Deep LiDAR Localization With Spatio-Temporal Constraints

Chen Liu 23020231154150[1], Yuyang Yang 23020231154247[1], Shujun Huang 23020231154140[1]
Yu Guo 23020231154184[2], Yongshu Huang 23020231154192[2]

[1]AI class
[2]Information class

Abstract

LiDAR localization plays a crucial role in enabling autonomous vehicles and robotics. Absolute pose regression, which directly estimates the mapping from a scene to a 6-DoF pose, has shown impressive results in learning-based localization.However, existing regression networks often face challenges in dealing with scene ambiguities, particularly in complex traffic environments, resulting in significant errors and limited practical applications. To overcome these limitations, we propose a novel LiDAR localization framework, which incorporates spatio-temporal constraints.The integration of spatio-temporal constraints enables the network to capture more context and dependencies between consecutive frames, leading to improved localization performance. We modified STLoc, replaced the feature extraction layer of STCLoc with RandLA-Net, and fine-tuned other parts of the model to improve the positioning accuracy of the model. Our source codes have released on https://github.com/the-full/RandLALoc

Introduction

LiDAR-based localization, is the problem of estimating the position and orientation from the LiDAR point cloud. In 3D computer vision, many applications such as autonomous driving (Lu et al. 2019), augmented reality (Dai et al. 2022) and robot navigation (Zou et al. 2022) are working in a highly dynamic environment.In general, LiDAR localization aims to estimate the 6-degrees-of-Freedom(6-DoF) pose of the system from point cloud data without the need of any external sensors like GPS.

A common solution is map-based localization (Yin et al. 2020), (Yu et al. 2021) divide the problem into two step, mapping and point cloud alignment. However, the cost of data collection and storage is huge. Recently, Absolute Pose Regression (APR) (Kendall, Grimes, and Cipolla 2015), (Kendall and Cipolla 2017), (Radwan, Valada, and Burgard 2018), (Xue et al. 2019), (Wang et al. 2020), (Shavit, Ferens, and Keller 2021) has convolutional neural networks to directly estimate 6-DoF pose from scene data in an end-to-end manner. Since it is generally believed that scene information is encoded and memorized into the regression network to form an implicit map representation (Kendall, Grimes, and Cipolla 2015), (Brahmbhatt et al. 2018) pre-built 3D maps are not required in the inference process. Thus it has a great value in the edge computing equipment. Although the absolute pose estimation method has such good properties, there still has some shortcomings that limit its wider application. In some challenging scenarios, as shown in Figure 1, there is a significant gap between the predicted results of the model and the real results, and presented in the form of outliers. In order to solve outliers, we usually introduce constraints as regular terms to constrain the prediction results keep smoothness. A lot of work has proposed various constraints to regularize the output of the model. It can simply divide these constraints into two parts: spatial and temporal. For spatial constraints, pixel-level semantic information aggregation (Tian et al. 2021) is introduced to regularize the absolute pose regression, which generate more distinguishable features for complex scenes. For temporal constraints, consecutive frames provide a wider field of views, resulting in more stable results, early attempts employ a recurrent model (Clark et al. 2017) or combine it with relative pose regression (Brahmbhatt et al. 2018) to leverage sequential smoothness.

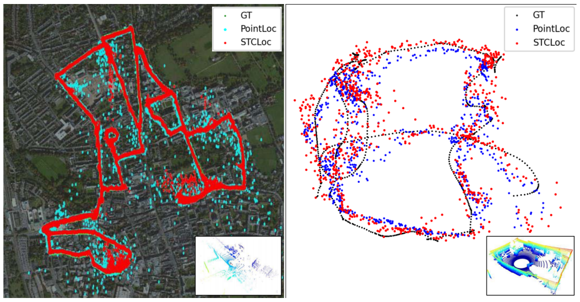

Figure 1: LiDAR localization in outdoor (left) and indoor (right) scenes from Oxford Radar RobotCar (Barnes et al. 2020a) and vReLoc (Wang et al. 2022) datasets.(this figure borrow from STCLoc(Yu et al. 2022))

Related Work

Map-Based Localization

LiDAR Localization aims to determine the precise 6-DoF pose of a given query point cloud. Conventional map-based methods focus on establishing correspondences between the query point cloud and a pre-constructed 3D map. On the

