

# Extracting Statistical Signatures of Geometry and Structure in 2D Occupancy Grid Maps for Global Localization

Su-Yong An  and Jaeyoung Kim 

**Abstract**—Global localization (or place recognition) is a method of finding the current location of a robot on a map generated by a mapping process, and it is an open field that has not yet been completely solved in the field of mobile robotics. Most existing approaches to global localization are based on extraction of interest point features and their descriptors whether from raw laser scans or 2D occupancy grid maps. In this letter, unlike most approaches, we propose a novel method of extracting a statistical signature of geometric and structural features from a submap. A boundary and free-space features can characterize a geometric shape, while a reflection symmetry can quantify a structural shape of the submap. Experiments using five pre-built map publicly available demonstrate that the proposed method outperforms the other state-of-the-art image-based methods by examining precision-recall curve especially when occupancy noise added to the submap is progressively increased.

**Index Terms**—Localization, SLAM, global localization, 2D LiDAR.

## I. INTRODUCTION

**R**ELIABLE positioning of mobile robots is essential for high-level tasks such as guidance, delivery, surveillance, and cleaning. This is referred as localization in mobile robotics literature. Localization includes local localization and global localization [1]. Local localization aims to compensate accumulated location errors based on previous position and sensor data during robot navigation. On the other hand, global localization can locate robots without prior knowledge of the location, allowing to solve the kidnapped robot problem. This is more significant and necessary than local localization in situations where the robots are likely to experience severe positioning errors.

Manuscript received October 10, 2021; accepted January 26, 2022. Date of publication February 14, 2022; date of current version February 24, 2022. This letter was recommended for publication by Associate Editor U. Frese and Editor S. Behnke upon evaluation of the reviewers' comments. This work was supported in part by the Korea Institute of Planning and Evaluation for Technology in Food, Agriculture and Forestry (IPET), in part by the Korea Smart Farm R&D Foundation (KosFarm) through Smart Farm Innovation Technology Development Program, funded by the Ministry of Agriculture, Food and Rural Affairs (MAFRA) and Ministry of Science and ICT (MSIT), and in part by Rural Development Administration (RDA) under Grant 421032-04. (Corresponding author: Su-Yong An.)

The authors are with Daegu-Gyeongbuk Research Center, Electronics and Telecommunications Research Institute (ETRI), Daegu 42994, South Korea (e-mail: syong.an@etri.re.kr; jaeyoung@etri.re.kr).

Digital Object Identifier 10.1109/LRA.2022.3151154

There are many methods of global localization depending on the type of sensors used in map building. As a general process, features are extracted from maps, stored in a database, and then compared with sensor data obtained from a current robot position. Loop closure detection [2], map merging for distributed multi-robot SLAM [3], and place recognition [4]–[6] in mobile robotics also deal with problems similar to global localization.

Existing approaches to global localization for mobile robots can be divided into two categories. The first is a method of extracting interest point features (i.e., keypoints) from 2D laser scan data and expressing them as descriptors using surrounding information around the features [4], [7]. By using acquisition characteristics of the laser scan data, keypoints are extracted based on range, normal, curvature, and orthogonality of the scan data. Descriptors are used for direct matching between keypoints or to generate scan signatures in conjunction with a bag-of-words (BoW) framework [5]. However, when applying in cluttered environments or using a laser scanner with low angular resolution, this method may fail to extract distinctive keypoints from a single scan due to sensor noise or its sparseness.

The second method is to convert accumulated laser scans into a 2D occupancy grid map and extract keypoints from this image. The keypoints and their descriptors can be extracted using SIFT [8] and SURF [9] which are representative methods in the field of visual object recognition. This approach has the advantage of reusing many algorithms adopted in computer vision, but is prone to inconsistencies due to noise generated in the process of projecting the scan data onto the image.

In this paper, we propose a new method to extract statistics of occupied and unoccupied areas in submap images instead of extracting keypoints. The proposed method divides the submap into boundary space and free-space, and represents distribution characteristics of each space as histograms to generate submap signatures. These two histograms reveal the geometric information of the submap. In addition, a symmetry score is introduced to quantify the structural information of the submap, and is used as a weight when calculating similarity between submaps.

In summary, the contributions of this paper are (1) to quantify geometric distribution of boundary and free-space as statistical signatures rather than extracting keypoints, (2) to introduce reflection symmetry to describe structural information (overall shape of submaps), and (3) to integrate geometric and structural

information of submaps explicitly to form as one metric used for submap matching.

## II. RELATED WORK

Place recognition (i.e., Global localization) is a field that is being continuously studied in the field of mobile robotics. There are a method using an image itself through deep learning and a method using 2D/3D maps generated by several types of sensors [10]. Since this paper deals with place recognition through the feature analysis of 2D grid maps, this section mainly focuses on the method using 2D LiDAR sensors.

Bosse and Zlot [4] outlined first works on extracting keypoints from raw laser scans and defining the information around the keypoints as descriptors. They applied combination of two keypoint detectors and six descriptors to a large-scale SLAM problem, reporting that combination of the curvature cluster-based keypoint detector and the moment grid descriptor showed the best performance. Since the descriptor was generated using information within a relatively large radius (i.e., 9 m) centered on the keypoint, it can be seen that descriptors play a role in encoding the entire submap, not as a local descriptor.

Fast Laser Interest Region Transform (FLIRT), a multi-scale interest region operator for 2D range data, is proposed in [7]. FLIRT is a method of extracting keypoints based on a curve approximation of a laser scan and defining occupancy information in the polar region around the keypoint as a descriptor. This approach was used together with RANSAC to prove its effectiveness by applying it to solving loop closure and global localization in indoor and outdoor environments.

FLIRT has been further extended to geometrical FLIRT phrases (GFP) for practical applications to large-scale datasets [5]. The idea of this approach is based on BoW scheme with introducing a weak geometrical verification. Each keypoint from a raw laser scan is associated with a predefined word, and a histogram of word frequency is used as a signature of a place. Subsequently, for higher retrieval performance in outdoor environments with a lot of laser scan data, geometrical landmark relations (GLARE) was proposed [6], which describes spatial relations of landmarks as a scan signature unlike BoW method. A 2D histogram of relative orientations and distances of co-occurring landmarks is defined as a scan signature through a soft-voting scheme which can deal with sensor uncertainties and noisy range data. It showed a faster search performance than GFP by applying an approximate nearest neighbor search in the query process.

