2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

SpoxelNet: Spherical Voxel-based Deep Place Recognition for 3D Point Clouds of Crowded Indoor Spaces

Min Young Chang[1], Suyong Yeon[2], Soohyun Ryu[2] and Donghwan Lee[2]

***Abstract*— With its essential role in achieving full autonomy of robot navigation, place recognition has been widely studied with various approaches. Recently, numerous point cloud-based methods with deep learning implementation have been proposed with promising results for their application in outdoor environments. However, their performances are not as promising in indoor spaces because of the high level of occlusion caused by structures and moving objects. In this paper, we propose a point cloud-based place recognition method for crowded indoor spaces. The method consists of voxelizing point clouds in spherical coordinates and defining the occupancy of each voxel in ternary values. We also present SpoxelNet, a neural network architecture that encodes input voxels into global descriptor vectors by extracting the structural features in both fine and coarse scales. It also reinforces its performance in occluded places by concatenating feature vectors from multiple directions. Our method is evaluated in various indoor datasets and outperforms existing methods with a large margin.**

I. Introduction

Accurate place recognition greatly enhances the performances of robot localization and simultaneous localization and mapping (SLAM). In terms of localization, recognizing places in a reference map provides positional information of robots in global coordinates. In a SLAM pipeline, place recognition solves the sensor drift problem by providing loop-closure candidates. With its key role in autonomous navigation, place recognition has been actively researched with different combinations of mathematical models and sensor selections [1]–[4], and its performance greatly improved with the recent application of deep learning and LiDAR sensors [5]–[7]. However, methods that have so far been reported mostly focus on outdoor roads with their application in self-driving cars. With our best knowledge, point cloud-based place recognition in indoor spaces has not been as thoroughly researched, despite its as much necessity for autonomous navigation of mobile service robots.

Point clouds of indoor spaces have characteristic differences from those of outdoor roads. The most prominent difference is the intensity of occlusions. In indoor spaces, robots' proximity to structures creates a great amount of occlusion, which does not happen as much in outdoor roads.

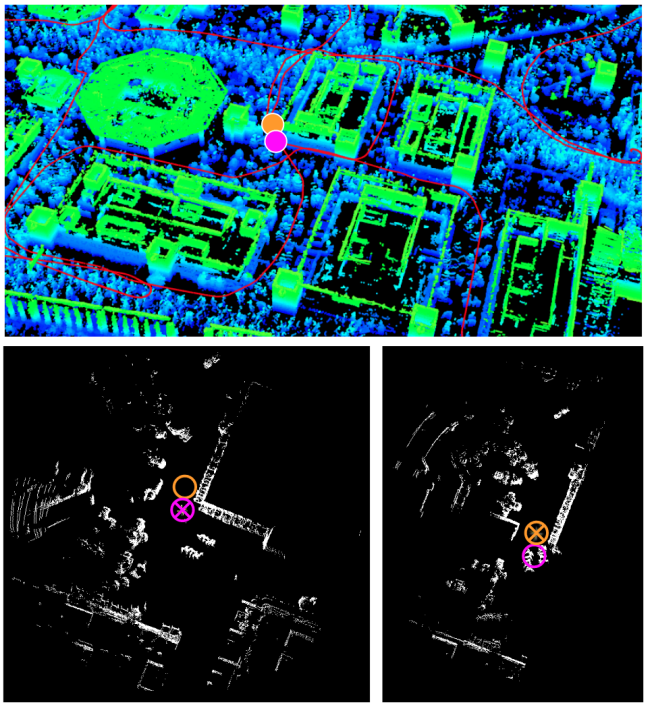

Fig. 1. Example of occlusion in a crowded indoor space. The picture on the top is the combined point cloud of a department store. Points are colored based on their height, and the red line is the trajectory plotted above the point cloud. In the top picture, magenta and orange dots represent the origin of each scan on the bottom pictures. The two scans are only one meter apart from each other, yet appear significantly different, because of the occlusion created by the wall and people.

