LiSA: LiDAR Localization with Semantic Awareness

Bochun Yang[1*] Zijun Li[1*] Wen Li[1] Zhipeng Cai[2†]
Chenglu Wen[1] Yu Zang[1] Matthias Müller[2] Cheng Wang[1†]
[1] Fujian Key Laboratory of Sensing and Computing for Smart Cities, Xiamen University [2] Intel Labs

Abstract

LiDAR localization is a fundamental task in robotics and computer vision, which estimates the pose of a LiDAR point cloud within a global map. Scene Coordinate Regression (SCR) has demonstrated state-of-the-art performance in this task. In SCR, a scene is represented as a neural network, which outputs the world coordinates for each point in the input point cloud. However, SCR treats all points equally during localization, ignoring the fact that not all objects are beneficial for localization. For example, dynamic objects and repeating structures often negatively impact SCR. To address this problem, we introduce LiSA, the first method that incorporates semantic awareness into SCR to boost the localization robustness and accuracy. To avoid extra computation or network parameters during inference, we distill the knowledge from a segmentation model to the original SCR network. Experiments show the superior performance of LiSA on standard LiDAR localization benchmarks compared to state-of-the-art methods. Applying knowledge distillation not only preserves high efficiency but also achieves higher localization accuracy than introducing extra semantic segmentation modules. We also analyze the benefit of semantic information for LiDAR localization. Our code is released at https://github.com/Ybchun/LiSA.

1. Introduction

LiDAR localization estimates the pose of a LiDAR point cloud in a global scene map, which is a fundamental task in computer vision and robotics.

Learning-based regression methods [3–5, 19, 23, 36, 45] have shown state-of-the-art performance in LiDAR localization, where they memorize the specific scene in a neural network. Based on their regression objectives, these methods can be divided into Absolute Pose Regression (APR) and Scene Coordinates Regression (SCR). APR [5, 19, 36, 45] directly regresses the sensor pose. Despite the compact architectures, direct pose regression without explicitly utilizing the geometric information limits the accuracy of APR. Unlike APR, SCR [3, 4, 23] regresses the coordinate of each point in the global map, which provides correspondences between the input and the global point clouds. Then RANSAC [10] is applied to the correspondences to estimate the final pose. Utilizing the geometric information more explicitly allows SCR to more effectively embed and understand the scene geometry in its model parameters, resulting in higher accuracy and better robustness.

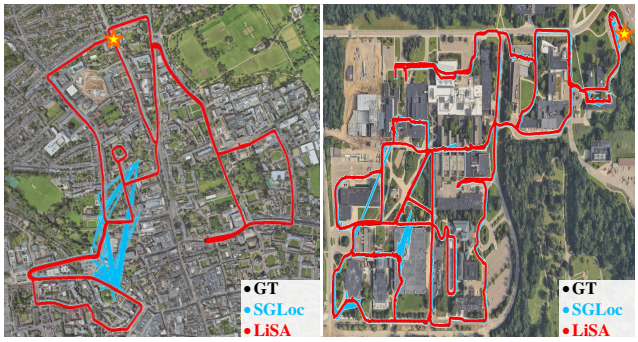


Figure 1. **Teaser.** This work studies the problem of LiDAR localization. We propose a novel method LiSA, which equips LiDAR localization methods with semantic awareness. We show the ground-truth and predicted trajectories on Oxford (left) and NCLT (right). Compared to the baseline method (SGLoc), LiSA achieves a much higher average localization accuracy and has much fewer catastrophic failures, without introducing extra computation or network parameters during inference.

SCR treats all points of the input equally. This is non-ideal for the task of localization – objects that are dynamic (*e.g.*, pedestrians and vehicles) or repetitive (*e.g.*, road surface and trees) intuitively should be less important than salient and static objects. Motivated by this observation, we propose *LiSA*, a novel method that equips SCR with semantic awareness. LiSA can robustly handle distracting objects and significantly improves the localization accuracy. To avoid introducing extra computation or network parameters during inference, we apply diffusion-based knowledge distillation during training so that we can enable semantic understanding of the original SCR network by extracting

*Equal contribution.
†Corresponding author.

knowledge from a teacher segmentation model, which is dropped after training. To the best of our knowledge, LiSA is the *first* method that effectively utilizes semantic context for LiDAR localization.

In our experiments, LiSA shows superior performance on standard LiDAR localization benchmarks compared to state-of-the-art methods. *Without* introducing extra computation or network parameters during inference, it achieves a relative improvement of more than 38% and 29% in position and orientation on the QEOxford dataset and 17% and 34% on the NCLT dataset (see Fig. 1 for an example).

2. Related Work

Conventional localization. *Conventional re-localization* methods can be broadly categorized into 1) *retrieval-based* [1, 42, 43, 51] and 2) *matching-based* [32, 33, 38] approaches. The former searches for frames most similar to the query frame in a pre-constructed global descriptor database, while the latter employs SfM tools to build a sparse 3D point cloud of the scene and matches features to find correspondences between the query frame and the point cloud. These methods require pre-stored map information, making them time-consuming and labor-intensive.

Regression-based localization. Due to the advancement of deep learning, *regression-based localization* methods have received much attention recently. Regression-based methods can be further separated into two categories: 1) Absolute Pose Regression (APR) and 2) Scene Coordinate Regression (SCR). *APR* directly regresses the 6-DoF pose of the sensor (*e.g.*, a camera or a LiDAR scanner). PoseNet[19] and several variants [5, 18, 24, 36, 44, 45] adopt a pipeline that includes only a feature extractor and an MLP to directly output the camera pose in an end-to-end manner. However, Sattler et al.[34] pointed out that APR methods are similar to retrieval-based methods, which only learn the mapping relationship between high-dimensional abstract feature vectors and poses rather than the scene information. To alleviate this issue, *SCR* [3, 4] estimates the sensor pose in a different way: the neural network predicts the 3D coordinates of each pixel/3D point in the world coordinate system, which embeds the scene information more explicitly and effectively. Then, candidate correspondences are constructed using the local coordinate and the predicted world coordinate of each point. These correspondences are finally used in RANSAC [10] to obtain the sensor pose.

In recent years, the widespread use of LiDAR stimulated several works for regression-based LiDAR localization. LiDAR APR methods, such as PointLoc[47], PosePN++[54], and HypLiLoc[46], and the SCR method SGLoc[22], have shown impressive performance on large-scale outdoor localization. Compared to the images, point clouds exhibit robustness to changes in lighting conditions and inherently carry depth information. In this work, we show that SCR methods have not fully utilized the semantic information of the scene. We introduce a novel method with semantic awareness during localization, which significantly improves the performance without introducing extra computation or network parameters during inference.