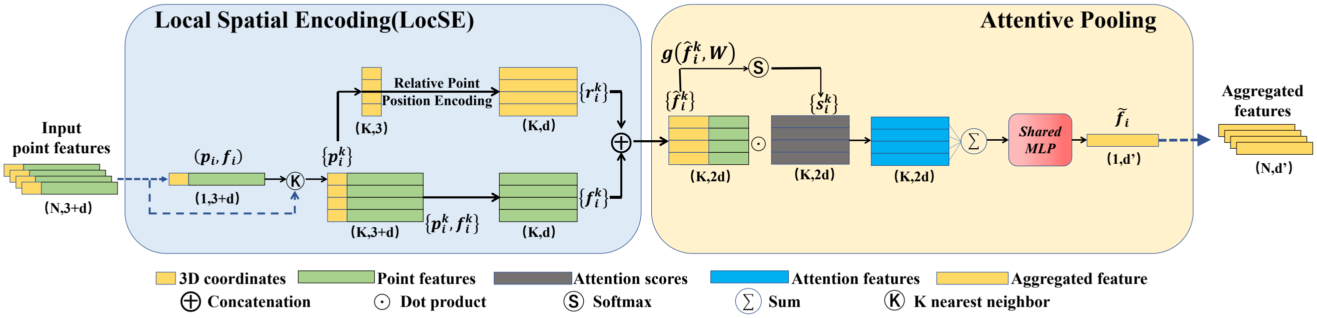

Figure 2: The proposed local feature aggregation module

other hand, retrieval-based localization identifies the most similar point cloud within a database, often resulting in fast inference times.However, these methods are typically limited to providing coarse location estimates in relation to the database.

In contrast, registration-based localization involves a fine-grained matching process between the query point cloud and the database map.USIP (Li and Lee 2019) has made improvements in feature matching by estimating keypoint positions. However, these methods require matching with the entire database during the first localization or relocalization.

Relative Pose Regression

Relative pose regression is a technique employed for localization, which involves estimating the relative motion (translation and rotation) between consecutive LiDAR frames.

One approach in this field is LO-Net (Li et al. 2019), which introduces a novel mask-weighted loss to effectively reject dynamic objects, enhancing the accuracy of the relative pose estimation.PWCLONet (Wang et al. 2021) introduces hierarchical embedding mask optimization for LiDAR odometry, improving the robustness of the localization process.

However, it's important to note that relative pose regression often requires an accurate initial pose for subsequent localization, making it less suitable for global localization tasks.

Absolute Pose Regression

Recent studies have proposed a shift towards localization using deep neural networks for regressing 6-DoF poses, eliminating the need for a pre-built 3D map during inference.

PointLoc (Wang et al. 2022) introduced a LiDAR-based learning framework for absolute pose regression, offering more robust localization due to the reliable structural information. STCLoc (Yu et al. 2022) introduced joint spatial and temporal constraints to reduce outliers for absolute pose regression. Extensive experiments further demonstrate the effectiveness of the proposed method in both outdoor and indoor datasets, which shows significant performance improvement over prior works.

Method

Inputing a large set of point clouds, first need to use neural networks for effective underground sampling and be able to do so without losing useful point features.Random sampling is computationally efficient because it is independent of the total number of input points and does not require additional memory for computation. Therefore, for large-scale point clouds, random sampling can simply and quickly realize downsampling, thereby reducing point cloud density.

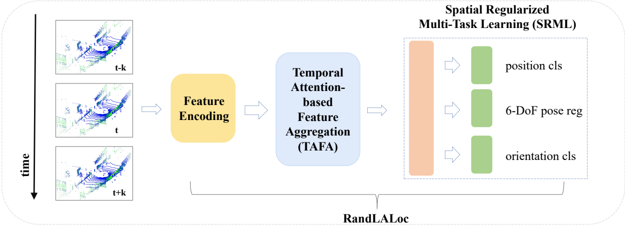

Figure 3: Full pipeline of RandLALoc

Local Feature Aggregation

Random sampling of a point cloud can result in the removal of many useful point features. Therefore, we propose a powerful local feature aggregation module to preserve the outstanding features of the point cloud, allowing the entire network to achieve an excellent trade-off between efficiency and effectiveness.