Compared to outdoor roads, most indoor hallways are much narrower, and the distances between robots and walls are much smaller than those between cars and buildings. If a robot positions very close to a wall, the laser projection of its LiDAR sensor is severely occluded and result in a point cloud that is significantly different from an unoccluded one, as compared in Fig. 1. Another major difference is the amount of occlusion and noise created by moving objects. In contrast to people who can freely approach robots, cars maintain a certain distance from each other and therefore create much less occlusion for LiDAR sensors. Also, in highly crowded buildings, point clouds include a significant number of points scanned from people, as described in Fig. 2. Without a segmentation process, which requires high computational cost, the vast amount of points that belong to moving objects will be included in the structural representation of places and therefore lower the performance of point cloud-based place

[1]Min Young Chang was with Research Intern of NAVER LABS, Seongnam-si, Gyeonggi-do, South Korea, and joining Columbia University as a graduate student. minyoung.chang@columbia.edu

[2]The authors are with Researchers of NAVER LABS, Seongnam-si, Gyeonggi-do, South Korea {suyong.yeon, soohyun.ryu, donghwan.lee}@naverlabs.com

This work was supported by the Institute of Information communications Technology Planning Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2019-0-01309, Development of AI Technology for Guidance of a Mobile Robot to its Goal with Uncertain Maps in Indoor/Outdoor Environments)

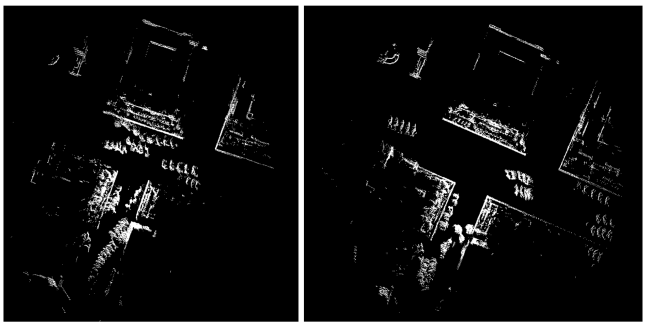

Fig. 2. The two point clouds represent the same place at different time. Although static structures remain the same, moving objects are scanned at different positions and make the two point clouds significantly dissimilar.

recognition methods.

To overcome these problems and successfully execute point cloud-based place recognition in crowded indoor spaces, we present a new method of voxelizing point clouds in a spherical coordinate system and an architecture of deep neural network specifically designed for understanding the spherical voxels. Our proposed spherical voxelization encodes the structural information of the occluded point clouds into voxels, of which the shape agrees with the radial laser projection of LiDAR sensors. The occupancy state of each spherical voxel is defined in ternary values (occupied, unoccupied, and unknown), which minimize the false representation of occluded spaces. Our SpoxelNet receives the spherical voxels as an input, extracts the structural features in both fine and coarse scales, and generate a global descriptor vector by concatenating features from multiple directions. The output global descriptor vector is then compared with a set of global descriptors of a prebuilt reference map, and retrieval of the closest descriptor from the set leads to recognition of the most structurally similar place within the map.

The main contributions of this paper are as follows:

- *Spherical Voxel with Ternary Occupancy Value*. Unlike widely used cartesian voxelization methods, the proposed method voxelizes point clouds in a spherical coordinate system, therefore more accurately represent the point clouds generated by LiDAR sensors. The occupancy state of each spherical voxel is defined in ternary values, which minimizes the misrepresentation of spaces in occluded directions.
- *SpoxelNet*. The network well interprets the structural information of input voxels, and encode it into global feature descriptors with the NetVLAD [8].
- *Validation of the Model's Performance in Various Indoor Areas.* The experiment is designed to validate various aspects of the proposed model as follows: overall performance in various indoor areas, robustness to moving objects, and robustness to temporal variation in structures.

II. Related Works

Point cloud-based place recognition approaches can be largely categorized into two types: non-learning-based and learning-based. Non-learning-based methods extract meaningful information from point clouds through handcrafted feature extraction, such as histogram-based features [9]–[11] and point cloud descriptors [1], [12]–[14]. Despite the promising performances reported from their experiments, the robustness to occlusions and moving objects of each model is not adequately validated. The difficulty of occlusion in place recognition problem is mentioned in [15]. In this paper, Guo *et. al.* present a method that enriches handcrafted features with LiDAR intensity information and report test results that include occluded cases.

