Xu, Sha Lu, Xieyuanli Chen, Rong Xiong, Shaojie Shen, Cyrill Stachniss, and Yue Wang. A survey on global LiDAR localization: Challenges, advances and open problems. *Int. J. Comput. Vis.*, 132(8):3139–3171, 2024.

[45] Shangshu Yu, Cheng Wang, Yitai Lin, Chenglu Wen, Ming Cheng, and Guosheng Hu. STCLoc: Deep LiDAR localization with spatio-temporal constraints. *IEEE Trans. Intell. Transp. Syst.*, 24 (1):489–500, 2022.

[46] Shangshu Yu, Cheng Wang, Chenglu Wen, Ming Cheng, Minghao Liu, Zhihong Zhang, and Xin Li. LiDAR-based localization using universal encoding and memory-aware regression. *Pattern Recognit.*, 128:108685, 2022.

[47] Shangshu Yu, Xiaotian Sun, Wen Li, Chenglu Wen, Yunuo Yang, Bailu Si, Guosheng Hu, and Cheng Wang. NIDALoc: Neurobiologically inspired deep LiDAR localization. *IEEE Trans. Intell. Transp. Syst.*, 25(5):4278–4289, 2023.

[48] Chongjian Yuan, Jiarong Lin, Zheng Liu, Hairuo Wei, Xiaoping Hong, and Fu Zhang. BTC: A binary and triangle combined descriptor for 3D place recognition. *IEEE Trans. Robot.*, 2024.

[49] Mubariz Zaffar, Liangliang Nan, and Julian F. P. Kooij. On the estimation of Image-Matching uncertainty in Visual Place Recognition. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 17743–17753, 2024. doi: 10.1109/CVPR52733.2024.01680.

[50] Tongzhou Zhang, Gang Wang, Yu Chen, Hai Zhang, and Jue Hu. Multi-constellation-inspired single-shot global LiDAR localization. In *Proc. AAAI Conf. Artif. Intell.*, volume 38, pages 10404–10412, 2024.

[51] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 2921–2929, 2016.

[52] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(4):4396–4415, 2022.

[53] Lei Zhou, Zixin Luo, Tianwei Shen, Jiahui Zhang, Mingmin Zhen, Yao Yao, Tian Fang, and Long Quan. KFNet: Learning temporal camera relocalization using Kalman filtering. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, pages 4919–4928, 2020.

Appendix

Table of Contents

A Benchmark and Protocol Details 15
- A.1 Oxford Ground-Truth Pose Alignment 15
- A.2 UrbanUnseen Acquisition and Splits 15
- A.3 Controlled Partial-Coverage Protocols on Oxford and NCLT 16
- A.4 Partition Visualizations 17

B Implementation Details 17
- B.1 SCR Backbone and Training 17
- B.2 Temporal Reliability Estimator 18
- B.3 Localization Policy and Training Objectives 18
- B.4 Odometry and Inference-Time State 19

C Additional Quantitative Results 19
- C.1 Manual Protocol on Oxford 19
- C.2 Manual Protocol on NCLT 19
- C.3 Runtime and Storage Analysis 19

D Qualitative Diagnostics 20
- D.1 Trajectory-Level Comparisons 20
- D.2 Confidence Visualizations 20

E Limitations 23

A Benchmark and Protocol Details

A.1 Oxford Ground-Truth Pose Alignment

For Oxford [3], we follow SGLoc [21] and use trajectory-aligned ground-truth poses for evaluation. The original repeated-route poses can contain noticeable discrepancies at the same road locations, which can confound frame-level localization errors when repeated traversals are compared directly. The aligned poses reduce this discrepancy while preserving the standard Oxford sequence split used by prior map-free LiDAR relocalization work. All Oxford experiments in this paper, including baselines and LINK, are evaluated under the same aligned poses.

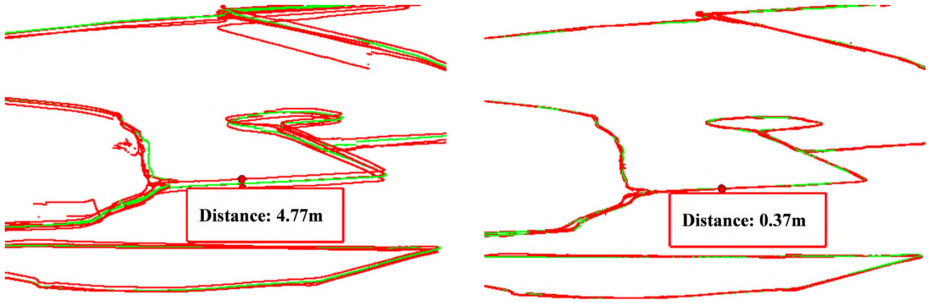


Figure 5: Oxford ground-truth pose alignment. The left panel shows repeated-route pose discrepancy in the original Oxford poses, while the right panel shows the trajectory-aligned poses used in our Oxford experiments. Green denotes the reference trajectory, and red denotes the corresponding trajectory before or after alignment. In this example, the local discrepancy decreases from 4.77 m to 0.37 m.