As shown in the figure 2, our local feature aggregation module is applied in parallel to each 3D point and consists of three neural units: 1.local spatial coding (LocSE), 2.attention pooling, and 3.expanded residual blocks.

A. Local Spatial Encoding: Given a point cloud P and the characteristics of each point, the LocSE unit explicitly embeds all adjacent points in the x-y-z coordinates, observing local geometric patterns, thereby enabling the network to learn the complex point cloud local structure. The LocSE unit consists of the following steps:

1) Find adjacent points: For the i^{th} point, its neighbors are collected by the nearest neighbor (KNN) algorithm LocSE unit to improve efficiency. 2) Relative point position coding: For the K points nearest the central point p_i, we explicitly encode the relative point position as follows:

$$\mathbf{r}_i^k = MLP\Big(p_i \oplus p_i^k \oplus (p_i - p_i^k) \oplus ||p_i - p_i^k||\Big) \qquad (1)$$

Where p_i and p_i^k are the coordinate information of points, $\oplus$ is the join operation, $||\cdot||$ calculates the Euclidean distance between adjacent points and the center point. Relative point position coding can help the network learn effective local features. 3) Point feature enhancement: For each adjacent point p_i^k, the encoded relative point position r_i^k is connected with its corresponding point feature f_i^k, resulting in the augmented feature vector f_i^k.

Finally, the output of the LocSE unit is a new set of adjacent features $\hat{\mathbf{F}}_i = \{\hat{\mathbf{f}}_i^1 \cdots \hat{\mathbf{f}}_i^k \cdots \hat{\mathbf{f}}_i^K\}$,that explicitly encodes the local geometry of the central point p_i.

B. Attentive Pooling: The network is used to aggregate the feature set $\hat{\mathbf{F}}_i$ of adjacent points. Inspired by the attention mechanism, the attention pool unit is designed as follows:Calculate the attention score. Given the local feature set $\hat{\mathbf{F}}_i = \{\hat{\mathbf{f}}_i^1 \cdots \hat{\mathbf{f}}_i^k \cdots \hat{\mathbf{f}}_i^K\}$, learn the unique attention score for each feature using the shared function g(), which consists of shared MLP and sof tmax. The formula is as follows:

$$\mathbf{s}_i^k = g(\hat{\mathbf{f}}_i^k, W) \tag{2}$$

Where W is the learnable weight that shares the MLP.The learned attention scores are weighted for the above local features. The features are weighted as follows:

$$\tilde{\mathbf{f}}_i = \sum_{k=1}^{K} (\hat{\mathbf{f}}_i^k \cdot \mathbf{s}_i^k) \tag{3}$$

C. Dilated Residual Block: Inspired by ResNet (He et al. 2016) and extended network (Engelmann, Kontogianni, and Leibe 2019), multiple LocSE and Attentive Pooling units are stacked with jump connections as extended residual blocks. In RandLALoc, we stack three sets of LocSE and Attentive Pooling into standard residual blocks to achieve a balance of efficiency and effectiveness.

All in all, given the input point cloud P, for the i^{th} point p_i, LocSE and Attentive Pooling unit learn and aggregate geometric patterns and features of its K-neighbor points, and generate an informative feature vector f_i. Then the corresponding extended residual connection is made to the obtained feature vectors, and the receptive field of each point is increased, while the geometric details of the point cloud are preserved.

Spatial Regularized Multi-Task Learning (SRML)

By applying the spatial constraints mentioned in stcloc, attitude classification headers are added to the absolute attitude regression network to reduce outliers.

Attitude classification is performed by subdividing the map into multiple geographic regions (locations) and point cloud directions into multiple angular regions (directions). For location classification, the map is evenly divided into blocks of the same size. For orientation classification, directions are distributed equally on the vertical axis. Each point cloud is then given its position label and direction label based on its position coordinates and Euler Angle. To take full advantage of category labels, we recommend regression using converged cross-layer features. The features of the regression flow are combined with the features from the classification flow by element multiplication. The formula W for cross-layer fusion is as follows:

$$W = f_{pos} \cdot g_{ori} \cdot h_{pose} \tag{4}$$

Among them, f_{pos}, g_{ori} and h_{pose} are the features of position classification flow, direction classification flow and postural regression flow. Both f_{pos} and g_{ori} are normalized.