Application of neural networks on point clouds for various purposes, such as classification, segmentation, and reconstruction [16]–[18], was extended to learning-based methods for point cloud-based place recognition. SegMap [19] and SegMatch [20] present a localization and place recognition based on data-driven descriptors, which were attained during point cloud reconstruction and classification process. Those methods demonstrate great performances for large scale multi-robot application, but it requires enough number of objects statically present in the environment. Furthermore, descriptors are created based on the objects used during training that the application of methods is limited to similar environments as the dataset [21] used for training. Another examples of learning-based methods are PointNetVLAD [6] and LPD-Net [7]. Combining the PointNet [16] and the NetVLAD [8], PointNetVLAD suggests the first direct application of point clouds to deep place recognition, which is trained in an end-to-end manner. LPD-Net builds upon the success of PointNetVLAD and improves the performance by extracting local features that encompass the spatial distribution of points and local structures. The great performance of each model is thoroughly validated, but the environments of the test datasets [22] are limited to outdoor road areas.

To the best of our knowledge, relatively much fewer papers have specifically focused on place recognition in indoor areas. Sahdev *et. al.* [23] proposed a vision-based place recognition method designed for the localization of mobile robots in indoor spaces. However, the method concentrates on solving the illumination-variation problem of vision-based methods, not the problem of occlusions in indoor spaces. Another example is the 3D laser-based method for place recognition in dynamic indoor environments, suggested by Zhuang *et. al.* [24]. This method retrieves place recognition candidates through handcrafted spatial features of point clouds, and confirm it with speeded-up robust features (SURF) of angle-bearing images created from the point clouds. The method demonstrates its robustness to moving objects, but the test result does not mention its performances in heavily occluded cases.

Our SpoxelNet is designed for crowded indoor spaces, in which the aforementioned approaches may suffer performance degradation because of the difference between point

clouds in indoor spaces and outdoor roads. Our method encompasses the occlusive nature of indoor spaces and is resistant to the noise created by moving objects. Furthermore, our model is tested in a great variety of indoor spaces, each with different structural characteristics.

III. Proposed Methods

This section explains how to voxelize point clouds in a spherical coordinate system and define each voxel in ternary values. Then, each and every part of the network architecture is introduced.

A. Spherical Voxelization

Point clouds are transformed into a constant dimension of voxels as an input of the SpoxelNet. In this process, spherical voxelization is implemented instead of the cartesian voxelization, which is widely used for various purposes [17], [25], [26]. Cartesian voxels have an unvarying size and it does not encompass the sparsity of laser projection that is proportional to the distance. In contrast, spherical voxels have radially varying sizes, and this characteristic enables the method to more precisely encode each place's structural information of point clouds into voxels [27].

Each voxel ($\mathcal{V}$) is defined based on a spherical coordinate system, *i.e.* $\mathcal{V}(\rho_i, \theta_j, \phi_k)$, representing radial distance, azimuthal angle, and polar angle respectively. In this definition, $i \in [1, N_\rho], j \in [1, N_\theta], k \in [1, N_\phi]$, where N_ρ, N_θ, and N_ϕ each represents the number of divisions in the corresponding coordinate. The values of N_ρ, N_θ, N_ϕ, $\Delta\rho$, $\Delta\theta$, and $\Delta\phi$ in Fig. 3 need to stay constant once set in the beginning of the process, in order to ensure consistency in the coverage of the voxel representation.

B. Ternary Occupancy Value

The proportion of the space beyond the occlusion in indoor spaces is significant. With a binary classification (occupied, unoccupied), the occluded spaces gets classified as unoccupied, and result in misleading structural representation. Therefore, the system simply defines the occupancy of each voxel in ternary values: 0 for unoccupied, 0.5 for unknown, and 1 for occupied (hit-voxels). Along the ρ coordinate, 0 is assigned to every voxel before the hit-voxel, 1 for the hit-voxel, and 0.5 for the ones behind, *i.e.* $\{\mathcal{V}(\rho_1, \theta_j, \phi_k), ...\mathcal{V}(\rho_{hit-1}, \theta_j, \phi_k)\} = 0$, $\mathcal{V}(\rho_{hit}, \theta_j, \phi_k) = 1$, and $\{\mathcal{V}(\rho_{hit+1}, \theta_j, \phi_k), ...\mathcal{V}(\rho_{N_\rho}, \theta_j, \phi_k)\} = 0.5$. If a point cloud is composed of multiple scans, more than one hit-voxels in a single row of ρ coordinate may exist. In this case, the voxels in between the two hit-voxels get 0 assigned instead of 0.5, because it is proven that the space is empty. For example, if there are two hit-voxels in a single row of ρ coordinate, $\{\mathcal{V}(\rho_{hit_1}, \theta_j, \phi_k), ...\mathcal{V}(\rho_{hit_2-1}, \theta_j, \phi_k)\} = 0$ and $\{\mathcal{V}(\rho_{hit_2+1}, \theta_j, \phi_k), ...\mathcal{V}(\rho_{N_\rho}, \theta_j, \phi_k)\} = 0.5$, given that $hit_1 < hit_2$.