Another method is to convert a raw laser scan into a 2D image and extract keypoints from the image using existing computer vision techniques. Li and Olson [11] proposed a method to extract stable keypoints by rasterizing LiDAR data by projecting it into 2D Euclidean space, followed by applying a multi-scale Kanade-Tomasi detector, a variant of a Harris corner detector. This method is easy to apply to SLAM applications because it uses the covariance matrix that naturally accompanies the corner detector algorithm to calculate spatial uncertainty of keypoints. Experiments with 2D/3D datasets have demonstrated that the

repeatability performance of the keypoint detector was superior to other methods.

Millane *et al.* [12] proposed a method of generating a signed distance function (SDF) map by applying distance transform after thresholding an occupancy probability grid of a submap. In this SDF map, points with high curvature were defined as keypoints, and their descriptors were constructed by adding an average SDF value to an orientation histogram of SIFT descriptors. Since the keypoint and descriptor extraction was performed on the SDF, it has the advantage of simultaneously capturing features of geometry of free and occupied space. This work has recently been extended to solve 3D localization [13].

A map merging method for multi-robot systems using spectrum analysis of a 2D grid map was proposed in [3]. For submaps created by different robots, orientations of the submaps were first compared using Hough spectra, and translations were obtained with additional X and Y spectrum through a projection to the x and y axes of the submap image. This approach has been proposed as a map merging for multi-robot systems, it can be also applied to solve global localization or loop closure.

Unlike the methods based on the extraction of keypoints and descriptors, the proposed method defines the geometric and structural distribution of whole submaps as statistical signatures. Since it does not rely on direct matching of keypoints extracted from submaps, it is insensitive to small noise or unexpected appearance of objects when constructing 2D grid maps. Moreover, the proposed method is compatible with any sensor capable of generating 2D grid maps (e.g., ultrasonic sensor, 2D/3D LiDAR, and 3D depth camera), and thus it is independent of sensor modality.

## III. SUBMAP FEATURE EXTRACTION

Given a submap image, we extract three features: a boundary, a free-space, and a symmetry feature. The boundary and free-space features can describe a geometric shape of the submap, whereas the symmetry feature can quantify overall shape of the submap. Our method explicitly considers the geometric and structural information of the submap concurrently, thereby having the advantage of being robust against noise.

*Boundary feature:* A submap can also be viewed as an object with a specific shape. Therefore, we define a shape function used in shape-based image classification to describe a distribution of a boundary area. We express the distribution of distance between two boundary points as a histogram by modifying the D2 function [14] used for 3D shape matching to a 2D version. Advantages of the D2 function include computational simplicity, invariant to rotation and translation, and insensitive to small perturbations.

*Free-space feature:* Since a free-space in maps indicates an area where robots can actually be located, the distribution of the free-space along with that of the occupied area provides very useful information for robot localization [12]. We represent the distribution of the free-space as a histogram by defining free-space nodes and calculating their shortest path distance from each node to the other nodes. Even if there are static or dynamic obstacles that were not in the free-space at the time of map

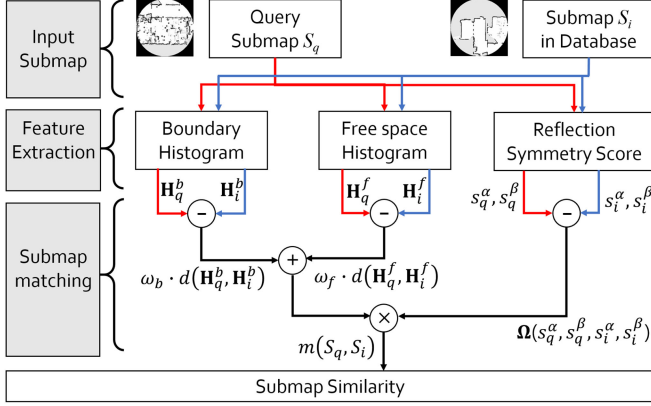


Fig. 1. Overall block diagram to calculate a submap similarity. Geometric features (boundary and free-space feature) and structural feature (reflection symmetry scores) are extracted from  $S_q$  and compared to all pre-built features from the submap  $S_i$  in a database to find the most similar submap.

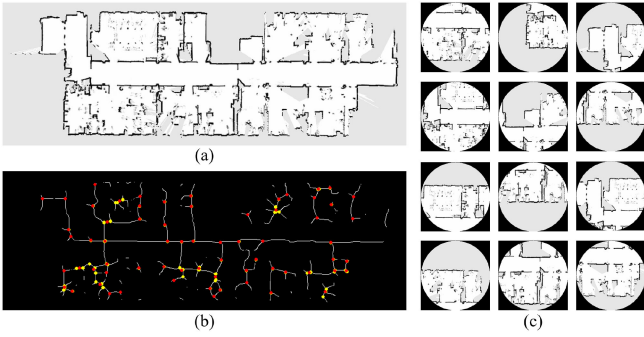


Fig. 2. Example of a submap partitioning procedure. (a) Original global map, *fr079* [21], (b) Extraction of junction nodes (yellow) and their clustering (red), (c) Submaps extracted from the global map.

building, the shortest path distance does not change much, thus it is insensitive to small perturbations.

*Symmetry feature:* A typical indoor environment can be classified into three types: corridors, rooms, and lobbies [15]. These spaces exhibit considerable symmetry due to structural characteristics of indoor buildings, and in particular, a reflection symmetry [16] can be a good feature to describe the overall structure of a place. Therefore, we compute a reflection symmetry score to quantify the structural feature of submaps and use it as a weighting factor for calculating submap similarity.

This section describes the extraction of the above three features from submaps and calculation of a similarity score between two submaps (Fig. 1).

### A. Submap Partitioning

Assume that we have created a complete 2D occupancy grid map  $\mathcal{M}$  using an available sensor such as a 2D laser scanner (Fig. 2(a)). To locate robots, we need to compare the most recently created submap with some part of  $\mathcal{M}$ , so the process of dividing the  $\mathcal{M}$  into submaps is necessary. We split it into submaps based on junction nodes extracted from  $\mathcal{M}$ .