The proposed position and direction classifications regularize position and direction regression, respectively, and act as constraints to force features toward a smaller and reasonable search space. The proposed multi-task learning can integrate the relationship between tasks. Performance improvements in classification will lead to improvements in regression and vice versa. Cross-layer features capture rich positioning information and provide a more discriminating representation for regression.

Temporal Attention-Based Feature Aggregation (TAFA)

Due to the lack of constraints on scene context information, absolute pose regression using single frame data usually results in many outliers. Using time constraints in sequential data is an effective method to reduce outliers. In order to learn feature correlations from sequences to reduce the fuzziness of single-frame laser scanning, a time-attention based feature aggregation (TAFA) module is proposed.

The TAFA module captures the correlation using Euclidean distance and cosine distance, as shown in equations 5 and 6. Note that the position code (Dosovitskiy et al. 2020) is added to the input feature to obtain the position information of the sequence. In the former form, we calculate the Euclidean distance for each corresponding feature to obtain similarity, as shown in equation 5.

$$\omega_i = 1/(1 + Euclidean(X_t, X_i)) \tag{5}$$

Where w_i is the similarity weight. X_t and X_i represent two adjacent or close frames. t and i are the number of frames, i = m or i = n. Then, the cosine similarity of each corresponding feature is calculated to obtain another similarity. As shown in Formula 6

$$\theta_i = 0.5 + 0.5 \cdot Cosine(X_t, X_i) \tag{6}$$

Where θ_i is the similarity weight. Both w_i and θ_i are normalized. Finally, the module combines two similarity mechanisms to perform feature aggregation. The aggregation feature O_t is represented as shown in 7:

$$O_t = X_t + \sum_{i=m, i \neq t}^{n} (\omega_i + \theta_i) \cdot X_i \tag{7}$$

The module fuses the current view into the accumulated observations and aggregates time series information to improve features using adjacent views, enabling the use of scene context information to expand the view.

Loss Function

The output of RandLALoc includes the predicted absolute attitude and attitude categories. Therefore, the design of the total loss function should consider both aspects, as shown in formula 8

$$L = l_{pose} + l_{cls} \tag{8}$$

l_{pose} and l_{cls} are loss functions for absolute pose regression and pose classification.

Table 1: Results on the Oxford

Methods	PoseLSTM	MapNet	AD-MapNet	AtLoc+	MS-Transformer	STCLoc	RandLALoc
Full7	74.00m,9.85°	61.01m,5.85°	39.18m,3.95°	34.03m,4.01°	65.38m,9.01°	6.93m,1.48°	21.72m,4.48°
Full8	128.25m,18.59°	75.35m,9.67°	66.21m,9.42°	71.51m,9.91°	88.63m,19.80°	7.44m,1.24°	20.63m,4.28°
Full9	19.12m,3.05°	44.34m,4.54°	15.10m,1.82°	10.53m,1.97°	7.62m,2.53°	6.13m,1.15°	17.68m,3.77°
Average	61.93m,9.51°	57.23m,6.53°	34.73m,4.62°	33.50m,5.16°	43.33m,9.25°	7.05m, 1.29	20.01m, 4.17°