C. Network Architecture

SpoxelNet extracts a global descriptor vector that encodes the structural features of a voxelized point cloud. The more similar the structures of the point clouds are, the smaller the distance between the corresponding global descriptor vectors. With a list of global descriptor vectors, place recognition can be executed by finding a vector that is closest to the input vector or finding multiple vectors that are located within a pre-set distance threshold from the input vector.

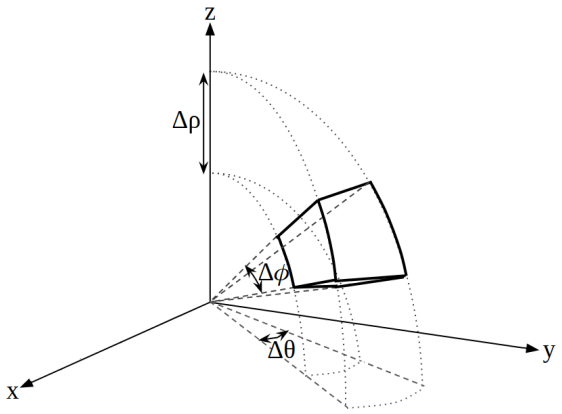

Fig. 3. Spherical Voxelization

As shown in Fig. 4, our model has two different modules that receive the same input voxels with a size of $B \times N_\rho \times N_\theta \times N_\phi \times 1$, with B for number of batches and 1 for the ternary occupancy value. When explaining feature dimensions, however, the number of batch B will be skipped for simplification from now on.

Fine Feature Extractor The Fine Feature Extractor encodes the structural relationship between directly neighboring voxels. The module consists of the multiple layers of 3D convolution networks, in which $3 \times 3 \times 3$ kernel slides in three dimensions and build a feature map of fine-scale structural relationships. The following fully connected layers then flatten the feature map into $N \times 64$ feature vectors, when $N = N_\rho \times N_\theta \times N_\phi$.

Coarse Feature Extractor The Coarse Feature Extractor encodes structurally more coarse features from the input voxels. Therefore, the module reduces the feature dimension to one-fourth of the original dimension through 3D pooling layers. Then, deconvolution layers recover the original dimension, so that it matches the output dimension of the Fine Feature Extractor. The output feature vectors from the two modules are then concatenated along columns, and become $N \times 128$ feature vectors.

Quad-View Integrator The Quad-View Integrator helps the model to extract structural features of space in a combination of four different directions at a time. With this module, every feature vector contains meaningful information even in highly occluded spaces. It concatenates each feature vector with three other vectors from correspondingly perpendicular azimuthal angles $(90°, 180°, 270°)$, as described in Fig. 4. In this work, the dimension of the parameters is $N \times 64$ after the two fully connected layers before the Quad-View Integrator. The Quad-View Integrator concatenates the feature vector of each voxel with that of three other voxels,