A.2 UrbanUnseen Acquisition and Splits

Figure 6 summarizes the UrbanUnseen acquisition platform and route coverage. The dataset was collected with a multi-sensor vehicle equipped with an Ouster OS1-128 LiDAR, HIKROBOT cameras, Robosense RubyPlus-128 and M1-64 LiDARs, and an FDI-Sigma navigation system. The route map illustrates the full-coverage trajectories used for training and fully known evaluation, together with the two partial-coverage trajectories used to evaluate route-level known/unknown transitions.

Table 10 compares UrbanUnseen with representative outdoor LiDAR localization datasets. The native unknown-route evaluation column refers to whether the dataset protocol includes real test-route segments outside the training-covered route support, rather than whether such segments can be simulated by repartitioning an existing repeated-route benchmark.

Table 10: Comparison with representative outdoor LiDAR localization datasets. Route length and coverage area follow the representative benchmark-scale statistics reported by the corresponding dataset papers. Native unknown-route evaluation indicates whether the dataset includes real test-route segments absent from the training-covered route support in its intended evaluation protocol.

Dataset	Rep. Route Length	Coverage Area	Native Unknown-Route Eval.	Repeated Trajectories	RGB Camera
KITTI [10]	4 km	–	×	×	✓
Boreas [6]	8 km	6 km^2	×	✓	✓
BiLiLo [43]	10 km	3 km^2	×	✓	×
NCLT [7]	5.5 km	0.5 km^2	×	✓	✓
Oxford [3]	10 km	2.5 km^2	×	✓	✓
UrbanUnseen	**45 km**	**23** km^2	✓	✓	✓

UrbanUnseen contains seven trajectories: 09-11, 09-29, 10-16, 10-17, 10-22, 10-23, and 10-24. Table 11 lists the trajectory length, average vehicle speed, Ouster point-cloud count, and sensor availability for each trajectory. The Robosense RubyPlus-128 and Robosense M1-64 streams are unavailable for 09-29 due to equipment failure. All experiments in this paper use the Ouster OS1-128 LiDAR stream.

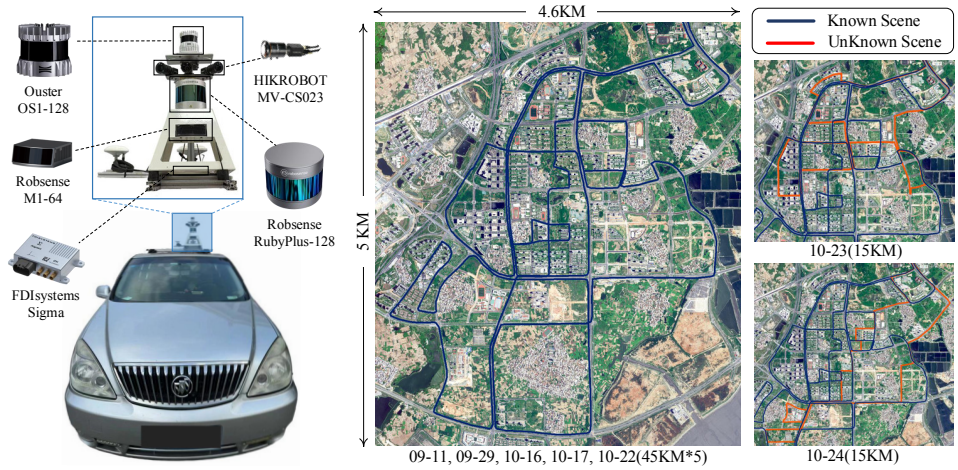


Figure 6: Overview of UrbanUnseen acquisition and route coverage. Left: collection vehicle and recorded sensor streams. Right: route coverage used in the UrbanUnseen split. The full-coverage routes are used for training and fully known evaluation, while trajectories 10-23 and 10-24 contain approximately 50% route segments outside the training coverage. Blue and red denote known and unknown route segments under the UrbanUnseen split.

Table 11: UrbanUnseen trajectory and sensor parameters. We report the trajectory length, average vehicle speed, and the number of point clouds collected by the Ouster OS1-128 stream. Sensor availability is indicated for each trajectory.

Trajectory	Ouster OS1-128			HIKROBOT camera (10 Hz)	Robosense RubyPlus-128 (10 Hz)	Robosense M1-64 (10 Hz)	FDI-Sigma (100 Hz)
	Length (km)	Speed (km/h)	Point clouds (K)				
09-11	45	26	62	✓	✓	✓	✓
09-29	47	27	62	✓	×	×	✓
10-16	44	25	64	✓	✓	✓	✓
10-17	47	26	65	✓	✓	✓	✓
10-22	46	26	63	✓	✓	✓	✓
10-23	15	34	15	✓	✓	✓	✓
10-24	15	35	15	✓	✓	✓	✓

FDI-Sigma provides ground-truth poses at 100 Hz, while the LiDAR streams run at 10–20 Hz. We therefore obtain the ground-truth pose for each LiDAR frame by temporally aligning the LiDAR timestamp to the FDI-Sigma trajectory and interpolating between neighboring navigation poses.

