Hierarchical probabilistic fusion framework for matching and merging of 3-D occupancy maps

Yue, Yufeng; Senarathne, P. G. C. Namal; Yang, Chule; Zhang, Jun; Wen, Mingxing; Wang, Danwei

2018

https://hdl.handle.net/10356/106304

https://doi.org/10.1109/JSEN.2018.2867854

Hierarchical Probabilistic Fusion Framework for Matching and Merging of 3-D Occupancy Maps

Yufeng Yue, P. G. C. Namal Senarathne, Chule Yang, Jun Zhang, Mingxing Wen, and Danwei Wang, *Senior Member, IEEE*

Abstract—Fusing 3-D maps generated by multiple robots in real/semi-real time distributed mapping systems are addressed in this paper. A 3-D occupancy grid-based approach for mapping is utilized to satisfy the real/semi-real time and distributed operating constraints. This paper proposes a novel hierarchical probabilistic fusion framework, which consists of uncertainty modeling, map matching, transformation evaluation, and map merging. Before the fusion of maps, the map features and their uncertainties are explicitly modeled and integrated. For map matching, a two-level probabilistic map matching (PMM) algorithm is developed to include high-level structural and low-level voxel features. In the PMM, the structural uncertainty is first used to generate a coarse matching between the maps and its result is then used to improve the voxel level map matching, resulting in a more efficient and accurate matching between maps with a larger convergence basin. The relative transformation output from PMM algorithm is then evaluated based on the Mahalanobis distance, and the relative entropy filter is used subsequently to integrate the map dissimilarities more accurately, completing the map fusion process. The proposed approach is evaluated using map data collected from both simulated and real environments, and the results validate the accuracy, efficiency, and the support for larger convergence basin of the proposed 3-D occupancy map fusion framework.

Index Terms—Map uncertainty modeling, hierarchical probabilistic fusion, information integration, multi-robot system.
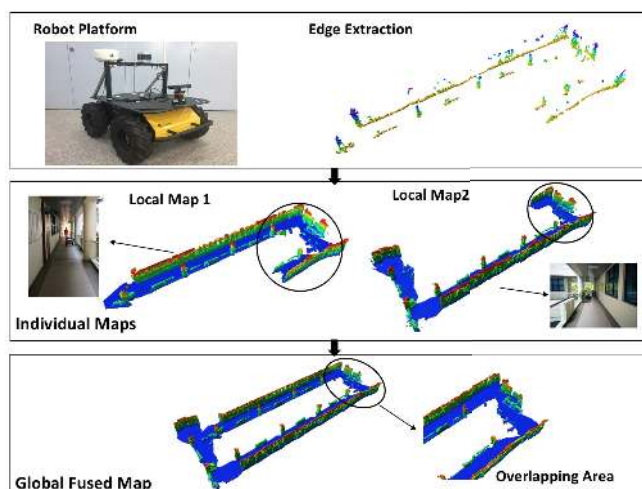
Fig. 1. Top: Husky A200 robot platform equipped with Velodyne VLP-16 and the 3D edges extracted from the map. Middle: two individual maps of the WKW environment with the common area (black circle). Bottom: the fused global map and the zoomed in image of the overlapping area.

I. INTRODUCTION

DETAILED 3D mapping of unknown or partially known environments is a major research problem in mobile robotics that has garnered significant attention in recent years with the improvements in 3D sensors, 3D mapping algorithms, and processing systems. Many modern autonomous mobile robotic applications ranging from environmental mapping [1], [2], surveillance [3] missions to autonomous vehicles require a detailed 3D map of the environment as a prerequisite for their optimal operation. Since the operating

environment gets larger, efficiency and robustness of map generation become a critical factor [1], [4], [5]. Utilizing multiple robots to conduct the mapping is going to be inevitable. The major challenge in these systems is to fuse the maps generated by individual robots (i.e., local maps) into a globally consistent map (i.e., a global map). This is further exacerbated when the robots are required to operate in a networked environment with real or semi-real time mapping requirement where the local maps are to be transferred over the network for map fusion. This article addresses the 3D map fusion problem in such multi-robot mapping systems with limited network bandwidth.

Real or semi-real time global mapping with limited network bandwidth requires a careful consideration for the choice of map type that balances the requirements of detailed 3D mapping, support for transfer over limited communication and their utility for robust map fusion. Point cloud maps [6], volumetric maps [7] and topological maps [8], [9] are some of the different types of maps that can be used to represent the 3D environment. While point cloud maps provide more detailed information about the environment, they are extremely challenging to store, manage and process and even more so to transfer over networks with limited connectivity. Therefore, the point cloud is not an ideal choice of map type

for semi-real time multi-robot mapping missions. However, the rich details in these maps have encouraged the development of 3D map fusion algorithms [6], albeit the off-line high computational effort. Topological maps, on the other hand, are easier to process and transfer over the network due to their lower memory footprint. The literature contains map fusion algorithms involving 2D topological maps [8]–[10]. However, the high level of abstraction in these maps do not lend them well for detailed 3D environment mapping and hence for 3D map fusion. Metric grid maps such as Volumetric maps [11], [12] can represent the 3D environments in sufficient detail without sacrificing the memory or processor usage. Combined that with the availability of compression schemes that allow semi-real time transferring over limited bandwidth makes them the ideal choice of map type for multi-robot mapping missions. Octomap [12] is a volumetric map type with a high compression ratio that can represent any type of 3D environment, hence is selected as the choice of map type in this research.