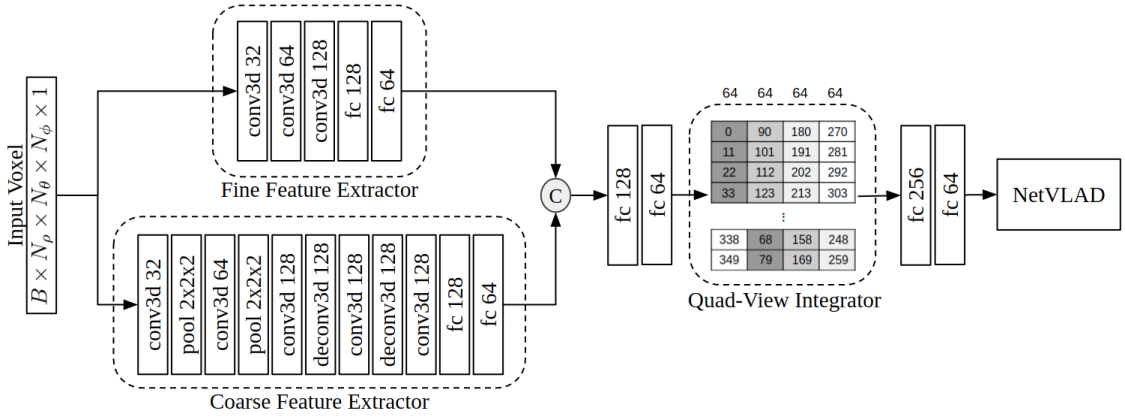

Fig. 4. Network Architecture of SpoxelNet. It has three main components as following: Fine Feature Extractor, Coarse Feature Extractor, and Quad-View Integrator. The outputs of the Fine Feature Extractor and the Coarse Feature Extractor are concatenated along columns. The numbers inside the boxes of Quad-View Integrator represent the corresponding azimuthal angles θ of the feature vectors, of which the dimension is shown above the module. The necessity of each part is empirically proven.

which are correspondingly perpendicular in azimuthal angle, *i.e.* $\{\theta_j, \theta_j + 90°, \theta_j + 180°, \theta_j + 270°\}$. The output of the Quad-View Integrator is $N \times 256$ feature vectors. Lastly, before the NetVLAD, two fully connected layers reduce the feature dimension from 256 to 64.

NetVLAD NetVLAD [8] was originally developed for an image retrieval purpose, but PointNetVLAD [6] successfully implemented it for 3D point cloud-based retrieval and trained it with end-to-end manner. Appreciating the accomplishment of PointNetVLAD, SpoxelNet also utilizes the NetVLAD for extracting global feature vectors.

IV. Experiments

In this section, SpoxelNet (SPX) is tested in various indoor datasets, and compared with other place recognition models. PointNetVLAD (PNVLAD) is compared with SpoxelNet as a deep learning-based counterpart, and Scan Context (SC) is compared as a non-learning-based counterpart. Since Scan Context is designed to output one best answer, its recall rate at the top 1 candidate is listed as a result and that of the top 1% candidates is not.

For experiments, local maps are created along trajectory by combining multiple neighboring point clouds. In this work, local maps are generated every one meter by combining every fourth one of twenty neighboring point clouds. Then, each local map is cropped in a radius of 10 meters from the origin and points of its ground plane are removed because those are mostly non-informative as suggested in [6]. The preprocessed local maps are then voxelized with $N_\rho = 16$, $N_\theta = 32$, $N_\phi = 8$, $\Delta\rho = 0.625$m, $\Delta\theta = 11.25°$, and $\Delta\phi = 2°$. These values were empirically chosen.

A. Benchmark Datasets

NAVER LABS indoor dataset[1] is used for testing the performance of SpoxelNet in highly occlusive indoor spaces. As listed in Table I, the dataset consists of data from five different places, which are scanned with a 16-channel LiDAR sensor (VLP-16) installed on a mapping robot. Multiple sequences of these places over several months are available, and all these places stayed public while the robot was scanning the areas. As a consequence, many people were scanned throughout the process and point clouds became very occluded and noisy.

Each place has different characteristics that are worth noticing before interpreting the experiment results. The Department Store B1 and Department Store 1F datasets are generated from the same building, but designs and structures of those are significantly different from each other, as shown in Fig. 5. The Subway Station dataset is generated in a highly crowded subway station. The point clouds of this place consist of points of even more moving objects compared to other datasets used in this work. The Office dataset is generated in the office of NAVER LABS. The dataset consists of only a few moving objects scanned, but the spaces are so narrow that the level of occlusivity is particularly high throughout the dataset. The Parking Lot dataset is created in an underground parking lot of a building. It is not as occlusive as other places but contains repetition of similar scenes of parked cars, which makes differentiation of places more difficult.

B. Data Augmentation and Training

For experiments, DB(a), DB(b), DB(c), and DB(d) in Table I are used equally for training SpoxelNet and PointNetVLAD. DB(e) was not included in the training set for a testing purpose. The training set was augmented in two different ways. First, we removed points of moving objects in each local map and generated a clean version of local maps. Second, both clean and raw local maps are rotated in different angles to increase the rotation-invariance of the trained model. As a result, the size of the training set increased from 5314 to 31884. About 70% of the augmented training dataset is used for training and the rest was used for validating the model during the training process. Local

[1]is available at https://www.naverlabs.com/storyDetail/172

Fig. 5. Pictures taken at each place of the dataset. The designs are significantly different from each other.