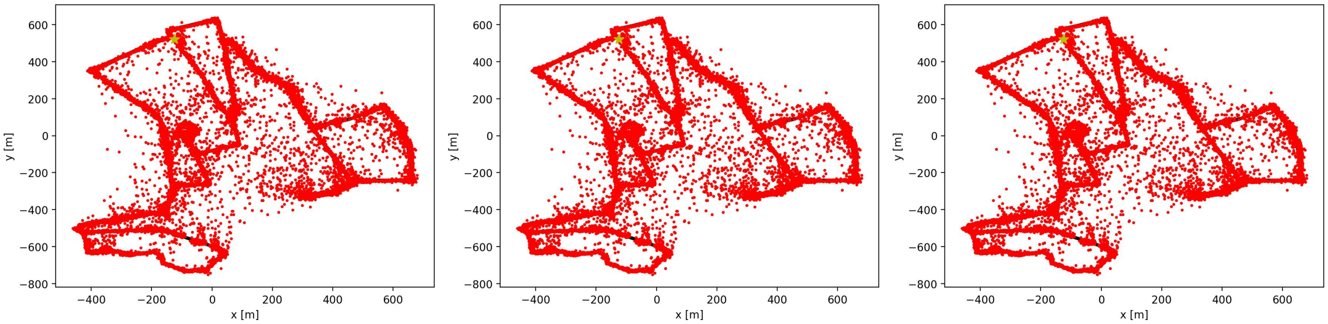

Figure 4: Trajectories of RandLALoc

Expriments

Datasets and Baselines

1). We conducted experiments on the Oxford Radar RobotCar(Barnes et al. 2020b) benchmark dataset. The Oxford Radar Robotic car is recorded by multiple on-board sensors in different weather, traffic and lighting conditions. The dataset provides 32 crossings of the Oxford Central Route (nearly 10km each).

2). The proposed network is implemented in PyTorch using the Adam optimizer with an initial learning rate of 0.001. We trained the network with two NVIDIA RTX 3090 GPUs.The point cloud is randomly downsampled to 4096 points before being fed to the network. The number of classes for the position and orientation categories is set to Oxford to 100 and 10. The hyperparameters α are set to 1.5, β to 15, and γ and σ to 1.

3). We're validating our proposed model's performance by benchmarking it against the latest absolute pose regression methods. STCLoc(Yu et al. 2022)is a LiDAR localization framework with spatio-temporal constraints. PoseLSTM (Walch et al. 2017) and MS-Transformer (Shavit, Ferens, and Keller 2021) leverage single images for pose estimation, while MapNet (Brahmbhatt et al. 2018), AD-MapNet (Huang et al. 2019), and AtLoc+ (Wang et al. 2020) utilize sequences of images for regression. All evaluations are conducted within the same environment, either using the provided source codes.

Result

We tested our method (RandLALoc) on the Oxford dataset. Considering its comprehensive collection across various challenging scenarios, this dataset demands localization methods with exceptional robustness. Our detailed comparison between RandLALoc and other methods is described in table1. Our analysis focuses on the mean position error and mean orientation error across the complete set of 4 trajectories. Notably, RandLALoc demonstrates an average error of 20.01m/4.17°, notably superior to other methods except STCLoc. It improves the image-based localization method(MS-Transformer) by 53.81% on position and 54.91% on orientation.Furthermore, RandLALoc achieves an average accuracy of 95.91% and 95.76% in position and orientation classification, respectively. The sensitivity of image-based methods to lighting fluctuations and shadow variations significantly impacts their performance, particularly evident in the notable decline on the Full7 and Full8 datasets. This also indicates our method's capability to effectively reduce scene ambiguities in large-scale outdoor environments. Consequently, we can achieve robust localization in outdoor settings. However, compared to STCLoc, Our method falls short due to our lack of sufficient computational power to train and fine-tune our model.

Conclusion

In this paper, we present RandLALoc, a novel approach tailored for LiDAR-based localization.We enhanced the formidable APR framework, STCLoc, by improving its feature extractor. Our approach integrates a classification task that categorizes the point cloud based on position and orientation. We leveraging attention-based feature aggregation to capture correlations within LiDAR sequences, facilitating the learning of discriminative features that mitigate scene ambiguities. Extensive experiments substantiate the effectiveness of our method across diverse outdoor and indoor datasets, demonstrating a significant performance boost over prior methodologies.

References

Barnes, D.; Gadd, M.; Murcutt, P.; Newman, P.; and Posner, I. 2020a. The Oxford radar RobotCar dataset: A radar extension to the Oxford RobotCar dataset. In *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 6433–6438.

Barnes, D.; Gadd, M.; Murcutt, P.; Newman, P.; and Posner, I. 2020b. The oxford radar robotcar dataset: A radar extension to the oxford robotcar dataset. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 6433–6438. IEEE.

Brahmbhatt, S.; Gu, J.; Kim, K.; Hays, J.; and Kautz, J. 2018. Geometry-aware learning of maps for camera localization. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2616–2625.