Generally, the *multi-robot 3D occupancy grid map fusion* consists of estimating the relative transformation between the local maps (i.e., map matching) and merging the occupancy map information (i.e., map merging), which are usually tackled independently of each other in this particular order [7], [13], [14]. Existing 3D occupancy map matching approaches utilize some form of dense registration algorithm to estimate the relative transform between maps. This is typically realized by converting the occupancy grid map into a point cloud, by representing each lowest level voxel in the 3D map by its center coordinate, and then employing a point set registration algorithm [7]. The map merging step consists of verifying the accuracy of the estimated transformation and fusing the occupancy probabilities of the matching voxels [10]. When a map matching algorithm returns an accurate transformation, the majority of map merging approaches merely resort to either stitching the overlapped voxels or to averaging the occupancy probabilities of matching voxels. While this two-step partitioning of 3D occupancy map fusion into a geometric problem and a probability fusion problem simplifies the solution and has seen significant development in recent years, it also introduces some issues that result in sub-optimal fusion results which need to be resolved. Besides, more detailed evaluation of these algorithms in the form of convergence and sensitivity to local map overlap changes warrants further examination and are the focus of this article.

Dense scan registration algorithms inspired initial solutions to map matching. These algorithms, such as ICP [15], NDT [16], GICP [17], are introduced to operate on raw scan data which are essentially samples of continuous surfaces with some noise. In the mapping process, sensor noise and robot localization errors are arbitrary digitized to generate 3D voxel maps with discontinuous and multiple voxel thick surfaces. Converting these voxel surfaces into the equivalent point clouds will not restore the continuous surfaces of samples as illustrated in Fig. 2. Hence the direct use of registration algorithms in 3D occupancy grid map matching results in sub-optimal voxel associations thus sub-optimal matching. Additionally, occupancy grid map formulation assumes
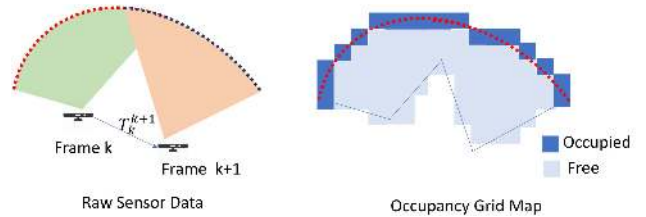

Fig. 2. The left image shows two consecutive scans of a continuous surface collected by the sensor. During the mapping process, motion uncertainty and sensor noise are accumulated and updated to a discontinuous occupancy grid map in the right image.

voxel-wise independence to make the map update process computationally tangible. Ignoring the correlations among neighboring voxels with structural information also results in sub-optimal matching results.

This paper addresses these issues by extending our previous work on map matching with occupancy probability [14] by introducing a probabilistic map matching approach where the uncertainties of the structural features and individual voxels are modeled and used during matching in a two-step approach. This increases the convergence rate of the optimization process as the use of local uncertainty prevents outliers from dominating the solution. A statistical testing model is applied to evaluate the output of matching with the acceptance threshold auto-tuned based on the overlapping between the maps. A Relative Entropy Filter is employed to improve the occupancy probability merging robustly. In addition to evaluating the accuracy of map matching and merging, the sensitivity of map matching algorithms on map overlapping is also addressed. The experimental results confirm the efficacy of the proposed approach in generating more accurate relative transformation estimates between maps with a faster convergence rate and a larger convergence basin. These advantages are further demonstrated by the ability of the fusion framework to generate global occupancy grid maps with reduced uncertainty.

The rest of the paper is organized as follows: Section II reviews the existing approaches to 3D occupancy map fusion and related algorithms. Section III then introduces our first contribution of modeling the uncertainty of 3D occupancy map data at voxel level and structural level as related to map fusion. The two-level probabilistic map matching framework based on the uncertainty modeling is detailed in Section IV. Section V presents the proposed statistical testing for transformation estimation and the relative entropy-based map merging approach. Section VI presents the results of the experiments conducted using both simulated and real data. Section VII concludes the paper with a discussion on future work.

II. RELATED WORK

This section reviews the existing work in the literature related to 3D map fusion. As the majority of the initial solutions to 3D map fusion is inspired by registration algorithms, sensor data based dense registration algorithms are reviewed first. Then, the approaches of map matching and merging are reviewed, where the strengths and limitations are carefully discussed.

A. Dense Registration of Sensor Data

The objective of dense registration algorithm is to find a 3D rigid transformation to register the consecutive sensor scans. To establish the point-wise correspondence, dense registration directly operates on the raw data obtained by range sensors such as Kinect and laser scanners. The well known ICP [15] is the most significant algorithm, which iteratively searches for corresponding pairs and computes the transform that minimizes the Euclidean distance. Since the introduction of ICP, many variants based on ICP have been proposed to improve the aspects like pre-filtering [18], [19], data association [17], [20]–[26], outlier removal [27], [28] and error function optimization [16], [29]–[31].