First, a binary map  $\mathcal{M}^B$  is created and then distance transform [17] is applied for free-space detection on  $\mathcal{M}$ . Since

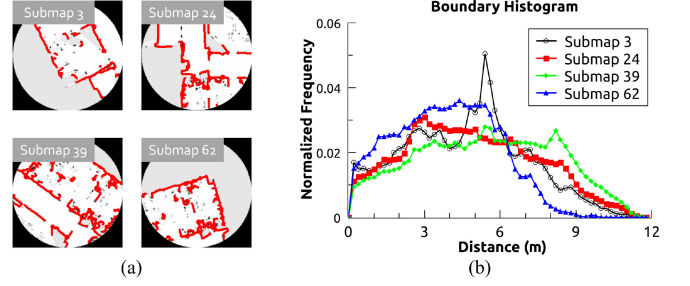


Fig. 3. Construction of boundary histogram. (a) Sampled boundary points in example submaps, (b) Representation of boundary histogram.

the raw distance transformed image contains many free-space regions, a morphological operator (erosion), followed by an image thinning technique [18] are applied to make 1-pixel wide edge map  $\mathcal{M}^E$  depicting only the major free-space. Finally, possible junction nodes are selected by performing junction test [19] on all nonzero pixels in  $\mathcal{M}^E$ , and clustered using DBSCAN [20] (Fig. 2(b)). The global map is thus expressed as the sum of submap images with a fixed radius  $r$  around each junction node (Fig. 2(c)),  $\mathcal{M} \simeq \bigcup_{v_i \in \mathcal{J}^M} \mathcal{S}_i$ , where  $\mathcal{S}_i$  is a  $i$ -th partitioned submap, and  $\mathcal{J}^M$  is the refined set of junction nodes. The submap radius  $r$  is dependent on range sensing capability of a sensor used.

### B. Boundary Feature

A boundary of a submap (i.e., distribution of occupied pixels) represents the overall outer shape of a place, which is the main characteristic that distinguishes the place. We perform the following procedure to quantify the boundary information.

1) *Submap Binarization:* Each pixel value of a submap represents a degree of occupancy. The binary image  $\mathcal{S}_i^B$  is obtained by performing inverse-binary thresholding ( $\mathcal{S}_i^B(u, v) = 1$ , if  $\mathcal{S}_i(u, v) < T^B$ , otherwise 0) to remove ambiguous pixels inside the submap and thus we only consider the clearly occupied pixels.

2) *Thinning:* During map building process, structures such as walls may be expressed as densely occupied areas due to sensor noise or mapping errors. Therefore, a thinning algorithm [22] is performed on  $\mathcal{S}_i^B$  to generate a boundary edge map  $\mathcal{S}_i^E$  with 1-pixel wide.

3) *Uniform sampling of boundary points:* Here, the boundary indicates a set of nonzero pixels in  $\mathcal{S}_i^E$ . We can reduce the amount of computation by reducing the number of boundary points  $\mathcal{P}_i^B = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_{N_B}\}$  after removing adjacent ones through uniform sampling (Fig. 3(a)).

4) *Construction of Histogram* [14]: For all pairs of boundary points in  $\mathcal{P}_i^B$ , we calculate the pairwise Euclidean distance and construct a  $N_B \times N_B$  symmetric matrix as

$$\mathbf{D}_i^B = \begin{bmatrix} d_{11} & \cdots & d_{1N_B} \\ \vdots & \ddots & \vdots \\ d_{N_B 1} & \cdots & d_{N_B N_B} \end{bmatrix}, d_{mn} = \|\mathbf{p}_m - \mathbf{p}_n\|, \quad (1)$$

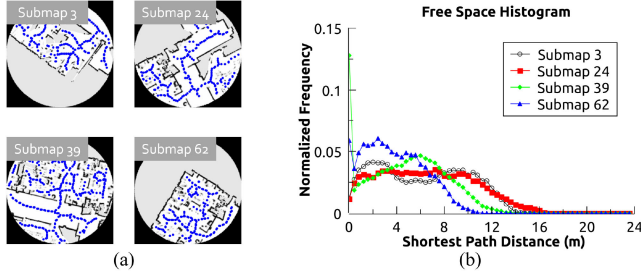


Fig. 4. Construction of free-space histogram. (a) Distribution of free-space nodes in example submaps, (b) Representation of free-space histogram.

where each element of  $\{d_{mn}|m < n\}$  is accumulated into the corresponding bin  $\lfloor \frac{d_{mn}}{\Delta_b} \rfloor$  of a 1-D histogram (Fig. 3(b)):

$$\mathbf{H}_i^b = \{h_i^b(k)|k = 1, \dots, N_{hb}\}, \quad (2)$$

where  $\mathbf{H}_i^b$  is a normalized vector denoting histogram of  $\mathbf{D}_i^B$ , and  $h_i^b(k)$  indicates the number of elements in the  $k$ -th bin. The number of bin  $N_{hb} = \lceil \frac{2r}{\Delta_b} \rceil$  is dependent on map size  $2r$  and fixed bin width  $\Delta_b$ .

Fig. 3 illustrates different submaps and their corresponding boundary histograms. The histogram of submap 3 and that of the other submaps are significantly different in the peak pattern. However, submap 24 and 39 exhibit similar pattern changes despite having different shapes. Therefore, there is a limit to distinguishing all submaps with only boundary distribution, and it has implications that not only boundary information but also utilization of free-space should be considered to improve submap matching performance.

### C. Free-Space Feature

A free-space provides useful information about internal structures of a place [12]. We quantify a distribution of the free-space by performing the following steps.

1) *Extraction of Free-Space Nodes*: The process of finding free-space nodes is analogous to the submap partitioning process (Section III-A) in that it goes through the process of image binarization, distance transform, and thinning. However, in this case, uniformly sampled free-space nodes  $\mathcal{V}_i^F = \{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_{N_f}\}$  instead of junction nodes are obtained from the 1-pixel wide edge map of the submap (Fig. 4(a)).

