1-Day Learning, 1-Year Localization: Long-Term LiDAR Localization Using Scan Context Image

Giseop Kim, Byungjae Park, and Ayoung Kim

***Abstract*—In this letter, we present a long-term localization method that effectively exploits the structural information of an environment via an image format. The proposed method presents a robust year-round localization performance even when learned in just a single day. The proposed localizer learns a point cloud descriptor, named Scan Context Image (SCI), and performs robot localization on a grid map by formulating the place recognition problem as place classification using a convolutional neural network. Our method is faster than existing methods proposed for place recognition because it avoids a pairwise comparison between a query and scans in a database. In addition, we provide thorough validations using publicly available long-term datasets, the NCLT dataset and the Oxford RobotCar dataset, and show that the Scan Context Image (SCI) localization attains consistent performance over a year and outperforms existing methods.**

***Index Terms*—Localization, range sensing, SLAM.**

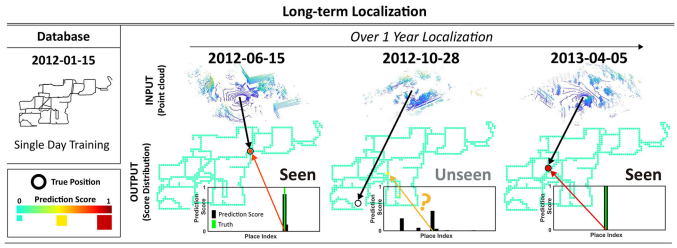


Fig. 1. In this letter, we describe our localization method that learned from data collected on a single day and had consistent performance for over one year. In defining the existence of an unlearned place (i.e., an area the robot has not visited before), the algorithm we employ is capable of handling unseen places, which appear during long-term navigation.

I. Introduction

LOCALIZATION in a coarse [5] or fine manner [6] is one of the most necessary and basic abilities of a mobile robot. Recently, focus has moved to long-term autonomy (LTA) [7] in order to operate in a real outdoor environment beyond a lab-level static and controlled environment. LTA is particularly important for localization because the appearance of an environment changes over time (e.g., light condition or occlusion), potentially resulting in robot localization failure. Although many methods [7], [8] have been proposed, few agree on a complete visual-based solution to overcome this problem. To accomplish the LTA in changing environments, many approaches [9] have tried to take multi-experiences into a localization framework. These approaches revealed inherent drawbacks because they need to capture various conditions for the same place a priori to increase the size of the database with the number of experiences.

Contrast to visual appearance, the physical structure of a place rarely changes over time. Hence, leveraging structural information that is perceptible within a place has benefits for long-term localization [10] than methods based on appearance only. In that sense, a single time observation using Light Detection and Ranging (LiDAR) could represent a canonical characteristic of a place, eliminating the need for multiple experiences for robust localization. In this line of research, a handcrafted descriptor-based [1], [11] and learning-based [2], [12] method for place recognition over a point cloud has been widely proposed. However, these studies hardly captured the long-term localization requirements, including a slow but massive structural variance (e.g., construction and demolition) and unexpected viewpoint from the road topology change.

Many LiDAR-based, global, coarse localization methods have focused on making a robust descriptor with a strong capability to discriminate between places. The current research on descriptors can be divided into *non-learning* and *learning-based*.

Non-learning based Descriptors: M2DP [1] is a handcrafted descriptor; it projects a point cloud into multiple planes, whose normal directions are manually determined. M2DP showed that, unlike previously proposed methods such as histogram-based [13], it can effectively perform place recognition even in an outdoor context with a noisy point cloud. Inspired by the concept of 3D isovists [14] used in urban design, Scan Context (SC) [11] has shown that extracting only the highest points of a visible point cloud outperforms others including M2DP. Recently, a

Manuscript received September 6, 2018; accepted January 21, 2019. Date of publication February 4, 2019; date of current version February 28, 2019. This letter was recommended for publication by Associate Editor N. Sunderhauf and Editor C. Stachniss upon evaluation of the reviewers' comments. This work was supported in part by the Korea Agency for Infrastructure Technology Advancement (KAIA) through the Ministry of Land, Infrastructure and Transport of Korea under Grant 19CTAP-C142170-02, and in part by the [High-Definition Map Based Precise Vehicle Localization Using Cameras and LIDARs] project funded by Naver Labs Corporation. *(Corresponding author: Ayoung Kim.)*

G. Kim and A. Kim are with the Department of Civil and Environmental Engineering, KAIST, Daejeon 34141, South Korea (e-mail: paulgkim@kaist.ac.kr; ayoungk@kaist.ac.kr).

B. Park is with the Intelligent Robot System Research Group, ETRI, Daejeon 34129, South Korea (e-mail: bjp@etri.re.kr).

This letter has supplemental downloadable multimedia material available at http://ieeexplore.ieee.org, provided by the authors. The Supplementary Materials contain a video illustrating three main contents of the paper: the overall pipeline for long-term LiDAR localization using the proposed descriptor (called Scan Context Image), entropy-based novel place detection, and the results from extensive datasets over a year. This material is 18.6 MB in size.