Localization using auxiliary information. Semantic information has been demonstrated to be important in conventional non-regression-based methods. Retrieval-based approaches [27, 35, 37] employ semantic information to enhance the robustness of the descriptors or serve it as the posterior information to refine the predicted pose. Several matching-based methods [9, 41], attach semantic labels to the point features during matching. This modification helps mitigate the risk of erroneous matching and significantly increases their localization accuracy.

Object detection has been applied to regression-based methods. For instance, AD-PoseNet[17] utilizes Mask R-CNN[12] to minimize the impact of dynamic foreground objects on scene understanding. ORGPoseNet[30] uses graph neural networks to learn the geometric position relationships of objects in the scene and gain more precise localization results. However, object detection tends to concentrate solely on foreground objects rather than the whole observation in the frame. On the contrary, semantic segmentation contains exhaustive scene geometry information and intricate details. This inspires us to explore the incorporation of semantic information to enhance the accuracy of regression-based LiDAR localization.

LiDAR-based semantic segmentation. LiDAR-based semantic segmentation [20, 39] aims to assign a semantic label to each point in the input LiDAR scan. Methods can generally be divided into several categories: point-based [15, 28, 29, 40], voxel-based [11, 55], projection-based [21, 25, 50], and their combination [39, 52]. Regardless of the specifics of the network architecture, they consistently expand the perception field by merging the individual point features with the coarse-grain features in the scene. In this way, the semantic segmentation networks can better understand the scene information and the characteristics of each point. In this paper, we apply a point cloud semantic segmentation model as the teacher to distill semantic knowledge to a LiDAR localization method, which effectively improves localization accuracy.

3. Method

LiSA relies on scene coordinate regression (SCR) for LiDAR localization [22]. The input of SCR is a query point cloud $\mathcal{P} = \{\mathbf{p}_i\}_{i=1}^{N}$ where $\mathbf{p}_i \in \mathbb{R}^3$ represents a 3D point in the local coordinate frame with the LiDAR scanner at the origin. The output of SCR is a set of coordinates $\mathcal{P}' = \{\mathbf{p}'_{i'}\}_{i'=1}^{N'}$ representing the location of the down-

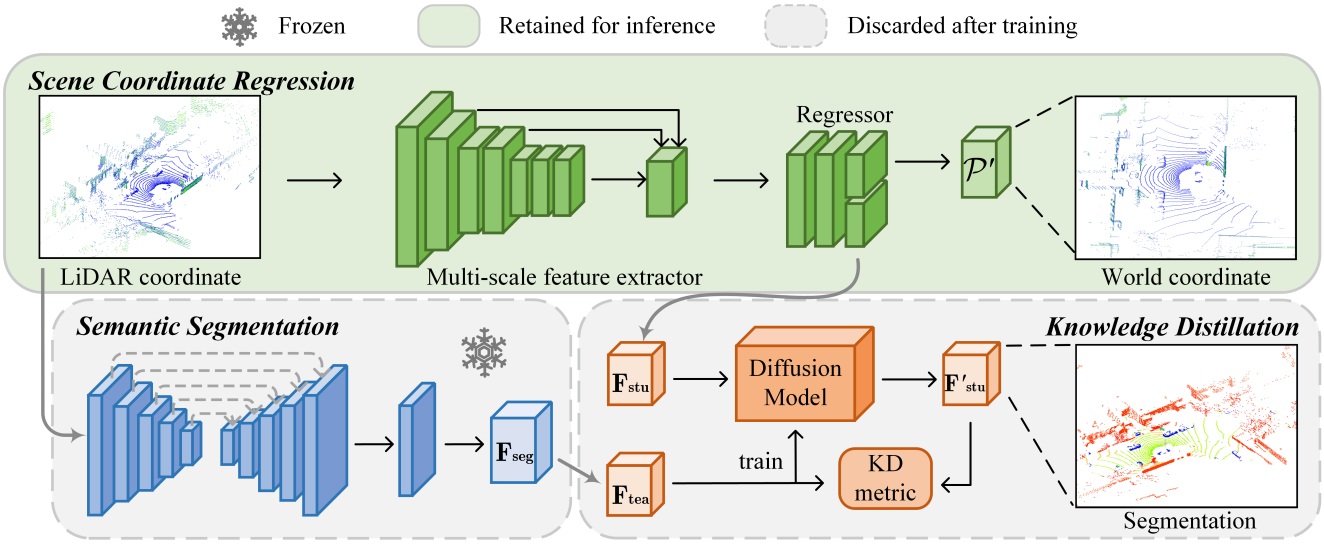


Figure 2. **The pipeline of LiSA.** It consists of three modules: scene coordinate regression, semantic segmentation (frozen), and knowledge distillation. In the scene coordinate regression module, the coordinate regression head in the regressor directly outputs scene coordinates $\mathcal{P}'$, and the semantic feature regression head learns semantic segmentation features ($\mathbf{F}_{\text{stu}}$) from the knowledge distillation module and the semantic segmentation module. After distilling the semantic knowledge during training, both semantic segmentation and knowledge distillation modules are discarded, which ensures that no extra computation and network parameters are introduced during inference.

sampled points in *world coordinates*. (SCR down-samples the input to $N' = \frac{N}{8}$ points, i.e., by a factor of 8.) The world coordinate $\mathbf{p}'_{i'}$ represents the corresponding location of $\mathbf{p}_{i'}$ in the global point cloud $\mathcal{P}^*$, which covers the complete scene of interest and is used during training.

SCR trains one model for each scene. The training loss is typically as follows:

$$\mathcal{L}_{loc} = \frac{\sum_{i=1}^{N'} |\mathbf{p}'_{i'} - \mathbf{p}^*_{i'}|}{N'}, \tag{1}$$

where $\mathbf{p}'_{i'}$ and $\mathbf{p}^*_{i'}$ are respectively the predicted and ground-truth world coordinates, $|\cdot|$ is the L1 loss.

To obtain the relative pose of $\mathcal{P}$ w.r.t. $\mathcal{P}^*$ for the final localization, RANSAC is performed on the candidate correspondences constructed by $\mathcal{P}$ and the predicted world coordinates $\mathcal{P}'$.

3.1. Motivation

The key idea of LiSA is to utilize semantic information to assist with localization. To motivate the importance of semantic information, we conduct a preliminary analysis where we filter out points belonging to specific classes in SCR, *i.e.*, the filtered points do not participate in inference and training. Instead of using the ground-truth semantic labels which are unavailable in practical applications, we use labels predicted by a pre-trained state-of-the-art 3D semantic segmentation model [20].