The UrbanUnseen split follows the route-level protocol used in the main paper. Trajectories 09-11, 10-16, 10-17, and 10-22 are used for training. Trajectory 09-29 is used for fully known evaluation. Trajectories 10-23 and 10-24 are used for route-level partially known evaluation; each contains approximately 50% unknown scenes under this split.

A.3 Controlled Partial-Coverage Protocols on Oxford and NCLT

We use Oxford and NCLT to construct controlled partial-coverage protocols because they do not natively distinguish route segments covered during training from route segments absent from training. Known scenes and unknown scenes are used only as protocol labels for constructing the training subset and diagnostic reporting. These labels are not provided to LINK or to any baseline during inference.

For Oxford, the training sequences are 11-14-02-26, 14-12-05-52, 14-14-48-55, and 18-15-20-12, and the test sequences are 15-13-06-37, 17-13-26-39, 17-14-03-00, and 18-14-14-42. For NCLT, the training sequences are 2012-01-22, 2012-02-02, 2012-02-18, and 2012-05-11, and the test sequences are 2012-02-12, 2012-02-19, 2012-03-31, and 2012-05-26.

We use two controlled partition protocols. In the random protocol, world-coordinate centers are sampled on the ground-truth training trajectory, and scans within a 30 m radius of each center are retained as known scenes. In the manual protocol, road intersections are retained as known regions.

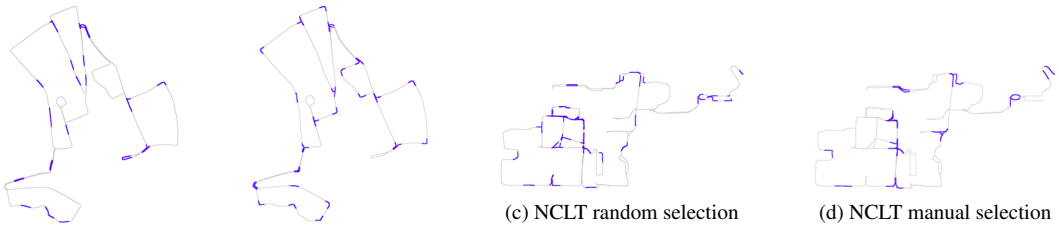
In both protocols, the initial route segment is treated as known by default to provide a valid globally referenced starting segment. Table 12 reports the sequence-level and average known-scene ratios.

Table 12: Controlled partition ratios for Oxford and NCLT. The table reports the proportion of training scans retained as known scenes under the random and manual protocols.

Method	dataset		Data				
Random Selection	Oxford	proportion of Known Scenes	**11-14-02-26**	**14-12-05-52**	**14-14-48-55**	**18-15-20-12**	**Average**
			20.9%	20.4%	21.5%	20.5%	20.8%
	NCLT		**2012-01-22**	**2012-02-02**	**2012-02-18**	**2012-05-11**	**Average**
			23.9%	30.7%	24.0%	24.2%	25.7%
Manual Selection	Oxford		**11-14-02-26**	**14-12-05-52**	**14-14-48-55**	**18-15-20-12**	**Average**
			36.7%	30.0%	34.2%	25.0%	31.4%
	NCLT		**2012-01-22**	**2012-02-02**	**2012-02-18**	**2012-05-11**	**Average**
			23.6%	22.2%	22.6%	20.0%	22.1%

A.4 Partition Visualizations

Figure 7 visualizes the known/unknown partitions for the two controlled protocols on Oxford and NCLT. In the random protocol, 25 centers are sampled for Oxford and 28 centers are sampled for NCLT. Because neighboring 30 m regions may overlap, the numbers of visibly separated known regions are 18 and 22, respectively. In the manual protocol, the retained known regions correspond to 23 key intersections on Oxford and 10 key intersections on NCLT.



(a) Oxford random selection

(b) Oxford manual selection

(c) NCLT random selection

(d) NCLT manual selection

Figure 7: Controlled partition visualization. The figure shows known/unknown partitions for Oxford and NCLT under the controlled partial-coverage protocols. Known scenes are shown in blue and unknown scenes in gray.

B Implementation Details

B.1 SCR Backbone and Training

LINK instantiates its SCR backbone with LightLoc [23]. The scene-agnostic encoder is kept frozen, and only the regression head is retrained on the known-scene subset of each target dataset. Under the partial-coverage protocols, the retraining set contains only the route segments designated as known by the corresponding protocol; segments designated as unknown are excluded from SCR retraining.

Training proceeds in three stages. The SCR regression head is trained first using Eq. (6). The temporal reliability estimator is then trained using pose hypotheses and intermediate features from the frozen SCR backbone. Finally, the localization policy is trained with the SCR backbone and reliability estimator fixed.