2) *Calculation of Shortest Path Distance*: We can easily find a shortest path from any  $\mathbf{v}_m$  to  $\mathbf{v}_n$  by constructing an undirected graph  $G(\mathcal{V}, \mathcal{E})$  whose vertices and edges are free-space nodes and their connections. The undirected graph  $G$  can be represented by an adjacency matrix  $\mathbf{A} = [a_{mn}]$ , where each element  $a_{mn}$  is the Euclidean distance between vertices (i.e., free-space nodes). If an occupied pixel exists on the straight path  $\mathbf{v}_m \leftrightarrow \mathbf{v}_n$ , then  $a_{mn}$  has an infinite value. We use Dijkstra's algorithm [23] to find a shortest path distance.

3) *Construction of Histogram*: In a similar manner to the construction of the boundary histogram (1), for all pairs of free-space nodes in  $\mathcal{V}_i^F$ , we calculate the shortest path distance and

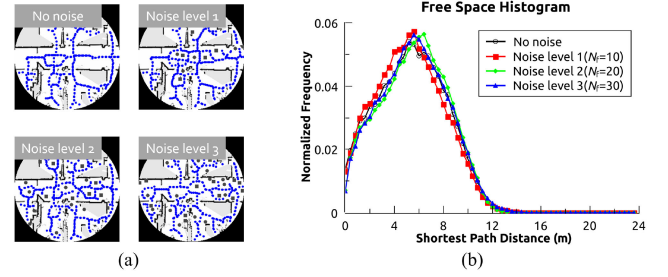


Fig. 5. Construction of free-space histogram by varying noise levels. (a) Distribution of free-space nodes in submap with noise added, (b) Representation of free-space histogram.

construct a  $N_F \times N_F$  symmetric matrix as

$$\mathbf{L}_i^F = \begin{bmatrix} l_{11} & \dots & l_{1N_F} \\ \vdots & \ddots & \vdots \\ l_{N_F 1} & \dots & l_{N_F N_F} \end{bmatrix}, l_{mn} = SP(\mathbf{v}_m, \mathbf{v}_n), \quad (3)$$

where  $SP$  is a function that returns the shortest path distance between  $\mathbf{v}_m$  and  $\mathbf{v}_n$ . Each element of  $\{l_{mn}|m < n\}$  is accumulated into the corresponding bin  $\lfloor \frac{l_{mn}}{\Delta_f} \rfloor$  of a 1-D histogram (Fig. 4(b)):

$$\mathbf{H}_i^f = \{h_i^f(k)|k = 1, \dots, N_{hf}\}, \quad (4)$$

where  $\mathbf{H}_i^f$  is a normalized vector denoting histogram of  $\mathbf{L}_i^F$ . The bin width  $\Delta_f$  is a constant as in the boundary histogram. We found empirically that the maximum value of  $l_{mn}$  is bounded within four times of submap radius  $r$ , so the number of bin  $N_{hf}$  becomes  $\lceil \frac{4r}{\Delta_f} \rceil$ .

Submap 24 and 39 were not distinguished well by the boundary histogram (Fig. 3), however, when the free-space histogram was applied, there is a more pronounced difference as shown in Fig. 4(b). On the other hand, interestingly, histogram pattern changes of submap 3 and 24 are similar unlike the case of the boundary histogram. This implies that the free-space histogram has complementary relationship with the boundary histogram, and the performance of distinguishing submaps can be improved through the combination of the two histograms.

The robustness of the free-space histogram against noise was tested by randomly placing  $N_f$  number of virtual obstacles (e.g., moving persons, temporarily located indoor structures such as tables or chairs) which did not exist during map building process (Fig. 5(a)). Although the distribution of free-space nodes alters due to the addition of virtual obstacles, this does not significantly affect the shortest path distance, thus leading to negligible changes in the histogram (Fig. 5(b)).

### D. Reflection Symmetry Score

Both the boundary and free-space histogram described above quantify a geometric shape of a submap, whereas a symmetry quantifies a global structural shape. Symmetry types include rotation, reflection, translation, and glide reflection symmetry [16]. Among these symmetries, a reflection is the most common

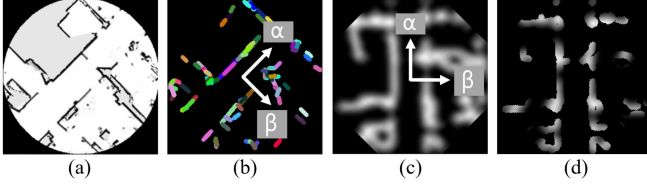


Fig. 6. Calculation of reflection symmetry scores based on two reflection symmetry axes. The two axes  $(\alpha, \beta)$  are not necessarily orthogonal to each other. (a) Original submap, (b) Extracted line segments and their reflection symmetry axes, (c) Rotated and Gaussian-blurred image of boundary edge map, (d) Intensity response of reflection symmetry based on the  $\alpha$ -axis.

type of symmetry found in indoor environments. We obtain two reflection symmetry scores for each submap through the procedure below.

1) *Extraction of Line Segments*: As a first step, we apply a probabilistic Hough Transform [24] to the boundary edge map  $\mathcal{S}_i^E$  to find line segments  $L_i^S = \{\lambda_1, \lambda_2, \dots, \lambda_{N_L} \mid \Gamma(\lambda_k) > \lambda_\theta\}$ , where  $\Gamma$  returns a length of  $\lambda_k$  (Fig. 6(a, b)).

2) *Finding Axes of Symmetry*: Since the indoor structure of the building is mostly rectangular, two axes of reflection symmetry  $(\alpha, \beta)$  are found by voting on the angles  $\Theta(\lambda_k) \in [-\frac{\pi}{2}, \frac{\pi}{2}]$  of the extracted line segments.

3) *Calculation of Reflection Symmetry Score*: We make a rotated boundary edge map  $\mathcal{S}_i^{E\alpha}$  by rotating the  $\mathcal{S}_i^E$  so that the symmetry axis  $\alpha$  is vertically aligned. Then a Gaussian smoothing which can attenuate sensor uncertainties and noisy range data is applied to obtain a blurred image  $\mathcal{S}_i^{G\alpha}$  (Fig. 6(c)). We calculate an average intensity of  $\mathcal{S}_i^{G\alpha}$  corresponding to the position of all nonzero pixels in  $\mathcal{S}_i^{E\alpha}$ , where the  $\mathcal{S}_i^{G\alpha}$  is vertically flipped image of the  $\mathcal{S}_i^{E\alpha}$  (Fig. 6(d)):

$$s_i^\alpha = \frac{\sum_{(u,v) \in \mathcal{N}_{nz}^\alpha} \mathcal{S}_i^{G\alpha}(u,v)}{|\mathcal{N}_{nz}^\alpha|}, \quad (5)$$

where  $\mathcal{N}_{nz}^\alpha$  and  $|\mathcal{N}_{nz}^\alpha|$  are the set of nonzero pixels in  $\mathcal{S}_i^{E\alpha}$  and its number, respectively. A score  $s_i^\beta$  by the reflection symmetry axis  $\beta$  can be calculated in a similar way.