Before registration, scan data is usually preprocessed to extract features like planar patches [18], or downsampling [19] to increase the efficiency. To find the data association, ICP assumes that the points from two sources have physical spatial correspondence. However, this is violated in the presence of sensor noise and limited data density. To model the sensor noise, a probabilistic version is proposed in [22], where points are represented by a normal distribution. In [23], probabilistic data association is applied to align two point clouds with different densities. While purely using position information might fall victim to correspondence ambiguity, additional information extracted from the point cloud like local planar structure [17], curvature&intensity [25], color information [26] are incorporated as multi-channel sources.

Once the data association is established, tuning the outlier filter is a critical task for the success of the alignment. To filter out the outliers, each match is weighted by the inverse of the distance between the points involved in the matching process in [27]. In [28], outlier filtering is automatically tuned concerning the overlap variations. Based on the actual correspondence, many ICP variations use a closed form solution to iteratively compute the alignment based on the cross-correlation between point clouds [16]. To allow more generic minimization function other than Euclidean distance, nonlinear optimization framework is introduced to add additional constraints and dimensions based on $L2$ norm distance [29], such as more penalization along the planar patch direction [30] or edge direction [31].

B. Map Matching

Over the last years, various research groups have developed efficient solutions for the map matching problem. These techniques range from 2D grid map matching [9], [10], [32]–[34], to sparse feature descriptors [35]–[38], to full 3D dense occupancy map registration [6], [7], [14], [39]–[41]. Here, this work focuses on map matching and makes the assumption that the local map generated by each robot is consistent.

For 2D grid maps, topology-based approaches like hough transform [32] and Voronoi diagram [9], [34] are applied to represent occupancy map on the abstract level. Besides, the probabilistic Voronoi matching is introduced to incorporate the motion uncertainty [10]. For feature-based maps, Delaunay triangulation [36] based topology network is adopted to perform matching.
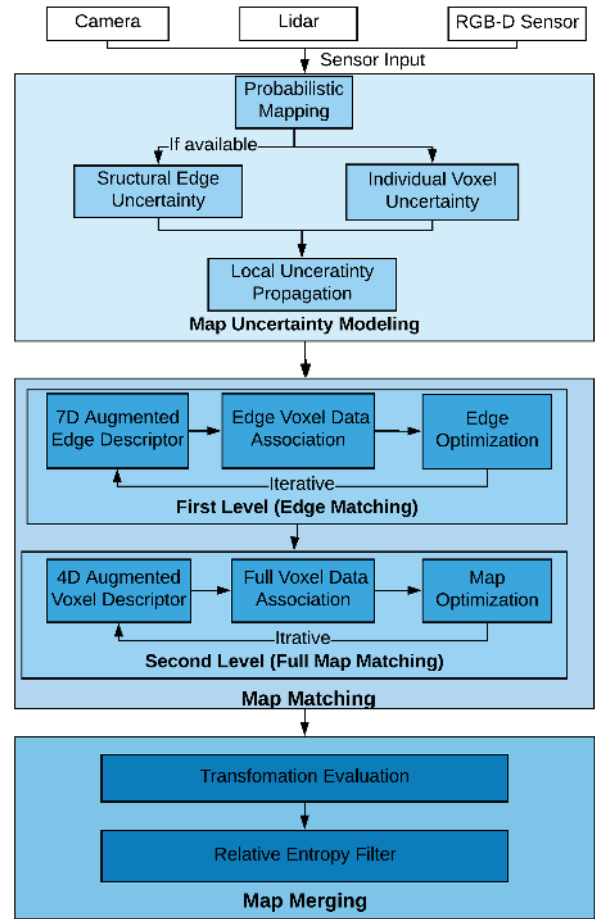

Fig. 3. The diagram of the system. The focus of this paper is to design the hierarchical system for map fusion, which can be divided into three modules, i.e., uncertainty modeling, map matching and map merging.

With the increasing availability of high-quality 3D sensors, the dense registration algorithms have been employed to match maps composed of raw sensor data. In [39], probabilistic hypothesis density(Ph.D.) SLAM is employed to match sparse features in the simulation. CSHOT [42] descriptor and Iterative Closest Point (ICP) [15] are integrated for submap-based matching [41]. For back-end optimization, [6] and [40] integrate raw sensor data to construct submaps and optimize constrained graph structures. However, aforementioned 3D map matching approaches are based on raw sensor data instead of probabilistic maps.