For the SCR regression head, the point cloud is encoded into a 512-dimensional scene descriptor following the LightLoc implementation. The descriptor is normalized to the unit sphere, augmented with Gaussian noise, normalized again, and concatenated before regression. The regression head is trained with a batch size of 256. The voxel size is set to 0.25 for Oxford, 0.3 for NCLT, and 0.5 for UrbanUnseen. The intermediate feature map produced by the frozen encoder is retained for temporal reliability estimation.

B.2 Temporal Reliability Estimator

The forward computation of the temporal reliability estimator follows Eqs. (2) and (3). We use a temporal window of $T = 10$ frames. Each temporal token concatenates the normalized 7-D SCR pose with the compressed feature map produced by $G(\cdot)$. The original temporal stream is paired with a shuffled perturbation stream that preserves frame order while perturbing the pose–feature tokens. The two streams are encoded by shared Mamba blocks [11], fused by cross-attention, temporally averaged, and mapped by a two-layer MLP with sigmoid activation.

Training uses the soft target and regression objective in Eqs. (7) and (8). The reliability estimator is trained with Adam [18] for 50 epochs using a batch size of 64, with the learning rate decayed from 1×10^{-4} to 1×10^{-6}. The resulting score is used as the pose-reliability signal for policy selection.

B.3 Localization Policy and Training Objectives

The main text defines the policy state, action space, action-conditioned pose rule, and reward in Eqs. (4), (1), and (5). This subsection records the implementation-level state representation and the PPO objective used to train the policy.

The reference-pose part of the policy state is a fixed-size candidate summary, $\boldsymbol{\psi}_t = [m_{1,t}, \delta_{1,t}, \ldots, m_{K,t}, \delta_{K,t}]$. For each candidate, $m_{k,t}$ indicates whether a valid historical reference pose is available, and $\delta_{k,t}$ is the normalized temporal span from that reference pose to the current frame. This summary is concatenated with the recent confidence history defined in Eq. (4). In the reported setting, the policy has ten actions: one direct-SCR action and nine reference-pose propagation actions. Ground-truth poses and rewards are used only during policy training; inference uses the learned policy and does not require ground-truth poses, rewards, or known/unknown labels.

For compact notation, let

$$\mathcal{R}_t = \mathcal{R}(\mathbf{s}_t, a_t), \tag{9}$$

where $\mathcal{R}(\mathbf{s}_t, a_t)$ follows the reward definition in Eq. (5). PPO [31] is used to train the policy for 200 episodes by maximizing the expected cumulative reward:

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}_t\right], \tag{10}$$

where $\pi_\theta(a_t|\mathbf{s}_t)$ is the policy parameterized by θ and γ is the discount factor.

The temporal-difference residual Δ_t^V and advantage $\hat{A}_t$ are estimated with generalized advantage estimation:

$$\begin{aligned} \Delta_t^V &= \mathcal{R}_t + \gamma V_\phi(\mathbf{s}_{t+1}) - V_\phi(\mathbf{s}_t), \\ \hat{A}_t &= \sum_{l=0}^{\infty} (\gamma\lambda)^l \Delta_{t+l}^V, \end{aligned} \tag{11}$$

where V_ϕ is the value function parameterized by ϕ and λ is the GAE parameter. The likelihood ratio is defined as

$$\rho_t(\theta) = \frac{\pi_\theta(a_t|\mathbf{s}_t)}{\pi_{\theta_{\text{old}}}(a_t|\mathbf{s}_t)}, \tag{12}$$

where θ_{old} denotes the fixed policy parameters before the PPO update. The clipped PPO loss minimized during training is

$$\mathcal{L}_{\text{PPO}}^{\text{clip}} = -\mathbb{E}_t\left[\min\left(\rho_t(\theta)\hat{A}_t, \text{clip}(\rho_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right], \tag{13}$$

where ϵ is the clipping parameter and $\mathbb{E}_t$ denotes empirical averaging over training timesteps. The value-function loss and entropy term are

$$\mathcal{L}_V = \mathbb{E}_t\left[\left(V_\phi(\mathbf{s}_t) - V_t^{\text{target}}\right)^2\right], \qquad \mathcal{L}_H = \mathbb{E}_t\left[\mathcal{H}(\pi_\theta(\cdot|\mathbf{s}_t))\right]. \tag{14}$$

Here, V_t^{target} is the value-regression target and $\mathcal{H}(\cdot)$ denotes policy entropy. The final decision-policy objective is

$$\mathcal{L}_{\text{DM}} = \mathcal{L}_{\text{PPO}}^{\text{clip}} + c_1\mathcal{L}_V - c_2\mathcal{L}_H, \tag{15}$$

where c_1 and c_2 weight value fitting and entropy regularization.