Digital Object Identifier 10.1109/LRA.2019.2897340

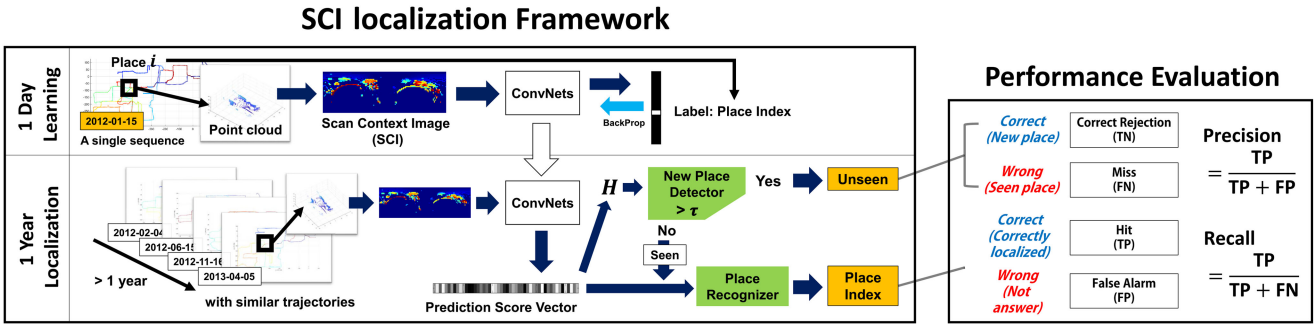


Fig. 2. Overall pipeline of the SCI localization and performance evaluation.

study on using intensity instead of structural information [15] was released.

Learning-based Descriptors: Recently, Uy and Lee proposed a network called PointNetVLAD [2], which combined PointNet [16] and NetVLAD [17] to generate a point cloud descriptor with achieving permutation invariance. They validated the network provided enough generality; that is, the network taught with the Oxford RobotCar [4] dataset works well for scans obtained by other robots in different environments. Unlike PointNetVLAD's borrowing metric learning, SegMap [12] brought an encoder-decoder system to make a descriptor into its simultaneous localization and mapping (SLAM) framework so as to enable both efficient reconstruction and robust loop-closure detection.

Although many methods have been proposed, there are few empirical studies showing the effectiveness of LiDAR descriptors on long-term localization capability in urban areas. Several works have attempted to address long-term accurate (centimeter-level) localization within a prior LiDAR point cloud map using Bayesian filtering. Maddern *et al.* [18] proposed a 2.5D rasterized, image-based, GPU-accelerated search. Recently, Withers and Newman [19] introduced a point-wise rejection method for handling scene changes and avoiding false localization. This kind of work usually focuses on how to model structural scene changes within a Bayesian framework, rather than the global place retrieval capability of large-scale localization.

Differing from the aforementioned descriptor-based category, an end-to-end localizer that infers a robot's pose directly using deep learning has nowadays been gaining attentions. This formulates the localization problem as 6D pose regression [6] or a course place classification [20]. Compared to these image-based localizers [6], [20], however, few direct methods accept a LiDAR point cloud as input have been proposed.

In this letter, we present a convolutional neural network (CNN)-based, end-to-end localization framework (Fig. 1). The proposed localizer is based on a point cloud descriptor called Scan Context Image (SCI) that effectively summarizes the unstructured point cloud into a structured form. We validate that only a single experience is sufficient to demonstrate the effectiveness of our method on the tested datasets. Refer to the video sciloc.mp4 in supplementary material as well.

Our approach is similar to PlaNet [20] in that we also consider a place as a class and formulate a localization task as a classification task using a CNN. Unlike PlaNet, which provides a rough location scope that cannot be used for mobile robot navigation, we guarantee successful localization within a few meters on a map of a several hundred or thousand meters. Our contribution points are summarized below.

- We introduce the classification-based place retrieval pipeline using an image-shaped point cloud descriptor called SCI.
- To alleviate false alarms during long-term localization, we propose an entropy-based detection module for unseen places.
- Evaluations for two long-term datasets (the NCLT dataset [3] and the Oxford RobotCar dataset [4]) are provided. The proposed method localizes a path of over 10 km for over a year and covers all seasons and severe structural and viewpoint changes.

II. SCI Generation and Training

In this section, we introduce a 3D point cloud descriptor in an image format named SCI. Because SCI is created from a point cloud descriptor, Scan Context (SC), we first provide a brief review of SC. We refer readers to [11] for more detail. Next, we introduce a deep learning based classification method for long-term localization. The overall pipeline, from the training to the procedure of the localization, is depicted in Fig. 2.

A. A Brief Review of Scan Context (SC)

Scan Context (SC) takes a 3D point cloud as an input and divides its planar-surrounding regions within a *maximum range* into *sectors* and *rings*, which are segments divided into azimuthal and radial directions, respectively. The intersection of a sector and a ring is called a *bin*. SC only takes the highest point value from each bin and arranges them into a 2D matrix form, through which the internal arrangement of bins is preserved. The top part of Fig. 3 shows the making process of SC from a raw point cloud. In this letter, the number of rings, the number of sectors, and the maximum range are 40, 120, and 80 m, respectively.