Tab. 1 shows the localization accuracy with different semantic categories on the QEOxford dataset [22]. It is clear to see that the localization performance varies significantly

Filter	Mean Error (m/°)	Filter	Mean Error (m/°)
all (no filter)	1.79m, 1.41°	all (no filter)	1.79m, 1.41°
no plant	1.20m, 1.97°	plant only	59.08m, 13.25°
no building	1.39m, 2.26°	building only	1.63m, 1.91°
no sidewalk	1.77m, 1.45°	sidewalk only	2.47m, 5.92°
no road	2.03m, 3.42°	road only	1.71m, 2.59°
no transportation	2.07m, 3.42°	transportation only	20.10m, 21.95°

Table 1. **Impact of semantic information on LiDAR localization.** Filtering out objects from different classes can significantly reduce or increase the position error. However, the noise in the semantic labels makes it hard to consistently improve both rotation and translation accuracy with point filtering, which motivates the design of LiSA.

with different filtering strategies, *e.g.*, without plants the position error reduces significantly. This result shows the importance of semantic information in SCR. However, due to noise in the predicted labels and the hard threshold, naive filtering does not fully utilize the semantic information – some filters improve the position accuracy while sacrificing the orientation accuracy. Moreover, segmentation models introduce significant overhead in terms of memory and computing during inference, which may be prohibitive for many practical applications. This motivates us to design LiSA, a novel framework that effectively and efficiently incorporates semantic information into SCR.

3.2. LiSA

As shown in Fig. 2, LiSA can be divided into three modules, namely scene coordinate regression (SCR), semantic segmentation, and knowledge distillation. During training, the

semantic segmentation module and the knowledge distillation module are used to transfer semantic knowledge into the SCR module. After training, only the SCR module is maintained, which can effectively utilize the transferred semantic knowledge for localization, *without* introducing extra computation or network parameters.

The *SCR module* mostly follows the architecture of SGLoc [22], which contains a multi-scale feature extractor and a regressor. To enable semantic awareness without extra computation, we add a new branch to the regressor, which is used to regress a per-point semantic feature $\mathbf{F}_{\text{stu}} \in \mathbb{R}^{N' \times d}$, where N' is the number of output points which is equivalent to the number of output world coordinates $\mathcal{P}'$, and d is the feature dimension. The original coordinate regressor branch outputs the world coordinates $\mathcal{P}'$ and is supervised by the loss in Eq. (1). The semantic regressor branch is supervised by distilling knowledge from the semantic segmentation model.

The *semantic segmentation module* consists of a pretrained 3D semantic segmentation model with frozen network parameters during training. We take the per-point last layer feature $\mathbf{F}_{\text{seg}} \in \mathbb{R}^{N \times d}$ of the segmentation model, down-sample it into $\mathbf{F}_{\text{tea}} \in \mathbb{R}^{N' \times d}$ so that $\mathbf{F}_{\text{tea}}$ has the same dimension as $\mathbf{F}_{\text{stu}}$. To verify the robustness of LiSA, we experiment with SphereFormer [20] and SPVNAS [39] pretrained on datasets non-overlapping with any LiDAR localization dataset. We ensure $\mathbf{F}_{\text{tea}}$ and $\mathbf{F}_{\text{stu}}$ have the same dimension d for each feature point by adjusting the output dimension of the semantic regressor branch.

Knowledge distillation [13] is the standard approach for reducing network parameters or transferring knowledge across different modalities, resulting in a (lightweight) student network that preserves the capabilities of the teacher network. LiSA applies knowledge distillation to transfer the semantic knowledge from the segmentation module to the SCR module. This is achieved by optimizing the following distillation loss:

$$\mathcal{L}_{kd} = \delta(\mathbf{F}_{\text{stu}}, \mathbf{F}_{\text{tea}}), \tag{2}$$

where $\delta(\cdot)$ measures the difference between the features from the teacher ($\mathbf{F}_{\text{tea}}$) and the student ($\mathbf{F}_{\text{stu}}$). Intuitively, the distillation process encourages the semantic feature $\mathbf{F}_{\text{stu}}$ from SCR to be similar to $\mathbf{F}_{\text{tea}}$, which requires the SCR module to be able to distinguish different semantic classes. The final loss function of the entire network is:

$$\mathcal{L} = \mathcal{L}_{loc} + \mathcal{L}_{kd}. \tag{3}$$

Various knowledge distillation methods can be applied to realize $\mathcal{L}_{kd}$. LiSA applies the recent diffusion-based distillation [16]. Specifically, we train a diffusion model with the teacher feature $\mathbf{F}_{\text{tea}}$ by gradually adding noise to $\mathbf{F}_{\text{tea}}$ and let the diffusion model learn to predict the noise. Following the forward noise process of DDPM [14], the noisy teacher feature $\mathbf{F}^{(t)}_{\text{tea}}$ at time step t can be obtained by:

$$\mathbf{F}^{(t)}_{\text{tea}} = \sqrt{\bar{\alpha}^{(t)}}\mathbf{F}_{\text{tea}} + \sqrt{1-\bar{\alpha}^{(t)}}\boldsymbol{\epsilon}^{(t)}, \boldsymbol{\epsilon}^{(t)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{4}$$

The loss for training the diffusion model is

$$\mathcal{L}_{ddpm} = ||\sigma(\mathbf{F}^{(t)}_{\text{tea}}, t) - \boldsymbol{\epsilon}^{(t)}||_2, \tag{5}$$

where $\sigma(\mathbf{F}^{(t)}_{\text{tea}}, t)$ represents the predicted noise from the diffusion model given the time step t and the noisy feature input $\mathbf{F}^{(t)}_{\text{tea}}$, $\boldsymbol{\epsilon}^{(t)}$ represents the ground-truth noise.

To enable the knowledge transfer between $\mathbf{F}_{\text{stu}}$ and $\mathbf{F}_{\text{tea}}$, we consider $\mathbf{F}_{\text{stu}}$ as the noisy version of $\mathbf{F}_{\text{tea}}$, and use the diffusion model to obtain a denoised version $\mathbf{F}'_{\text{stu}}$ of $\mathbf{F}_{\text{stu}}$. Then we apply the normal knowledge distillation (L1) loss to $\mathbf{F}'_{\text{stu}}$ and $\mathbf{F}_{\text{tea}}$:

$$\mathcal{L}'_{kd} = |\mathbf{F}'_{\text{stu}} - \mathbf{F}_{\text{tea}}|. \tag{6}$$

The final distillation loss is:

$$\mathcal{L}_{kd} = \lambda_1 \mathcal{L}_{ddpm} + \lambda_2 \mathcal{L}'_{kd}, \tag{7}$$

where λ_1 and λ_2 are hyper-parameters to balance the losses. Simply setting both λ_1 and λ_2 to 1 works well in practice. Note that we use the vanilla DDPM architecture instead of latent diffusion [31]. Hence, we do not have losses on autoencoders as in Sec. 3.2 of [16].

Besides introducing no overhead to SCR (by dropping both segmentation and distillation modules after training), knowledge distillation also allows the network to more effectively utilize the imperfect output of the semantic segmentation models — we only use the semantic features to transfer the knowledge, instead of using hard filtering based on the imperfect semantic labels. Our experiments (Sec. 4.2) show that this design remarkably reduces both the position and orientation error.