Since 3D occupancy grid map is generated by compressing the raw sensor data with a sensor measurement model [12], observation noise and motion uncertainty have been embedded into the map. The fusion of 3D probabilistic map is firstly proposed in [7], which utilizes ICP [15] to calculate the relative transformation by minimizing point-wise Euclidean distance. As shown in Fig.2, the raw sensor data generated from the simulated environment are highly compressed to occupied voxels. Hence, the available characteristics of the occupancy map should be modeled and combined into the registration process. Based on this observation, probabilistic
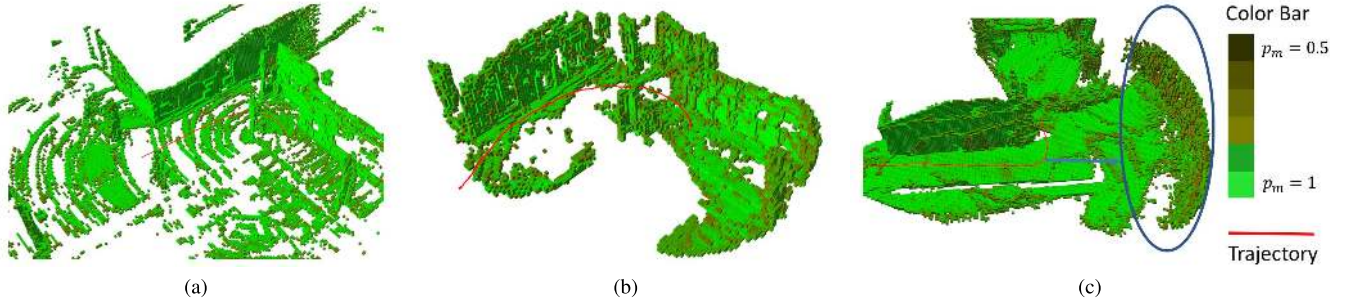
Fig. 4. Maps generated by different sensors, where the color bar represents the occupancy probability. Fig. 4a and 4b are the maps generated in the same environment with the same SLAM algorithm, hence represents the sensor uncertainty. The map generated by Velodyne has a smaller uncertainty compared with ZED stereo camera. Fig. 4c is generated using ASUS RGB-D camera, which shows the map uncertainty caused by localization error when a sharp turn happens. (a) Map generated using Velodyne VLP-16. (b) Map generated using ZED camera. (c) Map generated using ASUS RGB-D camera.

map information is incorporated to improve the accuracy of the estimated transformation in [14].

C. Map Merging

As for registration algorithms, there is no guarantee that a global minimum will be found. So, a model to verify the transformation is essential. In [43], an environment measurement model is proposed to verify the accuracy of the transformation by calculating the percentage of inliers.

To test the consistency of matched results, [44] models the histogram of the misalignment error metric into a truncated Gaussian distribution to evaluate the consistency.

When a robust and accurate transformation is achieved, the key issue is to integrate the probabilistic information in partial maps into a globally consistent map. In many of the previous methods, the final maps are generated by directly stitching the overlapped data [41] or averaging the occupancy probability value of corresponding pairs on voxel-wise level [7], where uncertainty propagation and merging are ignored. To measure the uncertainty of the probabilistic map, Shannon entropy [45] is used to measure the uncertainty of the map in robot exploration. In [10], the paper introduces an entropy filter to reject fusion leading to a higher entropy but is overconfident on the single map without considering both maps. Since the merging process should not introduce inconsistency for the global map, the Kullback-Leibler (KL) divergence [14] based relative entropy filter is adopted to fuse occupancy probability consistently.

III. MAP UNCERTAINTY MODELING

This section exhibits the probabilistic mapping process and an efficient structural information extraction algorithm. Furthermore, the process of propagating and merging individual voxel uncertainty and structural edge uncertainty is presented.

A. Individual Voxel Uncertainty

Given an occupancy grid map $M = \{M_i\}_1^N$, the information extracted from each voxel $M_i = (m_x^i, m_y^i, m_z^i, d_m, p_{m_i})$ is defined as a tuple that includes the position of the extracted voxel center $m_i = (m_x^i, m_y^i, m_z^i)$, the occupancy probability

value p_{m_i} and the voxel size d_m. Based on the independence assumption, the probability of a leaf node m_i given observation $z_{1:t}$ is updated by applying the Bayes filter in [46] (see Eq.(1)). The occupancy probability will update recursively when new sensor measurements are observed.

$$
\begin{aligned}
p(m_i|z_{1:t}^i) & \\
= \left(1 + \frac{1 - p(m_i|z_t^i)}{p(m_i|z_t^i)} \frac{1 - p(m_i|z_{1:t-1}^i)}{p(m_i|z_{1:t-1}^i)} \frac{p(m_i)}{1 - p(m_i)}\right)^{-1} & \\
\propto \underbrace{p(m_i|z_t^i)}_{\text{sensor model}} \cdot \underbrace{p(m_i|z_{1:t-1}^i)}_{\text{previous estimation}} & \quad (1)
\end{aligned}
$$

The updating of Eq.(1) depends on the current observation $p(m_i|z_t^i)$, the previous estimation $p(m_i|z_{1:t-1}^i)$ and the prior probability $p(m_i)$. The term $p(m_i|z_t^i)$ is the sensor model that represents the sensor uncertainty. Fig. 4a and 4b show the map generated in the same environment with different sensors, which represents the uncertainty caused by the specification of the sensors. Since the raw sensor data registration only considers the sensor uncertainty, what makes probabilistic map matching different from sensor data dense matching is the term $p(m_i|z_{1:t-1}^i)$. The previous estimation term $p(m_i|z_{1:t-1}^i)$ models the accumulation of previous observations and motion uncertainty. As shown in Fig. 4c, large error with several thicks of walls is introduced when a sharp turned happens.