B. Scan Context Image (SCI)

The previously defined SC is a single-channel matrix that encapsulates robust structural information (i.e., the maximum

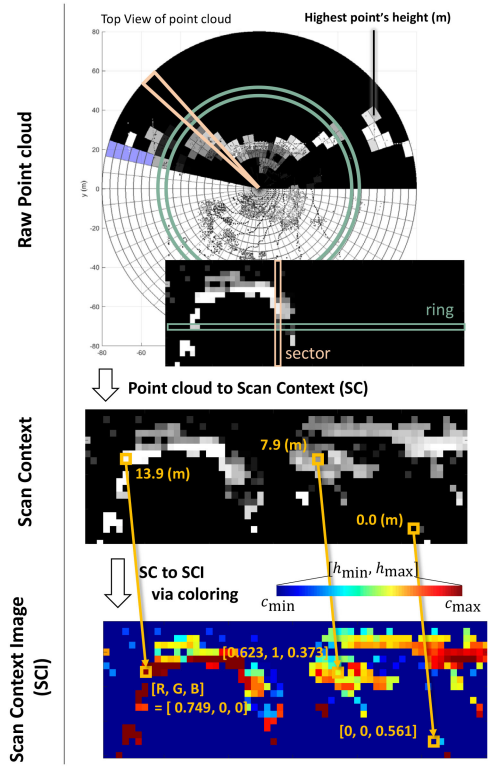


Fig. 3. Scan Context Image (SCI) generation from a raw point cloud and conversion to a 3-channel SCI.

height of points) around a scene. Although SC is already in an image-like form, we normalize it and convert this into three channels to be suitable as input for CNN. When converting, the structural height out of $[h_{min}, h_{max}]$ is saturated. In this work, we use a jet colormap, which has a larger variance than sequential colormaps, and the mapping function (f_c) with $h_{min} = 0\,\mathrm{m}$ and $h_{max} = 15\,\mathrm{m}$. In doing so, we empirically validate a small improvement compared with training with one channel image. The proposed SCI increases the discriminative power to more than that of SC and is also a more suitable format for inputting a CNN. This process is visualized in Fig. 3. We note that further investigation on network tuning for monochrome images or colormap selections may improve the localization performance.

C. Location Definition

Because we formulate the outdoor robot localization problem as a classification issue, we use a classification network. We first divide the region, which is covered in the training sequence, into equal-sized (e.g., 10 m by 10 m) grid cells on the x-y plane and assign a different index to each cell. A single cell represents a single place. Then, all SCIs acquired in a cell are used to train a CNN with its class label; the label is a one-hot encoded vector of the corresponding place index. The label dimension is equal to the total number of places because we consider each place as a unique class. Then, the network is trained with categorical cross-entropy loss, which is generally used to train a classification network.

TABLE I
A SIMPLE NETWORK STRUCTURE WE USED. BN AND MP ARE BATCH NORMALIZATION AND MAX POOLING, RESPECTIVELY, AND WE USED 2×2 POOLING SIZE. THE NUMBER IN THE CONV() AND FULLYCONNECTED() LAYER MEANS THE NUMBER OF FILTERS AND THE NUMBER OF NODES, RESPECTIVELY. 5×5 FILTERS WERE USED FOR ALL CONVNETS AND 0.7 (30% REMAINS) DROPOUTS WERE APPLIED FOR ALL DROPOUT LAYERS. N IS THE NUMBER OF TOTAL PLACES. WE TRAINED THE NETWORK WITH 64 OF BATCH SIZE AND USING ADAM OPTIMIZER WITH DEFAULT PARAMETERS (LEARNING RATE $= 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$)

Input	(batch_size, 40, 120, 3)
Conv1	BN(MP(ReLU(Conv(64, Input))))
Conv2	BN(MP(ReLU(Conv(128, Conv1))))
Conv3	Flatten(MP(ReLU(Conv(256, Conv2))))
FC1	FullyConnected(64, Dropout(Conv3))
FC2	softmax(FullyConnected(N, Dropout(FC1)))
Output	(batch_size, N)

D. Network Selection

Any CNN structure can be used to construct the proposed localization system, but we use a LeNet [21]-like network with regularization to demonstrate that our method works well with a simple network. A detailed structure and parameters are shown in Table I.

E. N-Way SCI Augmentation

We propose N-way augmentation to achieve the viewpoint invariance to tackle potential viewpoint variance in the long-term localization. Because the column order of SCI indicates the heading of a robot, viewpoint variation via synthetic SCI in the training phase is fairly simple (e.g., the column-shift). Here, N is the number of 360 degrees divided by a constant interval. An example of two-way augmentation (what we call reverse augmentation) is visualized in Fig. 7.

III. SCI LOCALIZATION

A. Un-Learned Place Detection

In the LTA scenario, the robot may visit a new place that is, not in the training set. Therefore, detection and proper handling of this unlearned location are critical in LTA. Prior to the localization module, we first identify whether a query place is a new place or not (i.e., a query point cloud is from a new place or not) to avoid false localization. We call the new place, *unseen place*, and an existing place in the training sequence, *seen place*. This task can be considered in unknown-unknown class detection [22], which has highly attracted computer vision researchers for constructing more robust classification system. For example, Dropout Variational Inference [23] can approximately provide a class probability but requires multiple predictions, which is time-consuming and thus may be difficult for real-time robot localization.

Unlike this costly method, we propose a way to directly use the entropy of the output vector (without dropout at the test time) from the network. Note that we do not aim to approximate each class probability; instead, we rather focus on

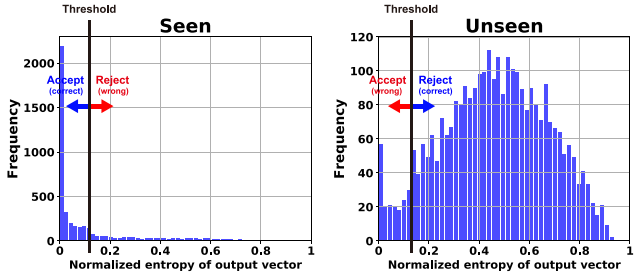


Fig. 4. The example of the distribution of entropies of prediction score vectors for seen and unseen. This example is from the test sequence 2015-06-12-08-52-55 of the Oxford RobotCar dataset. Left: The histogram of predictions' entropies from *seen* places usually has small values. Right: The histogram of entropies from *unseen (new)* places has a large variance, and there are higher entropies than the case of seen places.