4. Experiment

4.1. Setup

Baseline. We compare LiSA against different types of state-of-the-art methods: 1) For *retrieval-based* methods, we choose PNVLAD [43] which employs PointNet [28] and NetVLAD [1] to generate high-quality global descriptors. 2) For *3D matching-based* methods, we choose DCP [48] which utilizes PointNet [28] and DGCNN [49] as embedding networks. 3) For *APR* methods, we choose PointLoc [47], PosePN++ [54], HyLiLoc [46] and STCLoc [53]. For *SCR*, SGLoc [22] is the first method of this type in LiDAR localization and it is also the most competitive baseline.

Dataset. Following [22], we perform evaluations on 3 public datasets, namely Oxford Radar RobotCar [2], QEOxford [22] and NCLT [6]. *Oxford Radar RobotCar* contains

QEOxford dataset								
Methods	Retrieval	Matching	Absolute Pose Regression				Sence Coordinate Regression	
	PNVLAD	DCP	PointLoc	PosePN++	STCLoc	HypLiLoc	SGLoc	LiSA (ours)
15-13-06-37	10.90m, 2.49°	10.61m, 2.56°	10.75m, 2.36°	4.54m, 1.83°	5.14m, 1.27°	5.03m, 1.46°	1.79m, 1.67°	**0.94m**, **1.10°**
17-13-26-39	14.60m, 2.46°	11.44m, 2.14°	11.07m, 2.21°	6.44m, 1.78°	6.12m, **1.21°**	4.31m, 1.43°	1.81m, 1.76°	**1.17m**, **1.21°**
17-14-03-00	11.28m, 2.21°	10.90m, 2.01°	11.53m, 1.92°	4.89m, 1.55°	5.32m, **1.08°**	3.61m, 1.11°	1.33m, 1.59°	**0.84m**, 1.15°
18-14-14-42	9.00m, 1.90°	9.51m, 2.08°	9.82m, 2.07°	4.64m, 1.61°	4.76m, 1.19°	2.61m, **1.09°**	1.19m, 1.39°	**0.85m**, 1.11°
Average	11.45m, 2.27°	10.62m, 2.20°	10.79m, 2.14°	5.13m, 1.69°	5.34m, 1.18°	3.89m, 1.27°	1.53m, 1.60°	**0.95m**, **1.14°**

Table 2. **Quantitative results on QEOxford.** Mean position error (m) and mean orientation error (°) for various methods are reported. Best performance is highlighted in **bold**, lower is better. LiSA outperforms all baseline methods in terms of both position and orientation accuracy.

Oxford dataset								
Methods	Retrieval	Matching	Absolute Pose Regression				Sence Coordinate Regression	
	PNVLAD	DCP	PointLoc	PosePN++	STCLoc	HypLiLoc	SGLoc	LiSA (ours)
15-13-06-37	18.14m, 3.28°	16.04m, 4.54°	12.42m, 2.26°	9.59m, 1.92°	6.93m, 1.48°	6.88m, **1.09°**	3.01m, 1.91°	**2.36m**, 1.29°
17-13-26-39	24.57m, 3.08°	16.22m, 3.56°	13.14m, 2.50°	10.66m, 1.92°	7.55m, **1.23°**	6.79m, 1.29°	4.07m, 2.07°	**3.47m**, 1.43°
17-14-03-00	19.93m, 3.13°	14.87m, 3.45°	12.91m, 1.92°	9.01m, 1.51°	7.44m, 1.24°	5.82m, **0.97°**	3.37m, 1.89°	**3.19m**, 1.34°
18-14-14-42	15.59m, 2.63°	12.97m, 3.99°	11.31m, 1.98°	8.44m, 1.71°	6.13m, 1.15°	3.45m, **0.84°**	2.12m, 1.66°	**1.95m**, 1.23°
Average	19.56m, 3.03°	15.03m, 3.89°	12.45m, 2.17°	9.43m, 1.77°	7.01m, 1.28°	5.74m, **1.05°**	3.14m, 1.88°	**2.74m**, 1.32°

Table 3. **Quantitative results on Oxford.** Mean position error (m) and mean orientation error (°) for various methods are reported; best performance is highlighted in **bold**, lower is better. LiSA significantly improves both the position and rotation accuracy of the baseline SGLoc. The ground-truth trajectory noise in the original Oxford dataset makes SCR perform slightly worse in terms of rotational accuracy compared to the best APR methods HypLiLoc. However, the positional accuracy is much higher.

32 repeated traversals along the central Oxford with diverse weather and complex traffic conditions, roughly $10km$ of the trajectory length, and a $2km^2$ range coverage. *QEOxford* is a quality-enhanced version of the Oxford dataset, with GPS/INS errors minimized by alignment techniques, which has been demonstrated to be beneficial for localization [22]. *NCLT* is an extensive collection of LiDAR data captured at the Michigan North Campus. It comprises 27 tracks, each approximately $5.5km$ long, covering an area of $0.45km^2$. The dataset contains challenging scenarios such as dynamic objects, illumination variation, seasonal and weather changes, and long-term structural alterations due to ongoing construction projects.

Implementation. LiSA is implemented with Pytorch [26], Spconv [8] and the Minkowski Engine [7]. We conduct our experiments on a server equipped with an Intel(R) Xeon(R) Silver 4314 CPU, 256GB of RAM, and four NVIDIA RTX 3090 GPUs. During training, we employ the Adam optimizer with an initial learning rate of 0.01, weight decay of 0.95, and batch size of 100.

4.2. Main Results

Results on Oxford. Tab. 2 shows the quantitative results on the quality-enhanced Oxford dataset. LiSA achieves the lowest position and orientation error at 0.95m and 1.14°, respectively. This result improves the previous state-of-the-art (SGLoc [22]) by 38% and 29% on position and orientation respectively *without sacrificing the efficiency*. In addition, it is the first time that a sub-meter accuracy can be achieved without additional backend optimization, such as pose graph optimization (PGO) [5, 22]. LiSA achieves better results with much higher efficiency than SGLoc+PGO (38ms vs 288ms); see Appendix 3 for details. Fig. 3 shows the qualitative results on trajectory 15-13-06-37. LiSA closely aligns with the ground-truth trajectory and has significantly fewer outliers, *i.e.*, fewer catastrophic failures. The experimental results demonstrate that semantic awareness helps LiSA to understand the scene information and obtain more accurate localization results.

Tab. 3 reports the results on the (non-enhanced) Oxford dataset. LiSA also significantly improves over the baseline SGLoc in both position and orientation on this dataset. Although LiSA has some gaps in terms of orientation compared to the best APR method [46], it surpasses all previous methods in terms of position accuracy. We discuss this in detail in Appendix 3.