To simplify the calculation, the probability of voxel m_i given observation $z_{1:t}^i$ is updated using log-odds in Eq.(2):

$$
L(m_i|z_{1:t}^i) = L(m_i|z_{1:t-1}^i) + L(m_i|z_t^i) \quad (2)
$$

Note that log-odds can be easily converted to probabilities using Eq.(3).

$$
L(m_i|z_{1:t}) = \log \frac{p(m_i|z_{1:t})}{1 - p(m_i|z_{1:t})} \quad (3)
$$

In probabilistic updating process, the only output is the occupancy probability value $p(m_i|z_{1:t}^i)$, where sensor readings and previous estimations are integrated into occupancy probability. To model the covariance of a voxel, the inverse of occupancy probability is utilized to represent the spherical covariance of a voxel. The Gaussian distribution $\mathcal{N}_{p_i}(\mu_{m_i}, \Sigma_{p_i})$ is located at
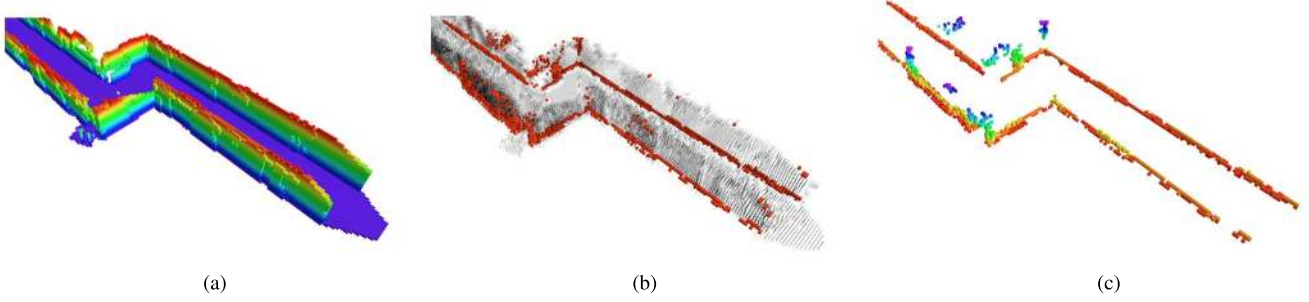
Fig. 5. The left image shows the input 3D occupancy grid map generated in an indoor hallway, then the middle image presents the computed normals and detected 3D edge voxels from the 3D map. Finally, the outliers are removed and outputs the 3D edge voxels in the right image. (a) The input 3D occupancy grid map. (b) The detected 3D edge voxels. (c) Outlier removal and outputs 3D edge voxels.

the center of the voxel and identical in each direction, as shown in Fig.6.

$$\Sigma_{p_i} = \begin{bmatrix} p_{m_i}^{-1} & 0 & 0 \\ 0 & p_{m_i}^{-1} & 0 \\ 0 & 0 & p_{m_i}^{-1} \end{bmatrix} \tag{4}$$

B. Structural Edge Uncertainty

The edge voxels in a 3D occupancy grid map are the voxels in the 3D map that exhibit a sudden large change in local curvature (i.e. sudden bend in the surface) and are extracted following the steps described in Alg. 1. First the surface normals of the occupied voxels are computed (lines 2-6). This is

---

Algorithm 1 Structural Edge Voxel Detection

Require: M^O : The occupied voxels of the 3D grid Map
1: $E = \phi$
2: **for** each $m_i \in M^O$ **do**
3:    $N_i = \{m_j : m_j \in M^O \text{ and distance}(m_i, m_j) < r_N \}$
4:    $(\Lambda, V) = \text{PCA}(\mathcal{N}_i)$
5:    $n_i = V[3]$
6: **end for**
7: **for** each $m_i \in M^O$ **do**
8:    **if** $\frac{\sum_{m_j \in \mathcal{N}_i} |\mathbf{n}_j \times \mathbf{n}_i|}{|\mathcal{N}_i|} > \kappa$ **then**
9:      create $W = (\mathbf{u}, \mathbf{v}, \mathbf{n}_i)$ right handed local coordinate frame at m_j
10:      **for** $\theta = \{0, \delta, \ldots, \pi - \delta, \pi\}$ **do**
11:        $\mathbf{v}_1^W = (cos(\theta), sin(\theta), 0, 1)^T$
12:        $\mathbf{v}_2^W = -v_1^W$
13:        $\mathbf{v}_1^M = T_W^M v_1^W, \mathbf{v}_2^M = T_W^M v_2^W$
14:        $t_1^M = \text{TraceVectorAlongSurfaceVoxels}(m_i, v_1^M, d_T)$
15:        $t_2^M = \text{TraceVectorAlongSurfaceVoxels}(m_i, v_2^M, d_T)$
16:        **if** $\alpha_{min} < acos(\mathbf{t}_1^M \cdot \mathbf{t}_2^M) < \alpha_{max}$ **then**
17:          $E = E \cup \{m_i\}$
18:          **break**
19:        **end if**
20:      **end for**
21:    **end if**
22: **end for**
23: $E = StatisticalOutlierRemoval(E)$
24: **return** E

---

done by performing Principal Component Analysis (PCA) on the local neighborhood N_i of each occupied voxel m_i, defined by the radius r_N (line. 4). The output of this operation are the eigen values $\Lambda = \{\lambda_1, \lambda_2, \lambda_3\}$ in decreasing order and the corresponding principal component vectors $\mathbf{V} = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$. The surface normal is $\mathbf{v}_3$ which is the principal component corresponding to the smallest covariance (i.e. hence smallest eigen value λ_3) (line. 5) [47]. Voxels with surface normals that on average has an angle cost larger than a given threshold κ to the surface normals within their neighboring voxels are selected as candidate edge voxels (line. 8). The average angle cost of a surface normal n_i with respect to its neighboring surface normals is computed using Eq. 5, where the angle cost between two normal vectors is computed using their cross product.