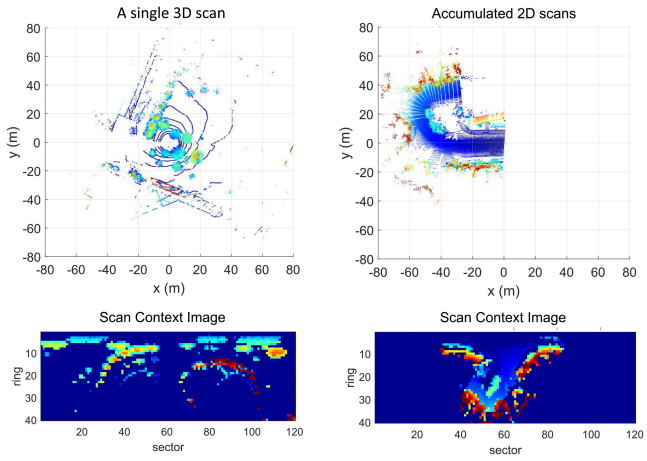


Fig. 5. A visualization of a point cloud and the associated SCI for a 3D scan from the NCLT dataset (left) and a submap, which accumulates 2D scans, from the Oxford RobotCar dataset (right).

identifying whether the query is seen or unseen. As will be shown in Section V, this entropy of the output vector has a substantially stronger discriminative performance than traditional pairwise distance-based thresholding (e.g., the Euclidean distance) for unseen place handling. Specifically, we use the following normalized entropy of the prediction score vector $H(\mathbf{p}) = -\frac{1}{H_{max}} \sum_{i=1}^{N} p_i \log_2 p_i$, where p_i is the ith element of the vector $\mathbf{p}$ and H_{max}, which is the maximum entropy of a N dimensional vector, exists for the normalization.

If the entropy of the prediction vector is higher than a given threshold τ (user parameter), it is considered as a new place and rejected without localization. On the other hand, we only perform localization in the following step for images classified as *seen*. Fig. 4 shows an example of the distribution of entropies from seen and unseen places of the sequence.

B. Localization

If the query is considered to be seen (i.e., the point cloud is obtained from seen places), then localization is performed using the prediction score vector. The index of this vector's element, which has the largest score, is concluded as the current place. More generally, we would say the localization is successful if the ground truth index of a query place belongs to a set of top N indexes whose scores are in a larger order in the network's prediction score vector.

IV. Experiments

A. Benchmark Datasets

We used two long-term datasets that are publicly available in the robotics community: the NCLT [3] and Oxford RobotCar [4] datasets. Both datasets provide multiple sequences along similar trajectories over a year and include various environmental changes for the same places.

The NCLT dataset provides 3D LiDAR scans and each scan is directly encoded into the SCI as described on the left of Fig. 5. For the Oxford RobotCar dataset, we used sequences with the *full* trajectory of nearly 10 km. This dataset has no 3D LiDAR, and 2D LiDAR were mounted perpendicularly to the vehicle's moving direction. Thus we accumulated 2D scans along a local trajectory for enough length as visualized on the right of Fig. 5. We set an accumulation length (or a window size) equal to the *maximum range*, which is the parameter of an SCI. We use the visual odometry the dataset provides for stacking scans. By stacking them, we use the relative motion between a previous scan and a recent scan is placed at the origin. In doing so, we can make a 3D point cloud (or a submap) with enough information to make an SCI. We considered the global coordinate available from the *ins.csv* file as the ground truth of each place. Only places with a reliable inertial navigation system (INS) status (i.e., INS_SOLUTION_GOOD) are used for training and tests.

The size of a grid cell for the main analysis (Fig. 6 and Fig. 8) is 10 m by 10 m. For this grid map resolution, the NCLT and Oxford RobotCar datasets have 579 and 700 places in their training sequences, respectively. The places from the NCLT and Oxford RobotCar datasets are trained with only a single sequence, and then the localization is evaluated for the following 10 sequences, which covers over a year. For training, an SCI and descriptors of comparison methods are sampled for every 1 m. The test sequences are also evaluated by sampling every 1 m. Details about the training and test sequences of the NCLT and Oxford RobotCar datasets are summarized in Table II. The seen and unseen rows indicates the number of queries from seen and unseen places.