Results on NCLT. Tab. 4 shows the quantitative result on the NCLT dataset. Though the segmentation model used to train LiSA performs much worse on NCLT than on Oxford (see Appendix 2 for visualization of segmentation results on Oxford and NCLT), LiSA still outperforms all competitors, achieving sub-meter accuracy on 3 of the 4 trajectories. Compared to the previous state-of-the-art method, SGLoc [22], there is an improvement of 17% and 34% in these metrics. This further demonstrates the robustness of our framework. Fig. 4 shows the qualitative results on trajectory

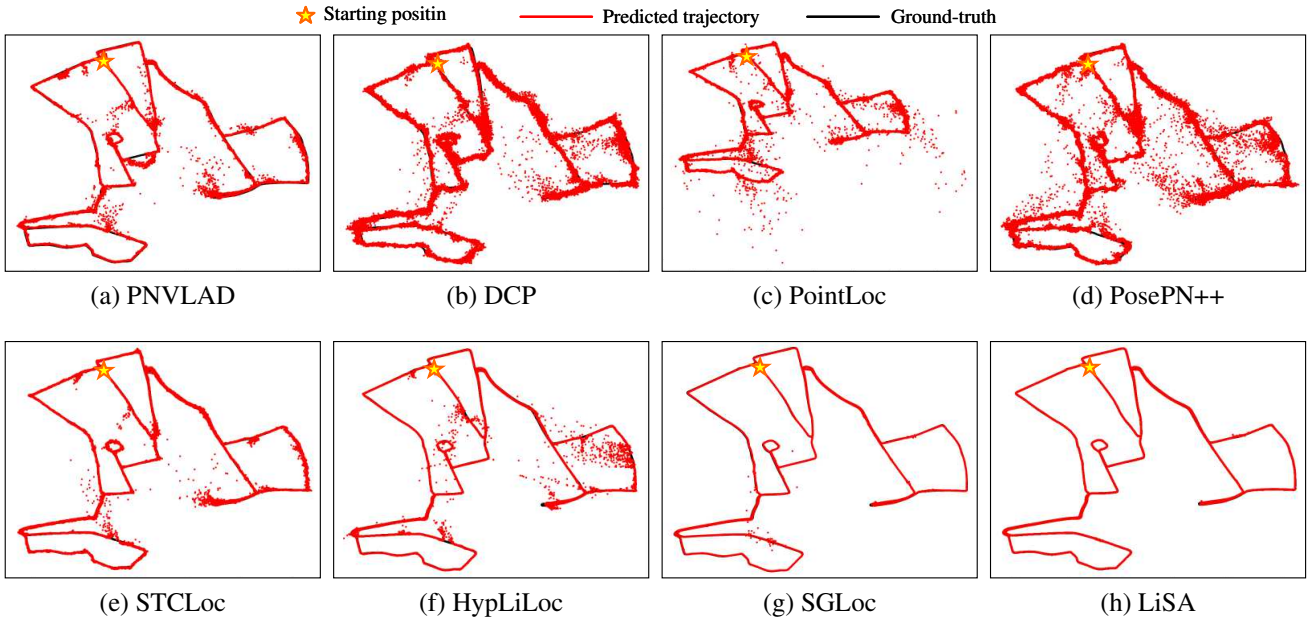


Figure 3. **Visualization of different methods on QEOxford.** The black and red points represent the ground-truth and estimation poses respectively, and the star indicates the first frame. Compared with retrieval and APR, trajectories of SCR methods (LiSA and its baseline SGLoc) are much more consistent with the ground-truth. LiSA not only is more accurate on average, but also does not suffer from catastrophic prediction errors existing in other baselines.

NCLT dataset								
Methods	Retrieval	Matching	Absolute Pose Regression				Sence Coordinate Regression	
	PNVLAD	DCP	PointLoc	PosePN++	STCLoc	HypLiLoc	SGLoc	LiSA (ours)
2012-02-12	7.75m, 6.49°	9.84m, 6.84°	7.23m, 4.88°	4.97m, 3.75°	4.91m, 4.43°	1.71m, 3.56°	1.20m, 3.08°	**0.97m, 2.23°**
2012-02-19	7.47m, 5.49°	8.27m, 5.16°	6.31m, 3.89°	3.68m, 2.65°	3.25m, 3.10°	1.68m, 2.69°	1.20m, 3.05°	**0.91m, 2.09°**
2012-03-31	6.98m, 5.67°	8.94m, 5.96°	6.71m, 4.32°	4.35m, 3.38°	3.75m, 4.04°	1.52m, 2.90°	1.12m, 3.28°	**0.87m, 2.21°**
2012-05-26	14.34m, 7.93°	15.62m, 7.99°	10.02m, 5.32°	9.59m, 4.49°	8.67m, 5.23°	**2.90m**, 3.47°	3.81m, 4.74°	3.30m, **2.84°**
Average	9.14m, 6.40°	10.67m, 6.49°	7.57m, 4.60°	5.65m, 3.57°	5.15m, 4.18°	1.95m, 3.16°	1.83m, 3.54°	**1.51m, 2.34°**

Table 4. **Quantitative results on NCLT.** Mean position error (m) and mean orientation error (°) for various methods are reported. Even though the semantic segmentation model [20] performs not perfectly on the NCLT dataset, LiSA still demonstrates outstanding performance, surpassing all competitors by a large margin.

'2012-03-31'. Similar to Fig. 3, LiSA demonstrates accurate and robust localization results with much fewer catastrophic failures than other methods. The results on NCLT provide further evidence that LiSA can effectively leverage semantic information for localization tasks to boost performance.

Speed. Inference time is a crucial metric in the localization task. Given the laser scanning rates of 20Hz and 10Hz for the Oxford and NCLT datasets respectively, a real-time algorithm needs to keep the inference time below 50ms and 100ms. Due to the use of distillation-based training, LiSA only uses the baseline SCR module during inference, which does not introduce additional time and memory compared to the baseline [22]. The average run time of LiSA on Oxford and NCLT is 38ms and 75ms respectively (batch size = 1), which satisfies the real-time speed requirement.

4.3. Analysis

Effect of semantic information. To further understand the behavior of LiSA, we compare the network activation with and without semantic awareness. Fig. 6 (a) illustrates the distribution of activation values on a point cloud sampled from QEOxford (shown in Fig. 5). Without semantic awareness, most activation values cluster around the center, indicating that nearly all points are treated equally. In contrast, LiSA (with semantic awareness) learns to adaptively focus on more important points and ignore distracting points, resulting in widely distributed activation values and a much higher localization accuracy (Fig. 6 (b)). In Fig. 5, we show the activation on individual points of the sampled point cloud in Fig. 6. As shown in the zoom-in views on the right, the activation of LiSA is higher on salient objects (trunks, buildings, and parked cars), which are impor-