### E. Submap Similarity

Once a query submap  $\mathcal{S}_q$  is generated during robot navigation, it is compared against all submaps to find a best matched submap in the database. The submap matching score  $s$  is calculated by aggregating the three features, leading to explicitly including geometric and structural information as follows

$$s = m(\mathcal{S}_q, \mathcal{S}_i) = \underbrace{\Omega(s_q^\alpha, s_q^\beta, s_i^\alpha, s_i^\beta)}_{\text{structural feature}} \times \underbrace{\left[ \omega_b \cdot d(\mathbf{H}_q^b, \mathbf{H}_i^b) + \omega_f \cdot d(\mathbf{H}_q^f, \mathbf{H}_i^f) \right]}_{\text{geometric feature}}, \quad (6)$$

where  $\omega_b$  and  $\omega_f$  are the weights of the boundary and free-space histogram, respectively, whose sum is 1.0. The function  $d(\mathbf{A}, \mathbf{B})$  is Bhattacharyya distance [25] of two histograms. The reflection symmetry axes  $(\alpha, \beta)$  are independent of each other, thus  $\Omega$  can

TABLE I  
DESCRIPTION OF DATASETS USED

Dataset	Size (pixel)	# of Submaps	Type
<i>fr079</i>	909 × 322	67	Corridor, Room
<i>intel</i>	579 × 581	68	Corridor, Room, Lobby
<i>csail</i>	969 × 1341	102	Corridor, Lobby
<i>aces</i>	1172 × 1113	98	Corridor, Lobby
<i>killian</i>	2491 × 1990	91	Corridor

be denoted as

$$\Omega = \phi(s_q^\alpha, s_i^\alpha) \cdot \phi(s_q^\beta, s_i^\beta), \quad (7)$$

$$\phi(s_i, s_j) = \eta e \left( -\frac{(s_i - s_j)^2}{2\sigma^2} \right)^{-1}, \quad (8)$$

where  $\eta$  is a scaling factor and  $\sigma$  is an adjusting factor that controls the effect of structural feature on the submap matching score  $s$ . The submap similarity  $\rho$  is the reciprocal of  $s$ , so it increases as the two submaps are similar.

## IV. EXPERIMENTAL EVALUATION

### A. Datasets and Baseline Methods

To validate the proposed method, we used five publicly available datasets for our experiments (Table I) [21]. All maps are provided as images which consist of corridors, rooms, and lobbies, varying in size from approximately  $45 \times 16$  m to  $125 \times 100$  m with a resolution of 0.05 m per pixel (Fig. 7). In general, the number of submaps should increase as the size of the map increases, however, fewer submaps are extracted for *killian* composed of long corridors because the submap database was built based on junction node extraction.

The proposed method was compared with the baseline methods below, which include Shape context [26], SIFT/SURF-based BoW and Term Frequency-Inverse Document Frequency (TF-IDF) [8], [9], [27], and SDF [12].

*Shape Context*: The shape context is an algorithm that describes global shapes in binarized images, and is mainly used for image matching, classification, and retrieval. The number of angular bins and radial bins are the main parameters of the shape context, and in our experiments, the matching performance was measured with 24 and 12, respectively.

*SIFT/SURF+BoW*: SIFT and SURF are methods for extracting local features and descriptors in images and are widely used in computer vision and robotics fields. A BoW in conjunction with the local feature extractor is mainly used for image classification, search, and scene recognition. In this paper, parameters of the local feature extractor were set so that no more than 300 features were extracted for each submap, and a codebook size of the BoW was set to 1024.

*SIFT/SURF+TF-IDF*: TF-IDF uses word frequency and inverse document frequency to weight the importance of each feature, and is generally known to outperform the BoW. The parameter setting was the same as that of the above BoW method.

*SDF*: Submap matching is performed using RANSAC based on keypoints and descriptors extracted from the local SDF geometry. Up to 200 keypoints were extracted per submap.

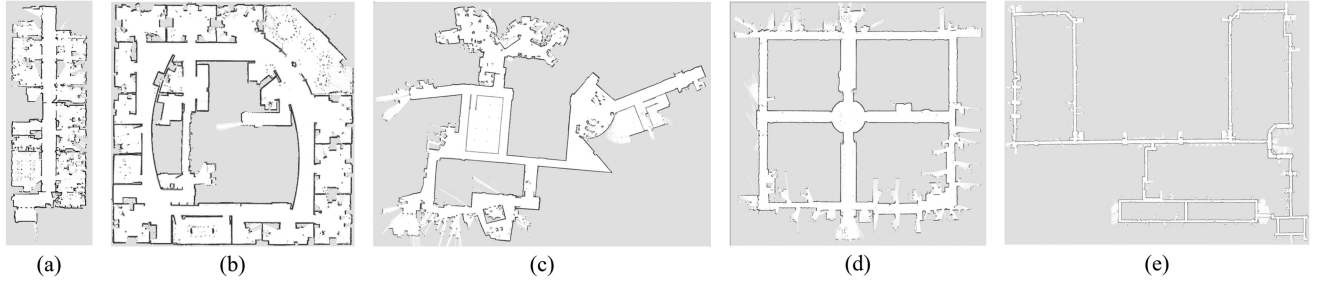


Fig. 7. The maps used in experiments [21]. (a) *fr079*, (b) *intel*, (c) *csail*, (d) *aces*, (e) *killian*.

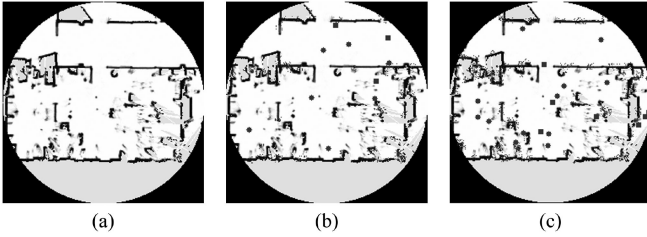


Fig. 8. Noisy submap generation by adding boundary and free-space noise. (a) Original submap without noise, (b) Submap with noise level 1 ( $N_b, N_f$ ) = (50, 10), (c) Submap with noise level 2 ( $N_b, N_f$ ) = (100, 20).