$$\overline{Cost}_\theta(n_i) = \frac{\sum_{m_j \in N_i} |\mathbf{n}_j \times \mathbf{n}_i|}{|N_i|} \tag{5}$$

A validity check is performed (lines. 9-20) in order to detect valid edge voxels from the candidate edge voxels. During this check, vectors are traced along the surface from each candidate voxel m_i in all directions of the local plane up to a distance d_T. Steps for initial trace vector generation for each m_i is given in lines. 11-13 where T_W^M is the transformation matrix from local frame W to global map frame M. t_1^M is the resultant vector after initial vector v_1^M is traced along the voxel surface from m_i up to d_T distance (line. 14). Value of 0.5m for d_T is used. If the resultant vectors t_1^M, t_2^M of any two opposing initial trace vectors v_1^M, v_2^M falls within the angle range $(\alpha_{min}, \alpha_{max})$, signifying a sudden bend in the local surface, m_i is detected as an edge voxel. Angle range is typically set to $(80°, 100°)$. The detected voxels in E are statistically filtered (line. 23) to remove any remaining outliers to generate the edge voxels of the 3D map. Output of the edge extraction process is depicted in Fig. 5.

Given the detected edge voxels, local uncertainty Σ_{τ_i} of each edge voxel m_i is derived using the neighboring edge voxels. Then, the spatial Gaussian model $\mathcal{N}_{\tau_i}(\mu_{m_i} \Sigma_{\tau_i})$ is located at the center of the edge voxel μ_{m_i}. By performing eigen decomposition, covariance Σ_{τ_i} is factorized into $\Sigma_{\tau_i} = \vec{v}_i \lambda_i \vec{v}_i^T$. The eigen values $\lambda_i = \text{diag}(\lambda_i^1, \lambda_i^2, \lambda_i^3)$ in ascending order correspond to eigen vectors $\vec{v}_i = [\vec{v}_i^1, \vec{v}_i^2, \vec{v}_i^3]$, where the
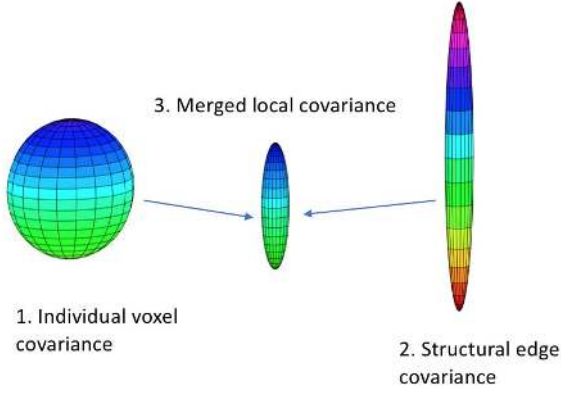
Fig. 6. The process of combing individual voxel covariance (left) and structural edge covariance (right), where the middle image shows the merged result (middle).

edge vector τ_i equals to $\vec{v_i^3}$. The edge covariance is shown in Fig. 6.

$$\Sigma_{\tau_i} = \begin{bmatrix} \vec{v_i^1} & \vec{v_i^2} & \vec{v_i^3} \end{bmatrix} \cdot \begin{bmatrix} \lambda_i^1 & & \\ & \lambda_i^2 & \\ & & \lambda_i^3 \end{bmatrix} \cdot \begin{bmatrix} \vec{v_i^1} & \vec{v_i^2} & \vec{v_i^3} \end{bmatrix}^T \tag{6}$$

C. Local Uncertainty Propagation

The significance of the structural information reconstruction comes at that, although the occupancy map is assumed to be voxel-wise independent, the continuous structural information is recovered by extracting the edges from the noisy map data.

Hence, with the individual voxel covariance $\mathcal{N}_{p_i}(\mu_{m_i}, \Sigma_{p_i})$ and structural edge model $\mathcal{N}_{\tau_i}(\mu_{m_i}, \Sigma_{\tau_i})$ available, the two uncertainty models are integrated into an unified representation of the local uncertainty model $\mathcal{N}(\mu_{g_i}, \Sigma_{g_i})$:

$$\mathcal{N}(\mu_{g_i}, \Sigma_{g_i}) \sim \mathcal{N}(\mu_{m_i}, \Sigma_{p_i}) \cdot \mathcal{N}(\mu_{m_i}, \Sigma_{\tau_i}) \tag{7}$$

Where Eq.(7) is the weighted averaging algorithm with the updated mean and covariance:

$$\mu_{g_i} = (\Sigma_{p_i}^{-1} + \Sigma_{\tau_i}^{-1})^{-1}(\Sigma_{p_i}^{-1}\mu_{m_i} + \Sigma_{\tau_i}^{-1}\mu_{m_i}) = \mu_{m_i} \tag{8}$$

$$\Sigma_{g_i} = (\Sigma_{p_i}^{-1} + \Sigma_{\tau_i}^{-1})^{-1} \tag{9}$$

The process of integrating the uncertainty is shown in Fig. 6. The benefit of the uncertainty modeling and merging comes at two parts. Firstly, it accurately models the local uncertainty of the map with a properly designed Gaussian ellipse, which preserves the most valuable information in the environment. Secondly, the fusion combines the individual voxel covariance and structural edge covariance to represent the final uncertainty.