B.4 Odometry and Inference-Time State

The relative-motion term $\Delta\mathbf{p}_{\tau_k \to t}^{\mathrm{odom}}$ in Eq. (1) is estimated using GICP odometry. In the partially known experiments, the "+FG" variants of SCR baselines use the same GICP odometry source and factor-graph optimization. This setting assesses whether augmenting baseline SCR outputs with factor-graph optimization is sufficient to explain the improvement, while LINK performs learned frame-wise mode selection between direct SCR localization and odometric propagation during inference.

At inference time, LINK is map-free in the sense used throughout the paper: it does not store, retrieve, or query an explicit pre-built 3-D map database or map-feature database. The SCR backbone, reliability estimator, and policy are learned model parameters. The reference-pose memory and odometry estimates are runtime state used for pose propagation, not an explicit map database. Known/unknown protocol labels are not provided to LINK or to any baseline during inference. This map-free setting should therefore not be interpreted as stateless or zero-memory.

C Additional Quantitative Results

This appendix complements the main experiments with additional evidence that is too detailed for the main paper. The manual protocols stress sparse coverage around road intersections, while the runtime and storage comparison clarifies the deployment cost of adding reliability estimation and policy inference. All localization entries follow the metric convention in Section 5.1: translation / rotation error [m/°], lower is better.

C.1 Manual Protocol on Oxford

The Oxford manual protocol concentrates known regions around road intersections rather than distributing them uniformly along the route. The per-sequence results in Table 13 test whether the improvements in the random partial-coverage protocol persist under this nonuniform split.

Table 13: Manual-selection results on Oxford. The protocol uses 31.4% known-scene coverage for training and evaluates on the full test trajectories.

Protocol	Method	15-13-06-37↓	17-13-26-39↓	17-14-03-00↓	18-14-14-42↓	Average [m/°]↓
Only 31.4% Scenes Known	SGLoc [21]	75.48/34.03	78.03/31.67	74.42/32.34	78.11/33.99	76.51/33.01
	SGLoc+FG [21]	8.21/5.26	8.28/6.24	8.34/6.60	8.06/6.47	8.22/6.14
	DiffLoc [22]	60.23/27.02	59.13/26.74	58.23/27.05	61.91/28.70	59.88/27.38
	DiffLoc+FG [22]	7.21/4.90	6.13/4.00	7.21/5.13	6.86/5.40	6.85/4.86
	LiSA [41]	68.63/29.72	71.31/30.14	69.22/30.67	71.07/29.44	70.06/29.99
	LiSA+FG [41]	7.03/5.35	7.22/5.90	7.59/5.44	7.34/5.81	7.30/5.63
	LightLoc [23]	78.54/32.93	70.07/29.69	70.84/34.73	70.12/30.59	72.39/31.99
	LightLoc+FG [23]	7.15/4.42	7.22/3.91	6.65/4.44	5.67/4.38	6.67/4.29
	LINK	**1.46/2.57**	**1.02/1.97**	**1.27/2.58**	**1.71/2.79**	**1.37/2.48**

C.2 Manual Protocol on NCLT

The NCLT manual protocol provides the same sparse-coverage test on a campus-scale route. Table 14 gives the per-sequence breakdown used to support the main-paper manual-protocol summary.

C.3 Runtime and Storage Analysis

The main paper summarizes inference runtime and model size on Oxford and NCLT in Table 9, where Reliability and Policy denote the same LINK inference components reported here. Table 15 provides a complementary sequence-specific NCLT view that further separates model storage, explicit map storage, and localization error.

Table 14: Manual-selection results on NCLT. The protocol uses 22.1% known-scene coverage for training and evaluates on the full test trajectories.

Protocol	Method	2012-02-12↓	2012-02-19↓	2012-03-31↓	2012-05-26↓	Average [m/°]↓
Only 22.1% Scenes Known	SGLoc [21]	121.37/55.11	118.09/55.24	125.55/54.39	122.30/57.78	121.83/55.63
	SGLoc+FG [21]	7.17/7.23	7.76/7.38	6.94/6.82	7.39/7.40	7.32/7.21
	DiffLoc [22]	118.50/54.88	124.47/53.23	128.34/57.04	124.12/57.83	123.86/55.75
	DiffLoc+FG [22]	6.95/7.32	7.45/6.96	7.68/6.29	7.79/7.08	7.47/6.91
	LiSA [41]	123.41/58.32	127.06/60.33	119.90/59.35	121.61/57.72	123.00/58.93
	LiSA+FG [41]	6.91/6.72	5.94/6.40	6.70/6.94	6.73/6.31	6.57/6.59
	LightLoc [23]	120.11/56.40	123.68/58.72	132.70/58.58	127.55/58.19	126.01/57.97
	LightLoc+FG [23]	6.88/6.74	4.48/5.32	5.00/4.10	6.41/9.34	5.69/6.38
	LINK	**2.58/3.65**	**3.51/4.06**	**1.84/2.83**	**4.81/3.96**	**3.19/3.63**

Table 15: Runtime and storage comparison. The table reports runtime, memory, storage, and localization error on NCLT sequence 2012-05-26 under the random partially known protocol. Storage is reported as model storage plus explicit map storage; LINK uses 86 MB model storage and 0 MB explicit map storage.