### B. Performance Evaluation

First, a query submap  $S_q$  was obtained by cropping an image with radius  $r$  around randomly selected location  $(u, v)$  on a global map, followed by randomly rotated. In addition, to verify robustness against noise, here we added two different types of noise which are a boundary noise and a free-space noise. In this paper, two-stage noise levels were defined according to the number of noise elements  $(N_b, N_f)$ , where  $N_b$  and  $N_f$  denote the number of the boundary noise and that of the free-space noise, respectively. Fig. 8 illustrates the query submap  $\tilde{S}_q$  where noise is injected. The boundary noise is distributed around the randomly selected  $N_b$  number of occupied pixels, simulating sensor noise or mapping errors. The free-space noise is placed on randomly selected  $N_f$  number of unoccupied pixels in the form of a circle with radius  $r_n$  or a square with width  $2r_n$ , denoting moving obstacles or temporarily installed internal structures.

The query submap  $\tilde{S}_q$  was compared to all submaps in the database to find the best match based on the submap similarity  $\rho$ . If the difference between the location of the query submap and that of the submap with the highest similarity is within 1.0 m, then this match is determined to be a true match; otherwise, it is a false match. We performed 2,000 queries per dataset with parameters listed in Table II to construct precision-recall curves by varying the submap similarity threshold. Based on this, we quantitatively evaluated the performance of the proposed method by examining the area under precision-recall curve (AUPRC), recall at a precision of 0.9 (RP9), and F1 score.

TABLE II  
PARAMETERS USED IN EXPERIMENTS

Notation	Description	Value
$r$	Submap radius (m)	8
$\Delta_b$	Bin width of boundary histogram (m)	0.2
$\Delta_f$	Bin width of free-space histogram (m)	0.4
$\lambda_\theta$	Minimum length of line segment (m)	0.3
$\omega_b$	Weight of boundary histogram	0.9
$\omega_f$	Weight of free-space histogram	0.1
$\eta$	Scaling factor of reflection symmetry score	0.08
$\sigma$	Adjusting factor of reflection symmetry score	120

The proposed method, SIFT+TF-IDF, and SDF showed reasonable performance in all datasets from the experiments in which only simple random rotation was added without noise injecting into the query submap (Fig. 9, Table III). However, the performance superiority of the proposed method to the other methods was remarkable as the noise level increased. To sum up, when based on the three measurements of AUPRC, RP9, and F1 score, the proposed method compared to SIFT+TF-IDF which had best performance among baseline methods in the experiment of noise level 2 demonstrated the average performance improvement of 10.3%, 104.7%, and 6.3%, respectively.

Overall, the performance based on the *fr079* and *intel* datasets was better than that based on the other datasets, which is due to relatively high proportion of distinct regions on the map (Fig. 7). In particular, in case of the *fr079* and *intel*, the map is composed of rooms of various sizes, whereas most of areas of the *killian* dataset consists of long corridors, leading to inherently include many duplicated submaps in the submap database. Therefore, the performance based on the *killian* dataset was lowest among all datasets.

A maximum sensing range of a sensor used in localization process affects the determination of the submap radius  $r$ . A visible range of general indoor environments except for special environments such as long corridors or very large halls can be limited to within 10 m, so the experiments were performed with three submap sizes up to a maximum radius of 10 m (Fig. 10, Table IV). As a result, we found that there were gradual performance improvements in all datasets without exceptions because distinct regions were more likely to be included as the submap radius increased.