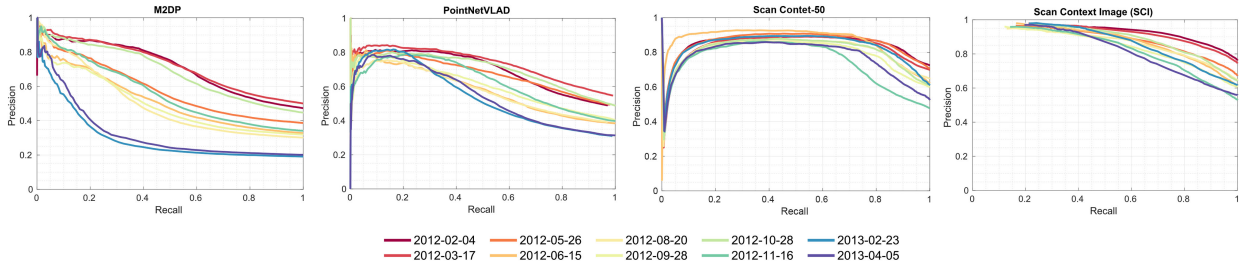
B. Comparison Methods

We compare our method, SCI-localization, with three state-of-the-art handcrafted and learning-based point cloud descriptors: M2DP [1], Scan Context [11], and PointNetVLAD [2]. For a fair comparison, both methods construct a database using only descriptors from a single sequence and are compared to a query descriptor from test sequences for over a year. The nearest candidate's index is considered a query's location.

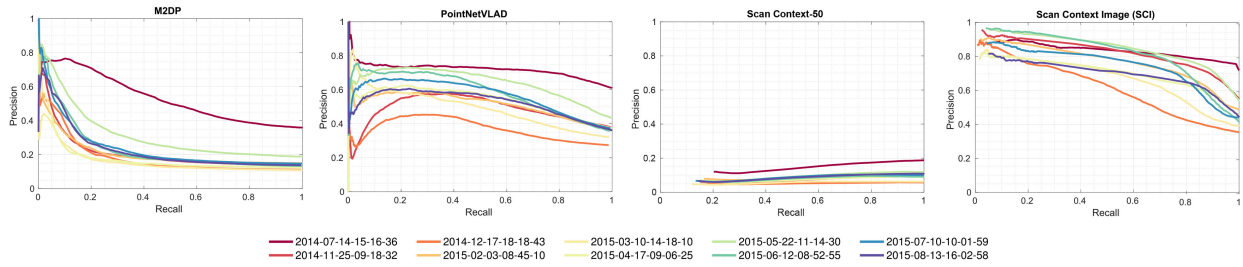
M2DP is a lightweight point cloud descriptor designed for loop-closure detection. The core idea of M2DP is projecting a 3D point cloud into multiple 2D planes. We used the same parameters and procedure as the original authors by using

TABLE II
SUMMARY OF DATASETS

Dataset	Train Seq.		Test Seq.									
NCLT	2012-01-15		2012-02-04	2012-03-17	2012-05-26	2012-06-15	2012-08-20	2012-09-28	2012-10-28	2012-11-16	2013-02-23	2013-04-05
	579 places	Seen	5170	5449	5533	3321	5146	4626	4623	3575	4114	3341
		Unseen	441	428	773	742	835	919	1034	1290	1095	1162
Oxford Robot Car	2014-07-14 -14-49-50		2014-07-14 -15-16-36	2014-11-25 -09-18-32	2014-12-17 -18-18-43	2015-02-03 -08-45-10	2015-03-10 -14-18-10	2015-04-17 -09-06-25	2015-05-22 -11-14-30	2015-06-12 -08-52-55	2015-07-10 -10-01-59	2015-08-13 -16-02-58
	700 places	Seen	4079	5484	3926	5657	5106	5485	5664	4321	4872	5043
		Unseen	414	2066	1769	2163	2782	2571	2232	3062	2585	2466



(a) NCLT dataset



(b) Oxford RobotCar Dataset

Fig. 6. Precision-recall curves for two long-term datasets, NCLT and Oxford RobotCar dataset.

the open-source of M2DP,[1] and we acquire a 192-dimension descriptor from a point cloud.

Scan Context-50 exploits a similar descriptor as the SCI, but in a non-learning based way. SC used the column-wise comparison to calculate the distance between a query and a candidate. To clearly validate the learning effect, we compare SCI against Scan Context-50 in [11], which is the method that takes 50 candidates for the pairwise comparison.

PointNetVLAD is a combination of PointNet [16] and NetVLAD [17], so it can directly consume a point cloud without any reformulation such as projection or voxelization. We applied preprocessing similar to the original paper; a ground-removed point cloud within a $[-25$ m, 25 m] cubic window[2] is filtered into the constant number (4096) points and rescaled into a $[-1, 1]$ range with a zero mean. This processed point cloud is fed to the network, and, finally, we get a 256-dimensional descriptor. We used the pretrained model (refined version) the authors provided.[3]

[1]https://github.com/LiHeUA/M2DP

[2]We empirically checked that an increased window size ($[-80$ m, 80 m]) deteriorated their performance for PointNetVLAD

[3]https://github.com/mikacuy/pointnetvlad

V. EVALUATION RESULTS

In this section, we provide intensive analyses to validate the effectiveness and robustness of the proposed method. The detailed information of training data and test sequences are described in Table II. The test sequences of each dataset were possibly selected to include at least one sequence per month to cover various conditions over the entire year. The number of samples in rows of seen and unseen places in Table II are sampled per every 1 m and are used for the evaluation.