Method	Reliability [ms]↓	Policy [ms]↓	GPU Mem.↓	Model↓	Map↓	Runtime↓	Error [m/°]↓
LightLoc+FG [23]	–	–	2.1 GB	70 MB	**0 MB**	**45** ms	6.36/5.69
BEVPlace++ [25]	–	–	1.9 GB	17 MB	841 MB	54 ms	11.83/4.64
LINK	8	11	2.5 GB	86 MB	**0 MB**	64 ms	**2.66/3.17**

D Qualitative Diagnostics

D.1 Trajectory-Level Comparisons

These trajectory diagnostics test whether the quantitative improvements correspond to route-level consistency rather than isolated frame-wise improvements. Figure 8 compares Oxford trajectories under the manual protocol, where known scenes are concentrated around road intersections.

The Oxford trajectories diagnose route-level consistency under sparse manual coverage. Direct LightLoc produces visible off-route errors under the manual protocol. LightLoc+FG reduces some drift, but does not fully remove route-level errors. LINK keeps the predicted trajectory closer to the ground truth over the full route, indicating that the improvement is not only an average-error effect.

UrbanUnseen provides a complementary diagnostic under route-level known/unknown transitions. Figure 9 includes both direct SCR errors and +FG variants so that the effect of LINK can be compared against raw and post-processed baselines.

The UrbanUnseen trajectories show that LINK's behavior extends from controlled public-benchmark partitions to route-level known/unknown transitions. This visualization is diagnostic rather than a substitute for the quantitative UrbanUnseen table, and the larger deviations on 10-23 and 10-24 indicate that route-level partial-coverage relocalization remains more difficult than controlled public-benchmark partitions.

The route-level plots above summarize complete trajectories. Figure 10 provides a complementary local view on the fully known UrbanUnseen 09-29 route, where the selected segment is nominally covered by training data. The example illustrates that local SCR instability can still appear in covered routes, and that LINK can correct such local drift through reliability-guided switching. This figure should be interpreted as a qualitative diagnostic rather than a separate quantitative benchmark.

D.2 Confidence Visualizations

The confidence diagnostics examine whether the reliability estimator suppresses unreliable absolute SCR hypotheses before policy selection. On NCLT, Figure 11 compares raw SCR hypotheses with the high-confidence predictions retained by LINK under the random partially known protocol.

On UrbanUnseen, Figure 12 applies the same diagnostic to route-level transitions.

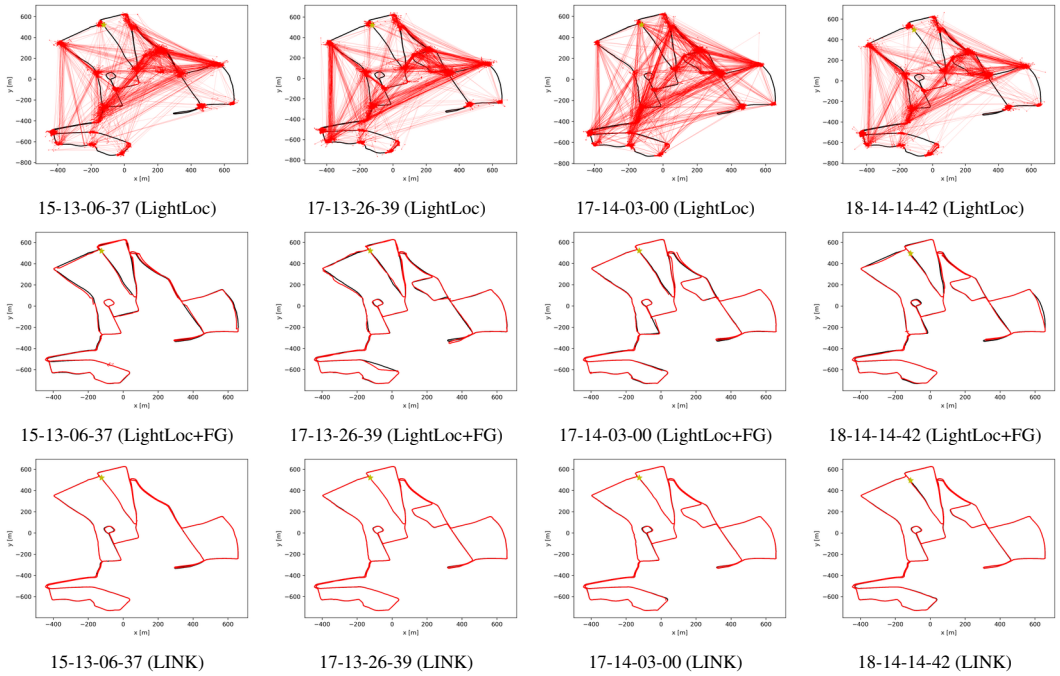


Figure 8: Oxford manual-coverage trajectory comparison. The figure shows trajectory-level results under the manual-selection protocol, where the known-scene coverage is 31.4%. Columns correspond to test sequences 15-13-06-37, 17-13-26-39, 17-14-03-00, and 18-14-14-42. Rows correspond to LightLoc, LightLoc+FG, and LINK, respectively. Black denotes the ground-truth trajectory, and red denotes the predicted trajectory.