TABLE I
BENCHMARK DATASETS

Place	Date	Local Maps	Code
Department Store B1 (Dept. B1)	2019-02-22	1684	DB(a)
	2019-04-16	1151	DB(b)
	2019-04-16	896	DB(c)
	2019-08-20	1583	DB(d)
	2019-08-21	1720	DB(e)
Department Store 1F (Dept. 1F)	2019-02-21	1533	DF(a)
	2019-04-16	1755	DF(b)
	2019-08-20	1759	DF(c)
	2019-08-21	873	DF(d)
Subway Station	2019-11-25	2524	SS(a)
	2019-12-04	1276	SS(b)
	2019-12-11	838	SS(c)
Office	2018-12-07	511	OF(a)
	2019-07-08	796	OF(b)
Parking Lot	2019-07-03	414	PL(a)
	2019-07-03	431	PL(b)
	2019-07-10	289	PL(c)
	2019-07-10	304	PL(d)

maps that are located within two meters from each other are listed as positive queries ($\mathcal{P}_{pos}$) of each other, and the ones outside are listed as negative queries ($\mathcal{P}_{neg}$). The model implemented the lazy quadruplet loss function that was introduced in PointNetVLAD, with its parameters as following: $\alpha = 0.5, \beta = 0.2$, number of $\mathcal{P}_{pos} = 2$, and number of $\mathcal{P}_{neg} = 18$.

C. Place Recognition in Various Indoor Spaces

As listed in Table II, the recall rates of PointNetVLAD and Scan Context are noticeably lower in crowded indoor spaces, compared to their reported recall rates in outdoor roads. This is expectable given that the indoor datasets contain a great amount of occlusion and points of moving objects, to which the two models are innately vulnerable. SpoxelNet, in comparison, has the highest recall rates in every dataset. It is also worth noticing that the SpoxelNet was only trained with point clouds from Dept. B1, yet does not encounter a great amount of performance degradation in other places.

D. Ablation Studies

As shown in Table III, the ablation studies of SpoxelNet were done by removing each module from the complete SpoxelNet and individually testing its performance. The input sequence was the most recent one of each place and the reference map was created with the rest of the sequences.

The absence of the Fine Feature Extractor (FFE) results in the biggest drop in recall rates in every dataset, implying

TABLE II
PLACE RECOGNITION IN VARIOUS INDOOR SPACES

	Recall @Top 1%		Recall @Top 1		
	SPX	PNVLAD	SPX	PNVLAD	SC
Dept. B1	98.8	78.2	95.4	45.3	34.1
Dept. 1F	98.5	59.3	90.1	46.0	31.6
Subway Station	70.0	42.0	43.3	21.2	28.6
Office	80.6	54.2	56.6	23.1	50.0
Parking Lot	87.5	70.7	63.8	41.8	66.4

the essential role of the module. The recall rates do not drop as much with absences of Coarse Feature Extractor (CFE) and Quad-View Integrator (QVI), but the two modules surely contribute to the robustness and performance of the complete model, as shown with the recall rate of the complete SpoxelNet being the highest in every dataset.

TABLE III
ABLATION STUDIES - TOP 1% RECALL RATES

	SpoxelNet	No FFE	No CFE	No QVI
Dept. B1	98.8	80.5	94.3	90.4
Dept. 1F	98.5	77.9	90.2	88.2
Subway Station	70.0	49.3	62.5	60.0
Office	80.6	61.8	73.3	74.2
Parking Lot	87.5	63.8	79.8	81.9

E. Robustness to Moving Objects

The presence of moving objects greatly hampers point cloud-based place recognition. Therefore, robustness to moving objects is a very important element of place recognition models. To validate each model's robustness, all three models are tested on two versions of the Dept. 1F dataset: the original Dept. 1F and the clean version of Dept. 1F. The clean version is created by removing points of moving objects in each local map of Dept. 1F. When testing in the clean version, a reference map is generated with a clean version of DF(a), DF(b), and DF(c), and the input query was a clean version of DF(d). As shown in Table. IV, the recall rates of both PointNetVLAD and Scan Context increased significantly when those are tested in the clean version of the local maps. This proves that the presence of moving objects greatly influences the performance of the two models. On the other hand, the recall rates of SpoxelNet are high in both versions, and the difference was less than one percent, which demonstrates SpoxelNet's robustness to moving objects.