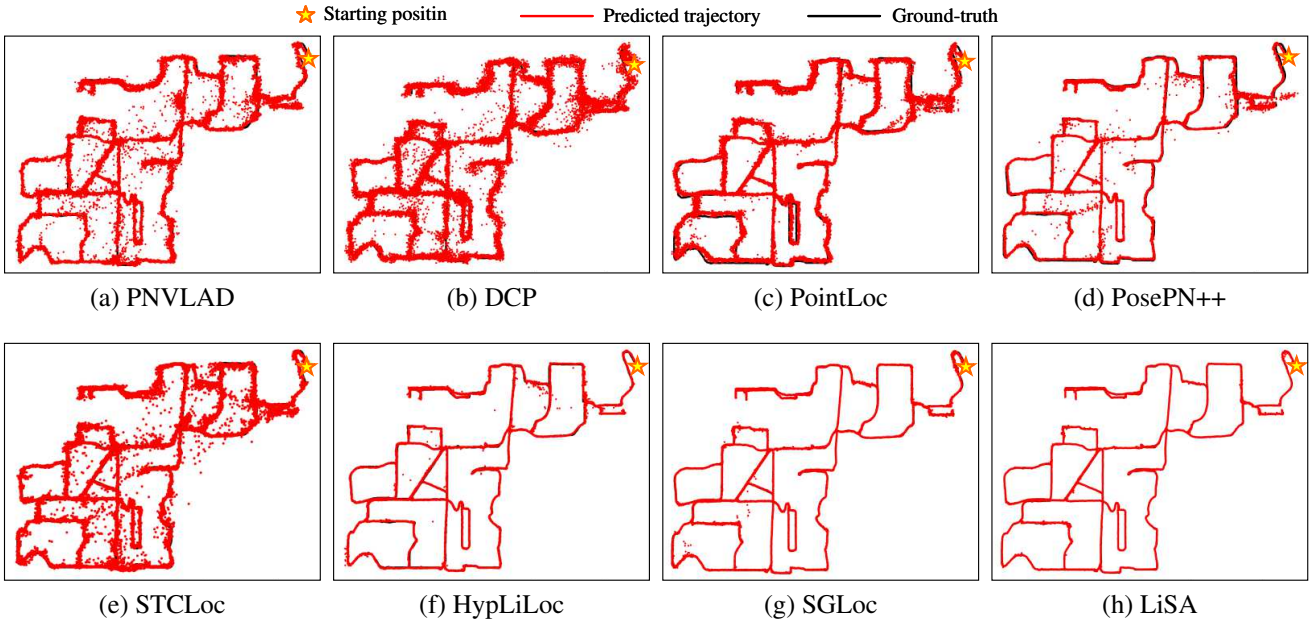


Figure 4. **Visualization of different methods on NCLT**. The black and red points represent the ground-truth and estimation poses respectively, and the star indicates the first frame. Similar to Fig. 3, LiSA outperforms all competitors with much fewer frequent catastrophic localization errors.

tant for localization. Dynamic objects such as pedestrians have lower activation values, making LiSA insensitive to them. In contrast, the activation value remains similar for different objects without the help of semantic knowledge. This behavior clearly shows the importance of the semantic knowledge for LiDAR localization.

The way to use semantic information. LiSA applies knowledge distillation (KD) to provide the SCR network with semantic awareness. Here, we analyze the importance of this strategy. Specifically, we replace this strategy with 3 alternatives: 1) *No semantic*, where we do not use any semantic information. 2) *Point filter*, which, as described in Sec. 3.1, ignores the points from specific classes, with the class labels provided by a segmentation network. We use the best setup from Tab. 1 in this case. 3) *Feature concate*, which concatenates the features from the semantic segmentation and localization networks and then feeds the aggregated features into the regressor. We perform analysis on the quality-enhanced Oxford dataset. As shown in Tab. 5, though more accurate than 'no semantic', 'point filter' and 'feature concate' cannot encode the semantic knowledge into the SCR network, resulting in worse performance than knowledge distillation (KD). Meanwhile, these two alternatives rely on the segmentation model during inference, which introduces extra computation and memory.

Segmentation quality. We also investigate the influence of the semantic segmentation quality on localization. As shown in Fig. 7, SphereFormer [20] has higher segmentation performance than SPVNAS [39]. Tab. 6 shows the performance of LiSA with both models, which reflects the positive correlation between the segmentation performance and the localization accuracy, indicating the potential improvement of LiSA by incorporating better segmentation models in the future. Meanwhile, LiSA still outperforms

	No semantic	Point filter	Feature concate	KD
15-13-06-37	1.83m, 1.43°	1.17m, 1.86°	1.11m, 1.57°	**0.94m, 1.10°**
17-13-26-39	2.10m, 1.47°	1.64m, 2.14°	1.47m, 1.72°	**1.17m, 1.21°**
17-14-03-00	1.59m, 1.39°	1.00m, 2.01°	0.99m, 1.60°	**0.84m, 1.15°**
18-14-14-42	1.62m, 1.33°	0.98m, 1.87°	0.97m, 1.51°	**0.85m, 1.11°**
Average	1.79m, 1.41°	1.20m, 1.97°	1.14m, 1.60°	**0.95m, 1.14°**

Table 5. **Different ways of combining semantic information.** Methods 'point filter' and 'feature concate' indeed show some improvement in accuracy after incorporating semantic information, but there is a slight decline in orientation. In comparison, LiSA, which integrates semantic information using knowledge distillation, exhibits a significant advantage.

	loss function of KD	quality of semantic features	Mean Error (m/°)
1	L1	low (SPVNAS)	1.28m/1.53°
2	L1	high (SphereFormer)	1.00m/1.15°
3	DDPM	low (SPVNAS)	1.15m/1.40°
4	DDPM	high (SphereFormer)	0.95m/1.14°

Table 6. **Impact of the segmentation quality and the KD loss on QEOxford.** Positive correlations exist between the segmentation accuracy of the teacher model and the localization accuracy of LiSA. Nonetheless, LiSA outperforms the baseline SGLoc by a large margin even with the low quality segmentation teacher (SPVNAS). Utilizing DDPM to obtain better student features also contributes to the improvement in accuracy.