Clark, R.; Wang, S.; Markham, A.; Trigoni, N.; and Wen, H. 2017. VidLoc: A deep spatio–temporal model for 6-DoF video-clip relocalization. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 6856–6864.

Dai, Y.; et al. 2022. HSC4D: Human-Centered 4D Scene Capture in Large-Scale Indoor-Outdoor Space Using Wearable IMUs and LiDAR. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 6792–6802.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Engelmann, F.; Kontogianni, T.; and Leibe, B. 2019. Dilated point convolutions: On the receptive field of point convolutions. *arXiv preprint arXiv:1907.12046*, 2.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Huang, Z.; et al. 2019. Prior guided dropout for robust visual localization in dynamic environments. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2791–2800.

Kendall, A.; and Cipolla, R. 2017. Geometric loss functions for camera pose regression with deep learning. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 5974–5983.

Kendall, A.; Grimes, M.; and Cipolla, R. 2015. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2938–2946.

Li, J.; and Lee, G. H. 2019. USIP: Unsupervised stable interest point detection from 3D point clouds. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 361–370.

Li, Q.; et al. 2019. LO-Net: Deep real-time LiDAR odometry. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 8473–8482.

Lu, W.; Zhou, Y.; Wan, G.; Hou, S.; and Song, S. 2019. L3-Net: Towards Learning-Based LiDAR Localization for Autonomous Driving. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 6389–6398.

Radwan, N.; Valada, A.; and Burgard, W. 2018. VLocNet++: Deep multitask learning for semantic visual localization and odometry. *IEEE Robot. Autom. Lett.*, 3(4): 4407–4414.

Shavit, Y.; Ferens, R.; and Keller, Y. 2021. Learning multi-scene absolute pose regression with transformers. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2733–2742.

Tian, M.; Nie, Q.; Shen, H.; and Xia, X. 2021. Deep auxiliary learning for visual localization using colorization task.

Walch, F.; Hazirbas, C.; Leal-Taixe, L.; Sattler, T.; Hilsenbeck, S.; and Cremers, D. 2017. Image-based localization using LSTMs for structured feature correlation. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 627–637.

Wang, B.; Chen, C.; Lu, C. X.; Zhao, P.; Trigoni, N.; and Markham, A. 2020. AtLoc: Attention guided camera localization. In *Proc. AAAI Conf. Artif. Intell.*, volume 34, 10393–10401.

Wang, B.; et al. 2022. PointLoc: Deep pose regressor for LiDAR point cloud localization. *IEEE Sensors J.*, 22(1): 959–968.

Wang, G.; Wu, X.; Liu, Z.; and Wang, H. 2021. PWCLO-Net: Deep LiDAR odometry in 3D point clouds using hierarchical embedding mask optimization. In *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 15910–15919.

Xue, F.; Wang, X.; Yan, Z.; Wang, Q.; Wang, J.; and Zha, H. 2019. Local supports global: Deep camera relocalization with sequence enhancement. In *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, 2841–2850.

Yin, H.; Wang, Y.; Ding, X.; Tang, L.; Huang, S.; and Xiong, R. 2020. 3D LiDAR-Based Global Localization Using Siamese Neural Network. *IEEE Trans. Intell. Transp. Syst.*, 21(4): 1380–1392.

Yu, S.; Wang, C.; Lin, Y.; Wen, C.; Cheng, M.; and Hu, G. 2022. STCLoc: Deep LiDAR Localization With Spatio-Temporal Constraints. *IEEE Transactions on Intelligent Transportation Systems*, 24(1): 489–500.

Yu, S.; Wang, C.; Yu, Z.; Li, X.; Cheng, M.; and Zang, Y. 2021. Deep regression for LiDAR-based localization in dense urban areas. *ISPRS J. Photogramm. Remote Sens.*, 172: 240–252.

Zou, Q.; Sun, Q.; Chen, L.; Nie, B.; and Li, Q. 2022. A Comparative Analysis of LiDAR SLAM-Based Indoor Navigation for Autonomous Vehicles. *IEEE Trans. Intell. Transp. Syst.*, 23(7): 6907–6921.