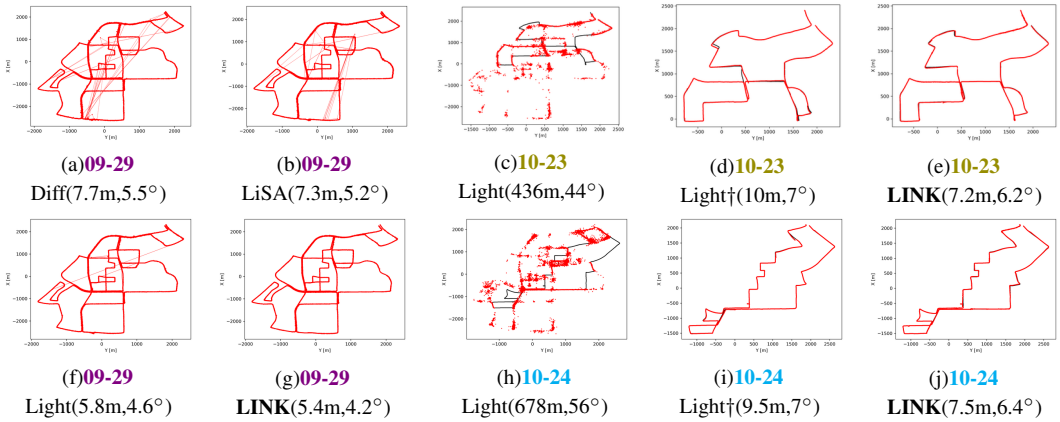
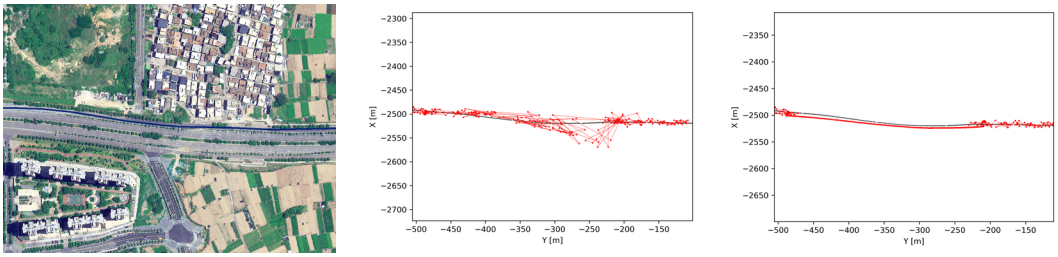


Figure 9: UrbanUnseen trajectory comparison. LINK is evaluated on trajectories **09-29** (fully known), **10-23** (approximately 50% scenes known), and **10-24** (approximately 50% scenes known). Ground truth and predictions are shown as black and red lines, respectively. † denotes factor-graph optimization.



(a) Remote-sensing view (b) 09-29 LightLoc (c) 09-29 LINK

Figure 10: UrbanUnseen local alignment detail on the fully known 09-29 route. Panel (a) shows the corresponding remote-sensing view, while panels (b) and (c) compare LightLoc and LINK on the selected road segment. This zoom-in complements the trajectory-level results by showing a local case where LightLoc exhibits drift and LINK remains better aligned.

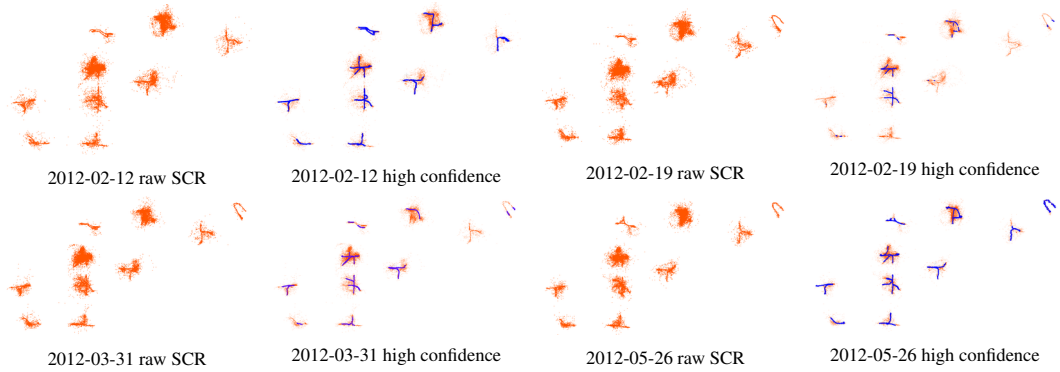


Figure 11: Confidence visualization on NCLT. Red denotes raw SCR hypotheses, and blue denotes high-confidence predictions retained by LINK. The figure diagnoses confidence behavior in complex scenes: confidence filtering suppresses many unreliable hypotheses, while some route segments may still receive low confidence because reliability is estimated as a soft signal rather than a perfect scene-membership label.