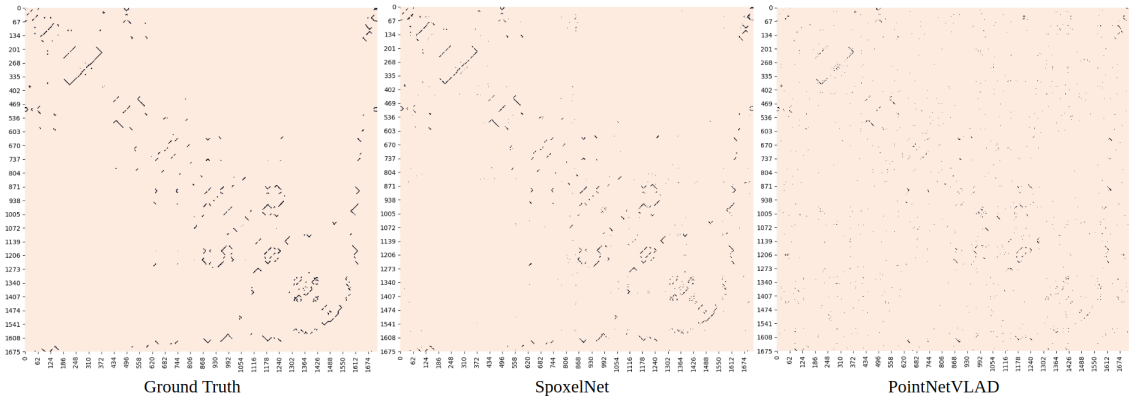

Fig. 6. Similarity Matrices at Department Store on 2019-08-21

TABLE IV
ROBUSTNESS TO MOVING OBJECTS

	Recall @Top 1%		Recall @Top 1		
	SPX	PNVLAD	SPX	PNVLAD	SC
Dept. 1F	98.5	59.3	90.1	46.0	31.6
Dept. 1F clean	98.6	75.7	91.0	52.0	70.9
Difference	0.1	16.4	0.9	6.0	39.3

F. Robustness to Temporal Variation

Indoor spaces, such as department stores, routinely undergo visual and structural changes. Therefore, robustness to temporal variation is another important element of place recognition models. To test this, each model was tested multiple times with a different set of input queries from Dept. 1F. As shown in Table V, the recall rates of SpoxelNet were consistently high with smaller variance, compared to other models.

TABLE V
ROBUSTNESS TO TEMPORAL VARIATION

Input Date	Recall @Top 1%		Recall @Top 1		
	SPX	PNVLAD	SPX	PNVLAD	SC
2019-02-21	96.0	65.4	82.9	39.8	33.6
2019-04-16	94.3	57.8	83.1	29.5	24.1
2019-08-20	96.0	60.3	84.5	37.7	22.2
2019-08-21	98.5	59.3	90.1	46.0	31.6

G. Finding Loop-closure Candidates

Place recognition can also be used for identifying loop candidates in a SLAM pipeline. For place recognition, top candidates in a prebuilt reference map are retrieved. In a SLAM situation, however, it is not guaranteed that a true loop exists. Therefore, instead of retrieving a designated number of candidates from the k-dimensional tree, the model can be modified to set a distance threshold and find loop candidates of which the global feature descriptor is placed within the threshold. In the test, total ten adjacent local maps before and after the input local map were excluded from the candidate pool to avoid retrieving a meaningless loop.

To compare the performance of SpoxelNet and PointNetVLAD as a loop candidate detector, similarity matrices are drawn in Fig. 6 by calculating the euclidean distances between the global descriptors of local maps and classifying those in binary with a distance threshold. The ground truth graph is drawn by identifying two local maps located within 2 meters from each other as a loop. The graphs of each model is drawn at its highest $\mathcal{F}_1$ score, which is calculated based on each model's number of true positive (TP), false negative (FN) and false positive (FP) as following:

$$\mathcal{F}_1 = \frac{2 \cdot TP}{2 \cdot TP + FN + FP} \tag{1}$$

As shown in Fig. 6, the similarity matrix by SpoxelNet is more similar to the ground truth graph, and contains much less noise, compared to the graph of PointNetVLAD.

V. CONCLUSION

This paper presents a solution to the point cloud based place recognition problem in crowded indoor spaces. The suggested spherical voxelization well represents the structures of the input point clouds, and the ternary occupancy value minimizes the false representation of occluded spaces. Furthermore, the proposed model architecture successfully recognizes the structural similarity between voxelized point-clouds, regardless of moving objects or temporal variation. The experiment results validate the model's state-of-the-art performance in various indoor spaces, which outruns the performances of other previously proposed methods.