IV. TWO-LEVEL PROBABILISTIC MAP MATCHING

To address 3D volumetric map registration, the two-level Probabilistic Map Matching (PMM) architecture is proposed. We start with formulating the two-level structure of the map matching algorithm, which starts from high-level edge features

and moves to low-level full voxel matching. Then, map matching is further solved by decomposing it into establishing the probabilistic data association and optimizing the rigid transformation based on the data association.

A. The Formulation of Two-Level Probabilistic Map Matching Problem

1) Mathematical Definition: **Problem** *Given a set of 3D occupancy grid maps, each map can be denoted as $M = \{m, g\}$, where the point cloud $m \triangleq \{m_i\}_1^N$ and geometry features $g \triangleq \{g_i\}_1^L$ are extracted from the map M. The objective is to estimate the relative transformation T that maximizes the overlap between two maps.*

For dense registration of two maps M_r and M_s, the common solution is to fix one map M_r as the model map, and another map M_s as the scene map, where the rigid transformation T transforms the map M_s to the coordinate frame of M_r. The maximization of the overlapping between M_r and M_s is formulated as:

$$T = -\min_T \ \log \ p(M_r|T, M_s) \tag{10}$$

where $p(M_r|T, M_s)$ is the maximum likelihood estimation (MLE) that aims to find the most likely relative transformation T by matching the two partial maps M_r and M_s. The estimated matrix $T \in SE(3)$ is a rigid transformation matrix with rotation matrix $R \in SO(3)$ and translation vector $t \in \mathbb{R}^3$. The transformation T is iteratively minimized until a certain criteria is met.

2) Two-Level Map Matching Formulation: Here, we define the map M_r and M_s as 3D occupancy grid map and derive the structure for two-level matching. For the model map M_r, we denote full point cloud $m_r = \{m_r^i\}_{i=1}^{N_{m_r}}$ and geometry feature $g_r = \{g_r^i\}_{i=1}^{N_{gr}}$. In this paper, the geometry feature g_r^i refers to the coordinate of the voxel that belongs to the extracted edge. Instead of simply considering T as a fixed unknown parameter, it is modeled as a 6D random variable and estimated with maximum likelihood estimation (MLE). The registration of M_r and M_s formulated in Eq.(10) is computed in two levels, where edge matching and full map voxel matching are tackled sequentially in Eq.(11).

$$T = \min_T \left(\underbrace{- \log \ p(g_r|T, g_s)}_{\text{Level 1: Edge Matching}} - \underbrace{\log \ p(m_r|T, m_s)}_{\text{Level 2: Full Map Matching}} \right) \tag{11}$$

The two-level structure works sequentially. Firstly, in the first level, the extracted edges are matched by utilizing the weighted linear combination of positional distance, occupancy probability distance and edge direction distance into an overall distance metric. Then, the probabilistic quadratic function is solved with the nonlinear optimization algorithm. In the second level, full voxel maps are matched to achieve accurate transformation and derive the corresponding pairs for probability merging. Here, the paper assumes voxel-wise correspondences to be independent, the probability density function in Eq.(11) can be simplified into Eq.(12), where $m_s^{c_i}$ and $g_s^{c_j}$ are

the candidate corresponding voxels of m_r^i and g_s^i, respectively.

$$T = \min_T \left(\underbrace{-\sum_j \log\ p(g_r^j|T, g_s^{c_j})}_{\text{Level 1: Edge Matching}} \underbrace{-\sum_i \log\ p(m_r^i|T, m_s^{c_i})}_{\text{Level 2: Full Map Matching}} \right) \quad (12)$$

The problem of estimating the transformation matrix is often solved in two steps: data association and nonlinear optimization. Since the data association is unknown, the problem in Eq.(12) is iteratively estimated. First, the data association step efficiently estimates the hidden corresponding pairs by evaluating the similarity of the local descriptor. Then, the optimization step solves the rigid transformation based on the previous step by marginalizing the likelihood function.

B. Probabilistic Data Association

To estimate transformation T, data association between the model voxel m_r^i (g_r^j) and scene voxel $m_s^{c_i}$ ($g_r^{c_j}$) needs to be established, which is usually estimated by performing unidirectional nearest neighborhood search of Euclidean distance.

With the presence of map noise, its hard to find the true correspondence with purely coordinate information [15]. In [25] and [26], geometrical features such as curvature, surface normal and point cloud density are introduced to find the accurate correspondence relationship.

In our previous work [14], occupancy probability is incorporated to establish voxel-to-voxel correspondences between all overlapped partial maps in $4D$ space. With the structural information extracted from Sec.III, it will provide additional information to restore the corresponding relationship between two occupancy grid maps. Here, edge matching is augmented to $7D$ local descriptor in the first level with $\{m_i^x, m_i^y, m_i^z, p_{m_i}, \tau_i^x, \tau_i^y, \tau_i^z\}$, which includes the center coordinate of voxel ($3D$), occupancy probability ($1D$) and edge direction ($3D$).