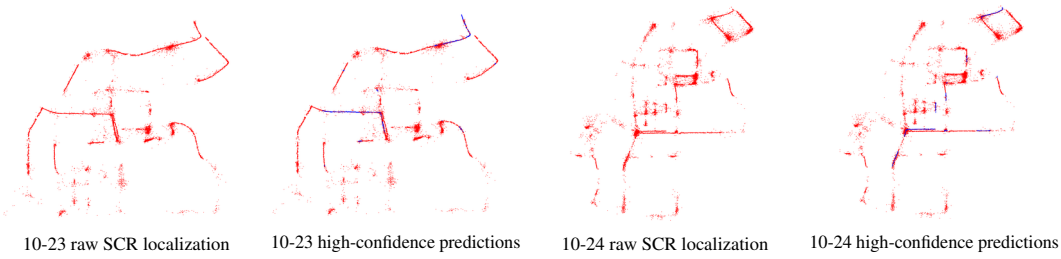


Figure 12: Confidence visualization on UrbanUnseen. Red denotes raw SCR hypotheses, and blue denotes high-confidence predictions retained by LINK. The diagnostic function of this figure is to show that raw SCR hypotheses can include numerically plausible but incorrect absolute predictions in unknown route segments, and confidence filtering suppresses many of these unreliable hypotheses before policy selection.

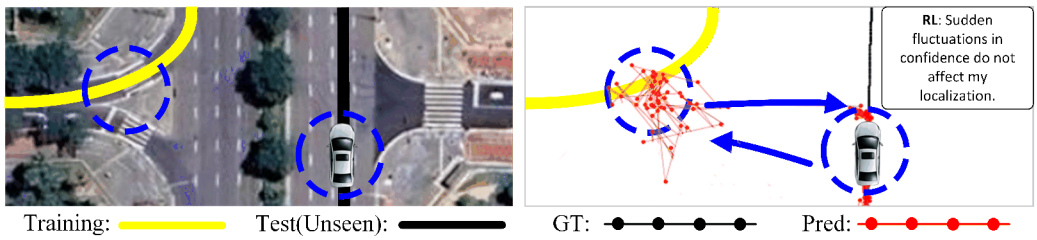


Figure 13: Ambiguous known/unknown passage. The yellow band denotes the training-covered route support, red points denote LINK predictions, and the black trajectory denotes the unseen test route. Brief passages near known support may produce confidence fluctuations, but the policy can still maintain a consistent globally referenced trajectory by selecting between direct SCR hypotheses and propagation from reliable historical poses.

The confidence maps should be interpreted as a soft pose-reliability signal, not a hard known/unknown classifier. In complex regions, some known segments may still receive low confidence, and brief passages near known route support may produce ambiguous confidence patterns, as illustrated in Figure 13. These ambiguities do not by themselves imply localization failure: confidence supports the decision policy by helping it avoid unreliable direct SCR predictions and by enabling propagation from a historical reference pose when the current absolute hypothesis is unreliable.

E Limitations

LINK targets outdoor LiDAR relocalization under partial route coverage, and the claims in this paper are bounded by the Oxford, NCLT, and UrbanUnseen settings evaluated in Sections 4 and 5. Two operating assumptions matter in this setting: the system benefits from occasional reliable absolute SCR hypotheses to keep historical reference poses globally anchored, and relative odometry should not be asked to bridge arbitrarily long intervals without trustworthy absolute updates. The policy is designed around these assumptions by using confidence history, reference-pose availability, and an odometric-span penalty. The confidence score should also be read as a soft pose-reliability signal rather than a hard known/unknown scene label; ambiguous passages may produce confidence fluctuations, but these cases can still be handled by policy selection, as shown in Appendix D.2. These boundaries are consistent with the paper's scope, while broader sensor setups or deployment conditions beyond the evaluated outdoor LiDAR relocalization protocols require separate validation.

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 [N/A] 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 [N/A] 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 E, we discuss the limitations of our work.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: Our work does not include theoretical results.

Guidelines:

- The answer [N/A] 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 [N/A] 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: [No]

Justification: The code is not yet publicly available and is being prepared for public release.

Guidelines:

- The answer [N/A] 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 5.1 and Appendix B.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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 5.1 and 5.3, with complementary storage details in Appendix C.3.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: [N/A]

    Justification: There is no societal impact of this work.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] 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 [N/A] 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: [No]

    Justification: UrbanUnseen is introduced and documented in the paper and is being prepared for public release.

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

Guidelines:

- The answer [N/A] 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.