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. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 231–237.

[2] S. Lowry, N. Sünderhauf, P. Newman, J. J. Leonard, D. Cox, P. Corke, and M. J. Milford, "Visual place recognition: A survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2015.

[3] L. Perdomo, D. Pittol, M. Mantelli, R. Maffei, M. Kolberg, and E. Prestes, "c-M2DP: A fast point cloud descriptor with color information to perform loop closure detection," in *Proc. of the IEEE Int. Conf. on Automation Science and Engineering (CASE)*, 2019, pp. 1145–1150.

[4] M. Bosse and R. Zlot, "Place recognition using keypoint voting in large 3D lidar datasets," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2013, pp. 2677–2684.

[5] G. Kim, B. Park, and A. Kim, "1-day learning, 1-year localization: Long-term LiDAR localization using scan context image," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1948–1955, 2019.

[6] M. A. Uy and G. H. Lee, "PointNetVLAD: Deep point cloud based retrieval for large-scale place recognition," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[7] Z. Liu, S. Zhou, C. Suo, P. Yin, W. Chen, H. Wang, H. Li, and Y.-H. Liu, "LPD-Net: 3D point cloud learning for large-scale place recognition and environment analysis," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2831–2840.

[8] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 5297–5307.

[9] M. Magnusson, H. Andreasson, A. Nuchter, and A. J. Lilienthal, "Appearance-based loop detection from 3D laser data using the normal distributions transform," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2009, pp. 23–28.

[10] T. Röhling, J. Mack, and D. Schulz, "A fast histogram-based similarity measure for detecting loop closures in 3-D lidar data," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015, pp. 736–741.

[11] J. Lin and F. Zhang, "A fast, complete, point cloud based loop closure for lidar odometry and mapping," *arXiv preprint arXiv:1909.11811*, 2019.

[12] N. Muhammad and S. Lacroix, "Loop closure detection using small-sized signatures from 3D lidar data," in *Proc. of the IEEE Int. Symp. on Safety, Security, and Rescue Robotics (SSRR)*, 2011, pp. 333–338.

[13] W. Wohlkinger and M. Vincze, "Ensemble of shape functions for 3D object classification," in *Proc. of the IEEE Int. Conf. on Robotics and Biomimetics (ROBIO)*, 2011, pp. 2987–2992.

[14] G. Kim and A. Kim, "Scan Context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2018, pp. 4802–4809.

[15] J. Guo, P. V. Borges, C. Park, and A. Gawel, "Local descriptor for robust place recognition using lidar intensity," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1470–1477, 2019.

[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. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 652–660.

[17] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2015, pp. 922–928.

[18] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 4490–4499.

[19] R. Dubé, A. Cramariuc, D. Dugas, J. Nieto, R. Siegwart, and C. Cadena, "SegMap: 3D segment mapping using data-driven descriptors," in *Proc. of the Int. Conf. on Robotics: Science and Systems (RSS)*, 2018.

[20] R. Dubé, D. Dugas, E. Stumm, J. Nieto, R. Siegwart, and C. Cadena, "SegMatch: Segment based place recognition in 3D point clouds," in *Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2017, pp. 5266–5272.

[21] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *Proc. of the IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3354–3361.

[22] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford Robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[23] R. Sahdev and J. K. Tsotsos, "Indoor place recognition system for localization of mobile robots," in *Proc. of the IEEE Int. Conf. on Computer and Robot Vision (CRV)*, 2016, pp. 53–60.

[24] Y. Zhuang, N. Jiang, H. Hu, and F. Yan, "3-D-laser-based scene measurement and place recognition for mobile robots in dynamic indoor environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 2, pp. 438–450, 2012.

[25] A. Milstein, "Occupancy grid maps for localization and mapping," *Motion planning*, pp. 381–408, 2008.

[26] J. Schauer and A. Nüchter, "The Peopleremover—removing dynamic objects from 3-D point cloud data by traversing a voxel occupancy grid," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1679–1686, 2018.

[27] S. Park, S. Wang, H. Lim, and U. Kang, "Curved-voxel clustering for accurate segmentation of 3D lidar point clouds with real-time performance," in *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2019, pp. 6459–6464.