A. Precision-Recall Curve

We first evaluate the general performance using the precision-recall curve for both datasets throughout the long-term operation (Fig. 6) by varying the entropy threshold τ for our method and the pairwise distance threshold (the Euclidean distance for M2DP, PointNetVLAD and Scan Context-50 uses its proposed distance (6) in [11]) for comparison methods for unseen place handling. The evaluation procedure is depicted in the right side in Fig. 2. If a query is considered as a seen place, we considered the localization to be correct if the index of the largest element of the network's output vector (for our method) or nearest

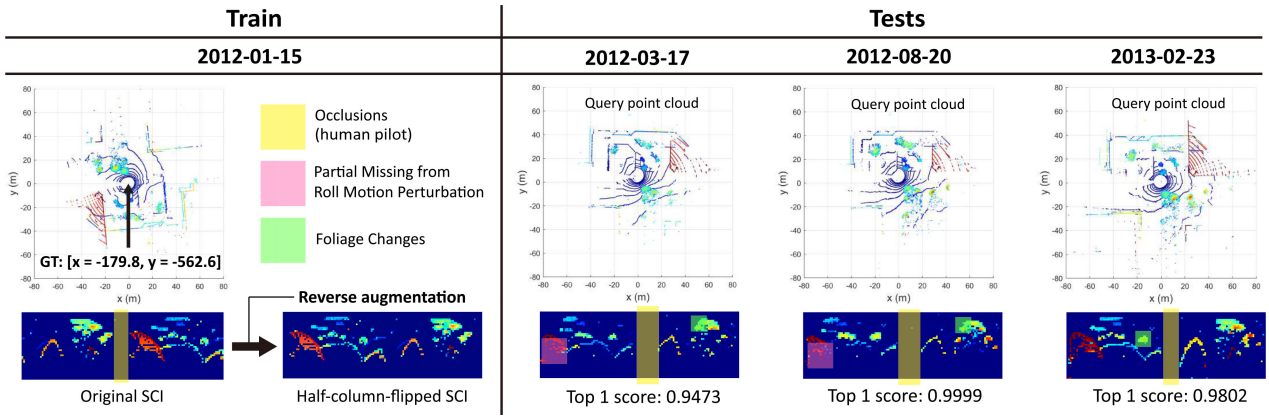


Fig. 7. Robustness to non-structural changes. Despite challenging factors (e.g., viewpoint changes, occlusions, and foliage), the proposed method successfully found its location with a high score for over hundreds of places over a year.

descriptor's place index (for M2DP, Scan Context-50 and PointNetVLAD) is the same as the answer (i.e., top 1 performance).

The learning-based descriptor, PointNetVLAD, outperformed the handcrafted method, M2DP, by a large margin. However, PointNetVLAD revealed lower performance than our method in terms of long-term localization performance. Moreover, the proposed SCI localization method presented less fluctuation than others in performance over time. For Scan Context-50, the column-wise matching function of SC assumes a surround-capturing LiDAR; thus it showed the poor performance at the Oxford RobotCar dataset, which used 2D LiDAR. SCI decreases performance over time but still performs better than other methods.

B. Retrieval Capability

For large-scale localization, not only the top 1, but taking more candidates (e.g., top 5 and top 25) would also be meaningful. Therefore, we provide a more in-depth analysis of the retrieval power of each method. We extended the criteria of the correct answer to the top 5 and top 25 candidates to investigate by how much the performance of each method increased.

Fig. 8 shows a comparison of overall performance. We plot the area under the curve (AUC) of the precision-recall curve of each sequence as a measure. The closer the AUC is to 1, the more perfect the localization. The AUC values of all methods have increased by allowing the top 25 candidates but our top 1 performance is comparable or better than others' top 25 performance.

C. Long-Term Robustness

In this subsection, we investigate two types of environmental changes; *non-structural* and *structural* changes.

Non-structural changes: Although the structural information of a scene is naturally robust for LTA, there are a few challenging factors that make a point cloud different from the experience. Fig. 7 visualizes the examples of challenging cases and their corresponding SCIs from the NCLT dataset. The NCLT dataset always had partial and varying occlusion due to an accompanying human pilot. In addition, the Segway-like robot used in the

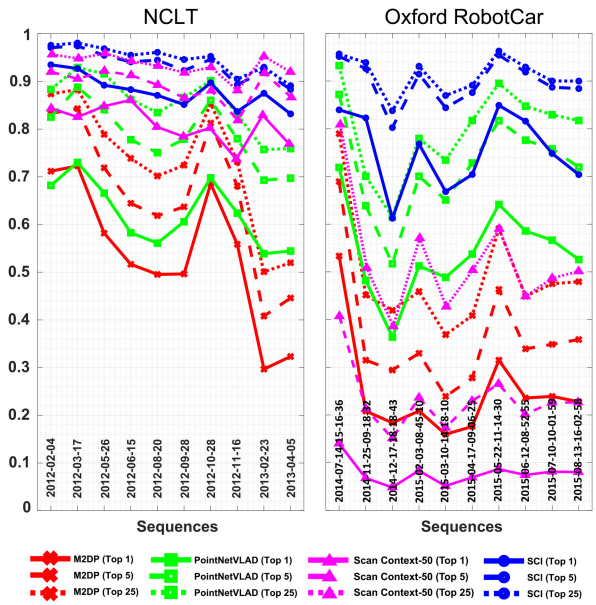
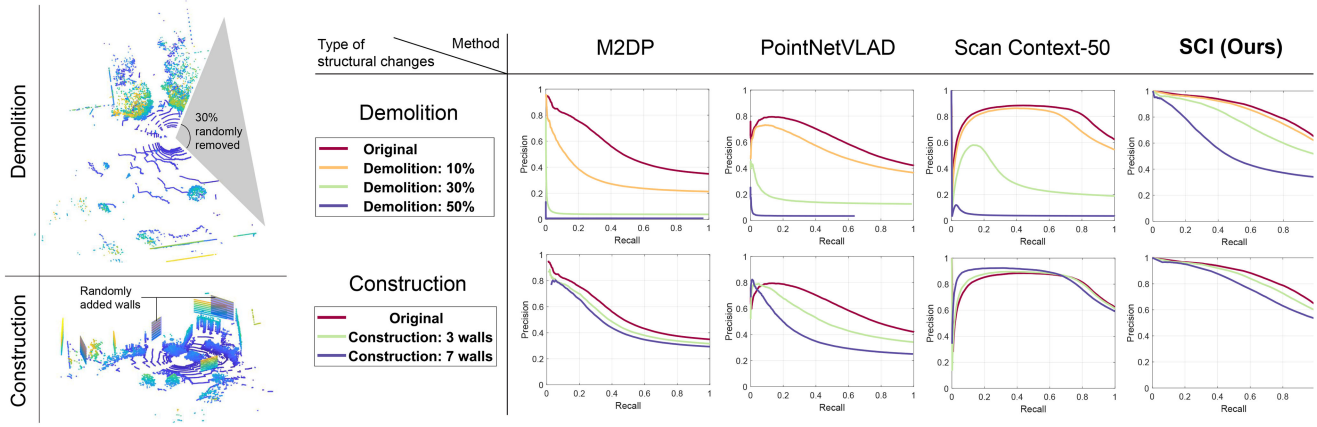