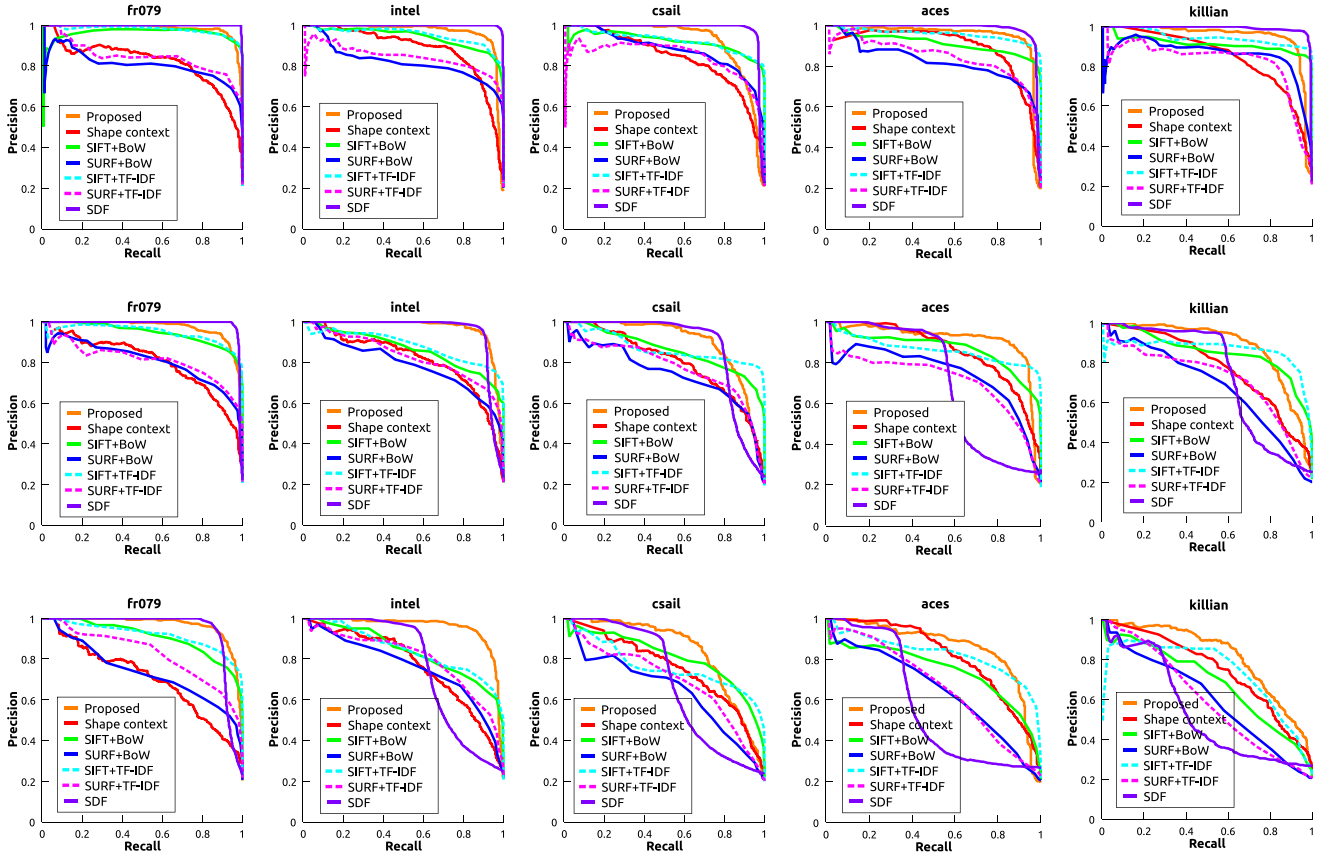


Fig. 9. Precision-recall performance comparison with six baseline methods on two noise levels. Each column represents datasets used in experiments. The first row: without noise, second row (noise level 1):  $(N_b, N_f) = (50, 10)$  noise added, third row (noise level 2):  $(N_b, N_f) = (100, 20)$  noise added.

TABLE III  
PERFORMANCE COMPARISON BY VARYING NOISE LEVEL

Noise Level	Method	<i>fr079</i>			<i>intel</i>			<i>csail</i>			<i>aces</i>			<i>killian</i>		
		AUC <sup>1</sup>	RP9 <sup>2</sup>	F1 <sup>3</sup>	AUC	RP9	F1	AUC	RP9	F1	AUC	RC	F1	AUC	RP9	F1
Without Noise	Shape Context	818 <sup>4</sup>	293	775	881	601	817	836	446	771	894	731	835	838	561	775
	SIFT+BoW	962	984	939	935	768	893	916	749	884	906	638	886	909	714	909
	SURF+BoW	798	149	801	810	136	798	870	459	819	814	147	793	854	271	852
	SIFT+TF-IDF	975	986	945	948	909	891	920	767	889	952	941	932	943	900	934
	SURF+TF-IDF	838	179	830	836	189	830	840	561	804	836	372	803	815	212	818
	SDF	<b>999</b>	<b>997</b>	<b>995</b>	<b>998</b>	<b>998</b>	<b>984</b>	<b>979</b>	<b>969</b>	<b>959</b>	<b>980</b>	<b>973</b>	<b>946</b>	<b>989</b>	<b>988</b>	<b>969</b>
	Proposed	984	966	943	965	946	931	921	796	848	949	940	928	949	924	916
Noise Level 1 <sup>5</sup>	Shape Context	800	338	743	811	387	757	800	335	727	859	575	773	770	429	696
	SIFT+BoW	939	785	892	856	426	808	850	440	807	854	487	805	841	353	807
	SURF+BoW	796	221	758	774	192	738	732	148	718	732	138	724	656	196	638
	SIFT+TF-IDF	942	871	898	876	570	845	862	290	<b>861</b>	853	276	855	859	551	<b>854</b>
	SURF+TF-IDF	805	139	778	815	340	773	762	146	746	696	22	708	694	69	698
	SDF	<b>988</b>	<b>980</b>	<b>969</b>	949	912	<b>926</b>	872	<b>787</b>	842	702	543	679	739	582	709
	Proposed	972	926	916	<b>965</b>	<b>921</b>	923	<b>896</b>	742	819	<b>912</b>	<b>852</b>	<b>879</b>	<b>883</b>	<b>699</b>	816
Noise Level 2 <sup>6</sup>	Shape Context	697	175	652	763	424	690	749	290	691	814	477	728	736	331	681
	SIFT+BoW	897	643	833	822	383	760	793	373	753	721	58	711	662	187	632
	SURF+BoW	717	141	692	741	210	723	630	68	635	618	52	610	567	99	562
	SIFT+TF-IDF	908	723	861	830	376	783	750	190	745	805	278	778	723	122	702
	SURF+TF-IDF	806	420	745	781	284	733	676	122	668	630	219	614	560	213	535
	SDF	939	891	907	762	591	712	683	486	632	565	335	494	531	218	456
	Proposed	<b>967</b>	<b>908</b>	<b>910</b>	<b>955</b>	<b>893</b>	<b>901</b>	<b>838</b>	<b>624</b>	<b>764</b>	<b>855</b>	<b>564</b>	<b>809</b>	<b>818</b>	<b>468</b>	<b>731</b>

<sup>1</sup>AUC: Area under precision-recall curve

<sup>2</sup>RP9: Recall at a precision of 0.9

<sup>3</sup>F1: The harmonic mean of precision and recall (F1 score)

<sup>4</sup>Numbers: All numbers should be divided by 1,000 to get the original meaning

<sup>5</sup>Noise Level 1: Boundary and free-space noise with  $(N_b, N_f) = (50, 10)$

<sup>6</sup>Noise Level 2: Boundary and free-space noise with  $(N_b, N_f) = (100, 20)$

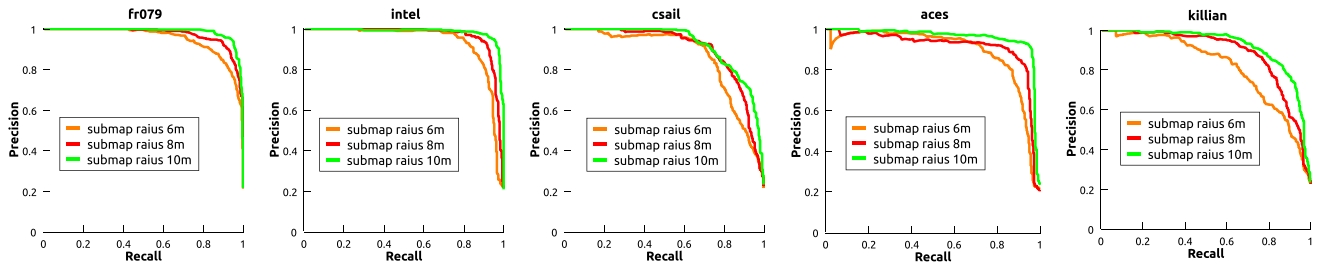


Fig. 10. Precision-recall performance comparison with three submap radii. As the submap radius increases, the performance of the proposed method also increases.