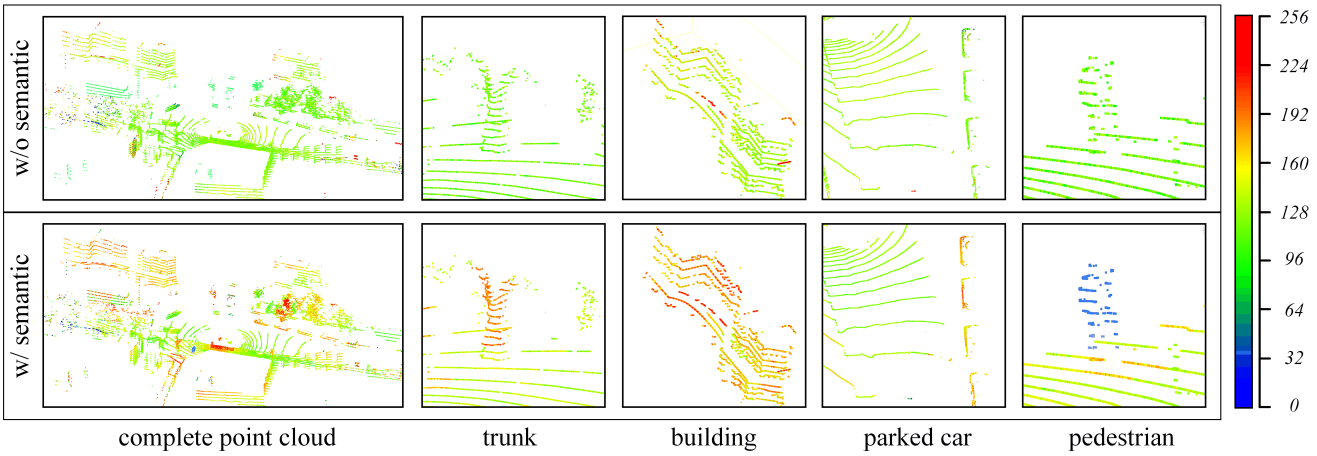
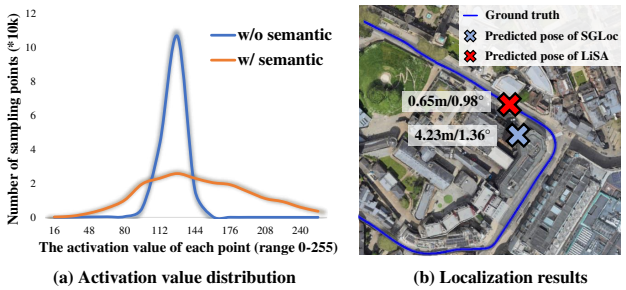


Figure 5. **The behavior of SCR with and without semantic awareness.** Given a point cloud sampled from QEOxford, we show the point-wise activation value with and without using semantic information. Warmer colors denote higher activation values. **Left:** The activation map of the complete point cloud. **Right:** Zoom-in local views. Similar activation values are assigned to all points if no semantic knowledge is utilized, whereas LiSA can discriminate important points for localization and down-weight distracting points such as pedestrians.



(a) Activation value distribution (b) Localization results

Figure 6. **Activation value distributions and localization results of LiSA (w/ semantic) and SGLoc (w/o semantic). (a):** Given a point cloud sampled from QEOxford, we first aggregate features in the regressor (before heads) and normalize their activation values into $[0, 255]$. Without semantic awareness, most activation values of SGLoc are clustered in the center, indicating that almost all points are treated equally. **(b):** The localization accuracy with semantic awareness (LiSA) is much higher than the baseline (SGLoc) without semantic awareness.

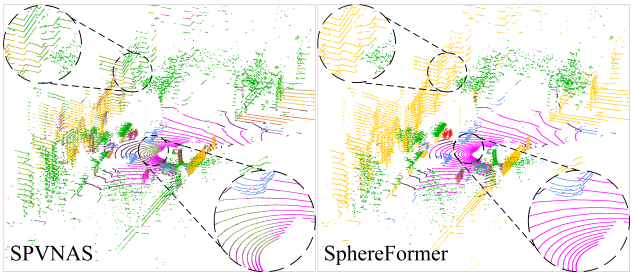


Figure 7. **Qualitative comparison of segmentation methods.** SphereFormer on the right performs better as SPVNAS misclassifies more points on the ground and buildings.

the baseline method SGLoc by a large margin, even with the low-quality segmentation model SPVNAS. This demonstrates the robustness of our pipeline w.r.t. the choice of segmentation methods.

Knowledge distillation loss. We further investigate the impact of different loss functions on the knowledge distillation module. As shown in Tab. 6, diffusion-based distillation performs better than using the L1 loss across different segmentation models, especially on low-quality segmentation models.

5. Conclusion

In this work, we propose LiSA, a novel scene coordinate regression framework for LiDAR localization. To the best of our knowledge, LiSA is the first work that integrates semantic information into regression-based localization. Instead of naively relying on labels generated by segmentation models, we apply diffusion-based knowledge distillation to transfer relevant semantic knowledge from a segmentation model directly into the SCR network. This enables adaptive extraction of semantic knowledge useful for localization with minimum negative impact from noisy segmentation. At the same time, due to the distillation-based training all extra modules can be discarded after training avoiding additional computation or network parameters w.r.t. the base SCR network during inference. LiSA achieves state-of-the-art performance on challenging LiDAR localization datasets, significantly surpassing previous methods.

Acknowledgements This work was supported by the National Natural Science Foundation of China (No.62171393), the Fundamental Research Funds for the Central Universities (No.20720220064, No.20720230033), and PDL (2022-PDL-12).

References

[1] Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *CVPR*, pages 5297–5307, 2016. 2, 4

[2] 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 *ICRA*, pages 6433–6438. IEEE, 2020. 4, 1

[3] Eric Brachmann and Carsten Rother. Visual camera relocalization from rgb and rgb-d images using dsac. *IEEE TPAMI*, 44(9):5847–5865, 2021. 1, 2

[4] Eric Brachmann, Alexander Krull, Sebastian Nowozin, Jamie Shotton, Frank Michel, Stefan Gumhold, and Carsten Rother. Dsac-differentiable ransac for camera localization. In *CVPR*, pages 6684–6692, 2017. 1, 2

[5] Samarth Brahmbhatt, Jinwei Gu, Kihwan Kim, James Hays, and Jan Kautz. Geometry-aware learning of maps for camera localization. In *CVPR*, pages 2616–2625, 2018. 1, 2, 5

[6] Nicholas Carlevaris-Bianco, Arash K Ushani, and Ryan M Eustice. University of michigan north campus long-term vision and lidar dataset. *The International Journal of Robotics Research*, 35(9):1023–1035, 2016. 4, 1

[7] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *CVPR*, pages 3075–3084, 2019. 5

[8] Spconv Contributors. Spconv: Spatially sparse convolution library. https://github.com/traveller59/spconv, 2022. 5

[9] Renaud Dubé, Daniel Dugas, Elena Stumm, Juan Nieto, Roland Siegwart, and Cesar Cadena. Segmatch: Segment based place recognition in 3d point clouds. In *ICRA*, pages 5266–5272. IEEE, 2017. 2

[10] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981. 1, 2

[11] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, pages 9224–9232, 2018. 2

[12] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 2

[13] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. In advances in neural information processing systems (neurips) deep learning workshop. 2015. 4

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 4

[15] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *CVPR*, pages 11108–11117, 2020. 2

[16] Tao Huang, Yuan Zhang, Mingkai Zheng, Shan You, Fei Wang, Chen Qian, and Chang Xu. Knowledge diffusion for distillation. 2024. 4

[17] Zhaoyang Huang, Yan Xu, Jianping Shi, Xiaowei Zhou, Hujun Bao, and Guofeng Zhang. Prior guided dropout for robust visual localization in dynamic environments. In *ICCV*, pages 2791–2800, 2019. 2