Fig. 8. AUC performance changes over time for different criteria of success localization.

NCLT dataset had an unstable roll motion compared to a car platform, and partial structures such as foliage changed over time. Despite these challenging factors, our method successfully localized a query because the SCI preserved the internal relations of the egocentric scene structure, unlike the other descriptors that lost the original scene's structural shapes.

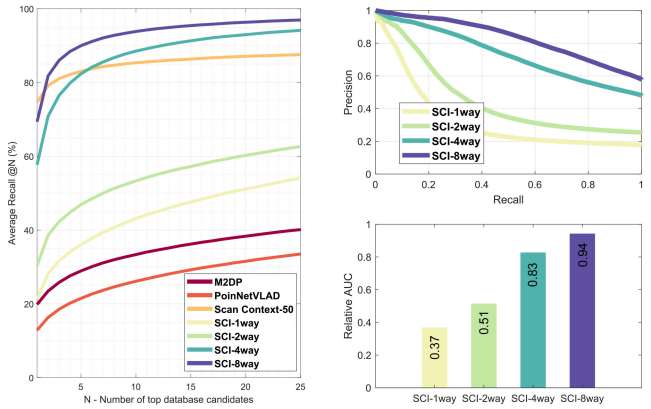
Structural changes: The long-term structural challenges arise from structural experience (i.e., structures that existed at the training sequence) that may have disappeared (*demolition*) or been newly constructed (*construction*) over time. For validation, we removed points within a randomly selected sector or added new randomly generated wall-shaped points, as in Fig. 9(a). Because the M2DP is based on point projection, it is less affected by the appearance of structures but is vulnerable to demolition. PointNetVLAD was sensitive to the removal and addition of points as it uses only 4096 points as the input. Although SC utilizes descriptors very similar to SCI, we verified that ConvNet



(a) Examples of structural changes

(b) Precision-recall curves for demolition and construction. Each line is a mean for 10 test sequences.

Fig. 9. Robustness to structural changes on the test sequences of the NCLT dataset.



(a) Retrieval performance of each method for random viewpoint changes

(b) Precision-recall curves for each N-way augmentation.

Fig. 10. Robustness to random viewpoint changes on the test sequences of the NCLT dataset. Each line is a mean for 10 test sequences.

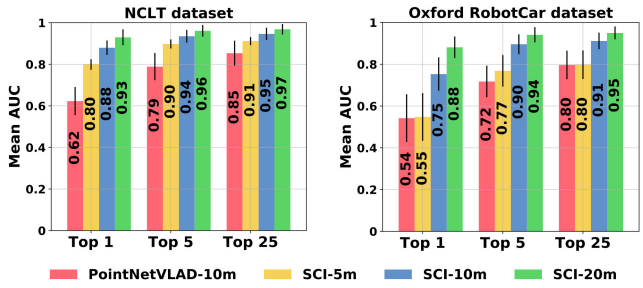


Fig. 11. Performances with respect to different grid cell sizes. The vertical black line pinned at each bar represents the standard deviation of all (10 for each dataset) test sequences.

based unseen place detection and classification-based retrieval are superior in localization performance.

D. Robustness to Viewpoint Changes

Arbitrary viewpoint variation inevitably occurs during the long-term localization. In this section, we examine the effect of N-way augmentation on the viewpoint change robustness by increasing N in the training phase. Unfortunately, the number of queries in original datasets are rather small for testing various viewpoint cases. Instead, we tested the trained network by randomly rotating a query point cloud's heading.

As seen in Fig. 10(a), existing descriptors, including the baseline of SCI-localization without N-way augmentation, failed to localize under arbitrary viewpoint changes. We empirically validated that using four-way augmentation could yield sufficient robustness to the viewpoint variation. In doing so, the general performance is also preserved as in Fig. 10(b). The bottom of Fig. 10(b) presents the AUC relative to the original performance in Fig. 6(a) without the intentional heading rotation.