TABLE IV  
PERFORMANCE COMPARISON BY VARYING SUBMAP RADIUS

Submap Radius (m)	<i>fr079</i>			<i>intel</i>			<i>csail</i>			<i>aces</i>			<i>killian</i>		
	AUC	RP9	F1	AUC	RP9	F1	AUC	RP9	F1	AUC	RP9	F1	AUC	RP9	F1
6	951	831	876	935	850	876	880	751	822	889	733	836	816	476	729
8	972	926	916	965	921	923	896	742	819	912	852	879	883	699	816
10	<b>986</b>	<b>960</b>	<b>945</b>	<b>985</b>	<b>976</b>	<b>951</b>	<b>918</b>	<b>762</b>	<b>843</b>	<b>957</b>	<b>964</b>	<b>934</b>	<b>931</b>	<b>825</b>	<b>863</b>

## V. CONCLUSION

The key contribution of this paper is the idea of extracting statistical signatures rather than keypoints to describe geometric and structural shape of submaps. In particular, by dividing a submap into a boundary and a free-space using existing computer vision algorithms, statistical values of each area were explicitly extracted. Moreover, reflection symmetry, which is often seen in building structures, was quantified in order to capture structural information of a submap. We explicitly integrated these three features to define a similarity score between two submaps. Our comparative experiments with the state-of-the-art image-based methods demonstrated that the proposed method was fairly effective at distinguishing between submaps especially when severe noise is injected to submaps. Overall, we achieved an average performance improvement of 10.3% (AUPRC), 104.7% (RP9), and 6.3% (F1 score), respectively, in experiments using five pre-built map publicly available.

## REFERENCES

- [1] S. Se, D. Lowe, and J. Little, "Local and global localization for mobile robots using visual landmarks," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. Expanding Societal Role Robot. Next Millennium (Cat. No. 01CH37180)*, 2001, vol. 1, pp. 414–420.
- [2] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 1271–1278.
- [3] S. Carpin, "Fast and accurate map merging for multi-robot systems," *Auton. Robots*, vol. 25, no. 3, pp. 305–316, 2008.
- [4] M. Bosse and R. Zlot, "Keypoint design and evaluation for place recognition in 2D LIDAR maps," *Robot. Auton. Syst.*, vol. 57, no. 12, pp. 1211–1224, 2009.
- [5] G. D. Tipaldi, L. Spinello, and W. Burgard, "Geometrical flirt phrases for large scale place recognition in 2D range data," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 2693–2698.
- [6] M. Himstedt, J. Frost, S. Hellbach, H.-J. Böhme, and E. Maehle, "Large scale place recognition in 2D LIDAR scans using geometrical landmark relations," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2014, pp. 5030–5035.
- [7] G. D. Tipaldi and K. O. Arras, "Flirt-interest regions for 2D range data," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2010, pp. 3616–3622.
- [8] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.
- [9] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded up robust features," in *Proc. Eur. Conf. Comput. Vis.*, Springer, 2006, pp. 404–417.
- [10] S. Lowry *et al.*, "Visual place recognition: A survey," *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.
- [11] Y. Li and E. B. Olson, "Extracting general-purpose features from LIDAR data," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2010, pp. 1388–1393.
- [12] A. Millane, H. Oleynikova, J. Nieto, R. Siegwart, and C. Cadena, "Free-space features: Global localization in 2D laser SLAM using distance function maps," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2019, pp. 1271–1277.
- [13] A. J. Millane *et al.*, "Freetures: Localization in signed distance function maps," *IEEE Robot. Automat. Lett.*, to be published, doi: [10.1109/LRA.2021.3052388](https://doi.org/10.1109/LRA.2021.3052388).
- [14] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Matching 3D models with shape distributions," in *Proc. Int. Conf. Shape Model. Appl.*, 2001, pp. 154–166.
- [15] S.-Y. An, L.-K. Lee, and S.-Y. Oh, "Ceiling vision-based active slam framework for dynamic and wide-open environments," *Auton. Robots*, vol. 40, no. 2, pp. 291–324, 2016.
- [16] S. Lee, "Symmetry-driven shape description for image retrieval," *Image Vis. Comput.*, vol. 31, no. 4, pp. 357–363, 2013.
- [17] H. Breu, J. Gil, D. Kirkpatrick, and M. Werman, "Linear time euclidean distance transform algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 5, pp. 529–533, May 1995.
- [18] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2007.
- [19] I. Ardiyanto and J. Miura, "Visibility-based viewpoint planning for guard robot using skeletonization and geodesic motion model," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2013, pp. 660–666.
- [20] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN," *ACM Trans. Database Syst.*, vol. 42, no. 3, pp. 1–21, 2017.
- [21] A. Howard and N. Roy, "The robotics data set repository (radish)," 2003. [Online]. Available: <http://radish.sourceforge.net/>
- [22] T. Y. Zhang and C. Y. Suen, "A fast parallel algorithm for thinning digital patterns," *Commun. ACM*, vol. 27, no. 3, pp. 236–239, 1984.
- [23] E. W. Dijkstra *et al.*, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [24] N. Kiryati, Y. Eldar, and A. M. Bruckstein, "A probabilistic hough transform," *Pattern Recognit.*, vol. 24, no. 3, pp. 303–316, 1991.
- [25] E. Choi and C. Lee, "Feature extraction based on the Bhattacharyya distance," *Pattern Recognit.*, vol. 36, no. 8, pp. 1703–1709, 2003.
- [26] S. Belongie, J. Malik, and J. Puzicha, "Shape context: A new descriptor for shape matching and object recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, 2000, vol. 13, pp. 831–837.
- [27] J. Deray, J. Sola, and J. Andrade-Cetto, "Word ordering and document adjacency for large loop closure detection in 2-D laser maps," *IEEE Robot. Automat. Lett.*, vol. 2, no. 3, pp. 1532–1539, Jul. 2017.