[18] Alex Kendall and Roberto Cipolla. Geometric loss functions for camera pose regression with deep learning. In *CVPR*, pages 5974–5983, 2017. 2

[19] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *ICCV*, pages 2938–2946, 2015. 1, 2

[20] Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical transformer for lidar-based 3d recognition. In *CVPR*, pages 17545–17555, 2023. 2, 3, 4, 6, 7, 1

[21] Felix Järemo Lawin, Martin Danelljan, Patrik Tosteberg, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. Deep projective 3d semantic segmentation. In *Computer Analysis of Images and Patterns: 17th International Conference, CAIP 2017, Ystad, Sweden, August 22-24, 2017, Proceedings, Part I 17*, pages 95–107. Springer, 2017. 2

[22] Wen Li, Shangshu Yu, Cheng Wang, Guosheng Hu, Siqi Shen, and Chenglu Wen. Sgloc: Scene geometry encoding for outdoor lidar localization. In *CVPR*, pages 9286–9295, 2023. 2, 3, 4, 5, 6, 1

[23] Xiaotian Li, Shuzhe Wang, Yi Zhao, Jakob Verbeek, and Juho Kannala. Hierarchical scene coordinate classification and regression for visual localization. In *CVPR*, pages 11983–11992, 2020. 1

[24] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Image-based localization using hourglass networks. In *ICCV*, pages 879–886, 2017. 2

[25] Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *IROS*, 2019. 2

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *NeurIPS*, 32, 2019. 5

[27] Maxime Pietrantoni, Martin Humenberger, Torsten Sattler, and Gabriela Csurka. Segloc: Learning segmentation-based representations for privacy-preserving visual localization. In *CVPR*, pages 15380–15391, 2023. 2

[28] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, pages 652–660, 2017. 2, 4

[29] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 30, 2017. 2

[30] Chengyu Qiao, Zhiyu Xiang, Xinglu Wang, Shuya Chen, Yuangang Fan, and Xijun Zhao. Objects matter: Learning object relation graph for robust absolute pose regression. *Neurocomputing*, 521:11–26, 2023. 2

[31] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 4

[32] Torsten Sattler, Michal Havlena, Konrad Schindler, and Marc Pollefeys. Large-scale location recognition and the geometric burstiness problem. In *CVPR*, pages 1582–1590, 2016. 2

[33] Torsten Sattler, Akihiko Torii, Josef Sivic, Marc Pollefeys, Hajime Taira, Masatoshi Okutomi, and Tomas Pajdla. Are large-scale 3d models really necessary for accurate visual localization? In *CVPR*, pages 1637–1646, 2017. 2

[34] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixe. Understanding the limitations of cnn-based absolute camera pose regression. In *CVPR*, pages 3302–3312, 2019. 2

[35] Johannes L Schönberger, Marc Pollefeys, Andreas Geiger, and Torsten Sattler. Semantic visual localization. In *CVPR*, pages 6896–6906, 2018. 2

[36] Yoli Shavit, Ron Ferens, and Yosi Keller. Learning multi-scene absolute pose regression with transformers. In *ICCV*, pages 2733–2742, 2021. 1, 2

[37] Tianxin Shi, Shuhan Shen, Xiang Gao, and Lingjie Zhu. Visual localization using sparse semantic 3d map. In *ICIP*, pages 315–319. IEEE, 2019. 2

[38] Linus Svärm, Olof Enqvist, Fredrik Kahl, and Magnus Oskarsson. City-scale localization for cameras with known vertical direction. *IEEE TPAMI*, 39(7):1455–1461, 2016. 2

[39] Haotian Tang, Zhijian Liu, Shengyu Zhao, Yujun Lin, Ji Lin, Hanrui Wang, and Song Han. Searching efficient 3d architectures with sparse point-voxel convolution. In *ECCV*, pages 685–702. Springer, 2020. 2, 4, 7, 1

[40] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *ICCV*, pages 6411–6420, 2019. 2

[41] Carl Toft, Erik Stenborg, Lars Hammarstrand, Lucas Brynte, Marc Pollefeys, Torsten Sattler, and Fredrik Kahl. Semantic match consistency for long-term visual localization. In *ECCV*, pages 383–399, 2018. 2

[42] Akihiko Torii, Relja Arandjelovic, Josef Sivic, Masatoshi Okutomi, and Tomas Pajdla. 24/7 place recognition by view synthesis. In *CVPR*, pages 1808–1817, 2015. 2

[43] Mikaela Angelina Uy and Gim Hee Lee. Pointnetvlad: Deep point cloud based retrieval for large-scale place recognition. In *CVPR*, pages 4470–4479, 2018. 2, 4

[44] Florian Walch, Caner Hazirbas, Laura Leal-Taixe, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. Image-based localization using lstms for structured feature correlation. In *ICCV*, pages 627–637, 2017. 2

[45] Bing Wang, Changhao Chen, Chris Xiaoxuan Lu, Peijun Zhao, Niki Trigoni, and Andrew Markham. Atloc: Attention guided camera localization. In *AAAI*, pages 10393–10401, 2020. 1, 2

[46] 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 *CVPR*, pages 5176–5185, 2023. 2, 4, 5, 1

[47] 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 Sensors Journal*, 22(1):959–968, 2021. 2, 4

[48] Yue Wang and Justin M Solomon. Deep closest point: Learning representations for point cloud registration. In *ICCV*, pages 3523–3532, 2019. 4

[49] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM TOG*, 38(5): 1–12, 2019. 4

[50] Bichen Wu, Alvin Wan, Xiangyu Yue, and Kurt Keutzer. Squeezeseg: Convolutional neural nets with recurrent crf for real-time road-object segmentation from 3d lidar point cloud. In *ICRA*, pages 1887–1893. IEEE, 2018. 2

[51] Yan Xia, Mariia Gladkova, Rui Wang, Qianyun Li, Uwe Stilla, Joao F Henriques, and Daniel Cremers. Casspr: Cross attention single scan place recognition. In *ICCV*, pages 8461–8472, 2023. 2

[52] Xu Yan, Jiantao Gao, Chaoda Zheng, Chao Zheng, Ruimao Zhang, Shuguang Cui, and Zhen Li. 2dpass: 2d priors assisted semantic segmentation on lidar point clouds. In *ECCV*, pages 677–695. Springer, 2022. 2

[53] Shangshu Yu, Cheng Wang, Yitai Lin, Chenglu Wen, Ming Cheng, and Guosheng Hu. Stcloc: Deep lidar localization with spatio-temporal constraints. *IEEE TITS*, 24(1):489–500, 2022. 4

[54] 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. *PR*, 128:108685, 2022. 2, 4

[55] Hui Zhou, Xinge Zhu, Xiao Song, Yuexin Ma, Zhe Wang, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. *CVPR*, 2021. 2