E. Grid Cell Size

We also evaluated localization performance by considering different grid sizes to identify whether finer localization is possible. We conducted the same experiment as in Section V-A but with different grid cell sizes. The grid cells are finer (5 m by 5 m) and coarser (20 m by 20 m). The number of output nodes in the SCI localization network was reset to the new total number of places and retrained. The results for different grid cell sizes are shown in Fig. 11. Despite increased labels of over 1000 places for 5 m^2 resolution and a slight decrease in performance, our method still presented higher performance with lower variance than PointNetVLAD with a 10 m resolution for both datasets.

F. Runtime Evaluation

Another strength of the proposed method is the lightweight implementation. For the runtime comparison in Table III, all implementations used Matlab except for a few parts that used the deep network by using Python at NVIDIA GTX 1080Ti with a test batch size of one.

PointNetVLAD showed the longest time for a generation, requiring both preprocessing (e.g., ground removal and filtering) and passing the network. However, both PointNetVLAD and M2DP are lightweight descriptors, and thus find a nearest in the database quickly (i.e., short retrieval time). Scan Context-

TABLE III
AVERAGE TIME COST FOR EACH METHOD. THE COMPARISON IS CONDUCTED ON THE 2013-04-05 OF THE NCLT DATASET

Method	Descriptor Generation (sec)	Retrieval (sec)	Total (sec)
SCI	0.0434	0.0047	0.0481
SC-50	0.0413	0.4633	0.5046
PNVLAD	0.1374	0.0220	0.1594
M2DP	0.0758	0.0195	0.0953

50 is the slowest for retrieval, as reported in [11]. Unlike other methods, the SCI's retrieval time is the shortest because SCI-localization directly obtains scores for N places via a single pass through the network rather than a pairwise comparison with the whole database.

VI. CONCLUSION

We presented a global end-to-end localization method based on deep learning by learning the point cloud descriptor, SCI. The proposed SCI with a classification network is more robust for long-term robot localization than other state-of-the-art pairwise score-based place retrieval methods [1], [2], [11]. Our method showed a consistent, and state-of-the-art performance for over a year even though the network was trained using only a single sequence. Due to its robust and global performance, we expect the proposed framework could also be used for the kidnapped robot problem or an initialization for the finer localization as ICP [11].

In the future work, we plan to extend our work on how to flexibly add new places to the existing network's knowledge without forgetting and in order to avoid whole learning again. We will also investigate an end-to-end method that calculates a global 6D pose by using the coarse localization result from the SCI-localization framework.

REFERENCES

[1] L. He, X. Wang, and H. Zhang, "M2DP: A novel 3D point cloud descriptor and its application in loop closure detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 231–237.
[2] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4470–4479.
[3] N. Carlevaris-Bianco, A. K. Ushani, and R. M. Eustice, "University of michigan north campus long-term vision and lidar dataset," *Int. J. Robot. Res.*, vol. 35, no. 9, pp. 1023–1035, 2016.
[4] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *Int. J. Robot. Res.*, vol. 36, no. 1, pp. 3–15, 2017.
[5] D. Galvez-Lpez and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Trans. Robot.*, vol. 28, no. 5, pp. 1188–1197, Oct. 2012.
[6] A. Kendall, M. Grimes, and R. Cipolla, "PoseNet: A convolutional network for real-time 6-DOF camera relocalization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2938–2946.
[7] J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler, "Semantic visual localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6896–6906.
[8] H. Porav, W. Maddern, and P. Newman, "Adversarial training for adverse conditions: Robust metric localisation using appearance transfer," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 1011–1018.
[9] W. Churchill and P. Newman, "Experience-based navigation for long-term localisation," *Int. J. Robot. Res.*, vol. 32, no. 14, pp. 1645–1661, 2013.
[10] Y. Ye, T. Cieslewski, A. Loquercio, and D. Scaramuzza, "Place recognition in semi-dense maps: Geometric and learning-based approaches," in *Proc. Brit. Mach. Vis. Conf.*, 2017, pp. 72.1–72.13.
[11] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2018, pp. 4802–4809.
[12] R. Dubé, A. Cramariuc, D. Dugas, J. Nieto, R. Siegwart, and C. Cadena, "SegMap: 3D segment mapping using data-driven descriptors," in *Proc. Robot., Sci. Syst. Conf.*, 2018.
[13] W. Wohlkinger and M. Vincze, "Ensemble of shape functions for 3D object classification," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2011, pp. 2987–2992.
[14] M. L. Benedikt, "To take hold of space: Isovists and isovist fields," *Env. Planning B: Planning and Des.*, vol. 6, no. 1, pp. 47–65, 1979.
[15] K. Cop, P. V. K. Borges, and R. Dubé, "DELIGHT: An efficient descriptor for global localisation using LiDAR intensities," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 3653–3660.
[16] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 77–85.
[17] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 5297–5307.
[18] W. Maddern, G. Pascoe, and P. Newman, "Leveraging experience for large-scale LIDAR localisation in changing cities," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2015, pp. 1684–1691.
[19] D. Withers and P. Newman, "Modelling scene change for large-scale long term laser localisation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2017, pp. 6233–6239.
[20] T. Weyand, I. Kostrikov, and J. Philbin, "PlaNet - photo geolocation with convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 37–55.
[21] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[22] D. Miller, L. Nicholson, F. Dayoub, and N. Sünderhauf, "Dropout Sampling for robust object detection in open-set conditions," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2018, pp. 3243–3249.
[23] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?," in *Proc. Adv. Neural Inf. Process. Syst. Conf.*, 2017, pp. 5574–5584.