The first level of the registration applies full 7D local descriptor with the distance metric defined in Eq. (13), the distance metric $d(g_r^i, g_s^j)$ is described by Euclidean point distance $d_e(g_r^i, g_s^j)$, occupancy probability distance $d_o(g_r^i, g_s^j)$ and edge geometry distance $d_\tau(g_r^i, g_s^j)$. The weights for the corresponding pair search are inversely proportional to the variance of the particular channel of the source information. The weighting vector $\omega = \{\omega_o, \omega_\tau\}$ is introduced to determine the importance of the occupancy probability and geometry distance relative to the positional distance.

$$d(g_r^i, g_s^j) = d_e(m_r^i, m_s^j) + \omega_o d_o(p_{m_r^i}, p_{m_s^j}) + \omega_\tau d_\tau(\tau_r^i, \tau_s^j)$$
$$= ||T \oplus m_r^i - m_s^j||^2 + \omega_o||p_{m_r^i} - p_{m_s^j}||^2$$
$$+ \omega_\tau||T \oplus \tau_r^i - \tau_s^j||^2 \quad (13)$$

When it moves to the full map matching level, the structure information is not available. In this case, the dimension of descriptor reduced to $\{m_i^x, m_i^y, m_i^z, p_{m_i}\}$ in 4D space, which is

the same as [14]. The error metric defined in Eq.(14) denotes the error metric of full map matching, where $\omega_\tau = 0$.

$$d(m_r^i, m_s^j) = d_e(m_r^i, m_s^j) + \omega_o d_o(p_{m_r^i}, p_{m_s^j})$$
$$= ||T \oplus m_r^i - m_s^j||^2 + \omega_o||p_{m_r^i} - p_{m_s^j}||^2 \quad (14)$$

The weighting vector $\omega = \{\omega_o, \omega_\tau\}$ measures the importance of the occupancy probability and geometry distance relative to the positional distance. The weight parameters $\{\omega_o, \omega_\tau\}$ are scale factors to determine the contribution of occupancy probability and geometry distance. To model the two weights $\{\omega_o, \omega_\tau\}$, the uncertainties of occupancy probability, structural information and positional error are estimated as described below in Sec. IV-B(1-3). Computation of the overall weights based on these uncertainties is detailed subsequently Sec.IV-B.4.

1) The Uncertainty of Occupancy Probability: As 3D probabilistic map is generated using SLAM pose estimates and noisy sensor data, there is always an inherent level of uncertainty associated with the map. When the uncertainty of the map is high, we put less confidence in the probability value. Therefore, Shannon entropy is introduced to measure the uncertainty of the probabilistic voxel map.

$$H_m = -\sum_i^n \frac{p_{m_i} \log\ p_{m_i}}{n} \quad (15)$$

When the voxel is occupied certainly with $p_{m_i} = 1$, the entropy achieves minimum with zero, which means there is no uncertainty. On the contrary, entropy reaches maximum value when the occupancy probability $p_{m_i} = 0.5$. During the matching process, H_m is constant given input of two occupancy grid maps.

2) The Uncertainty of Structural Information: For structural spatial covariance Σ_{τ_i}, the corresponding eigenvalues of $\lambda^i = \text{diag}[\lambda_i^1, \lambda_i^2, \lambda_i^3]$ is in ascending order. The eigenvalues denote the length of the principal coordinate axes, which are the structural uncertainties along three directions. More specifically, the largest eigenvalue λ_i^3 corresponds to the edge vector τ_i. Here, we define Eq.(16) to represent the uncertainty of the edge.

$$\lambda_\tau = \sum_i^n \frac{\lambda_i^3}{n(\lambda_i^1 + \lambda_i^2 + \lambda_i^3)} \quad (16)$$

When the edge is perfect without any noise on the perpendicular plane, we have $\lambda_\tau = 1$. And the extreme case is the three eigenvalues are equal, and the three axes have the same length, then $\lambda_\tau = \frac{1}{3}$. In the second level of full voxel map registration, $\lambda_\tau = 0$ as no structural information is available.

3) The Uncertainty of Positional Value: As shown in Fig. 7, let d_{TS} be the true distance between model point m_r^i and scene point m_s^j. At each iteration, point m_s^j iteratively attempts to find its ground truth point m_r^i, however would match to its nearest neighboring point m_c^i with a distance d_{CS}. So, there is always an offset between ground truth distance d_{TS} and the closet point distance d_{CS}.

In [48], conditional expectation is applied to model the error due to misalignment in each iteration. The author assumes that
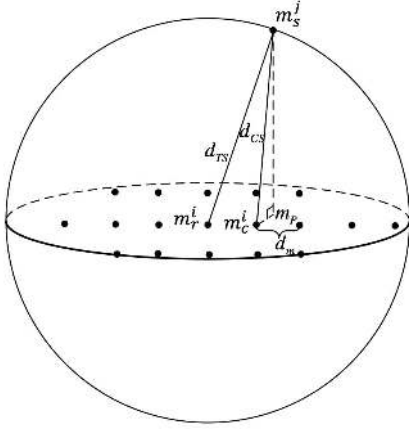
Fig. 7. Positional error model estimated in a spherical coordinate system. The true positional distance(unknown) d_{TS} between m_r^i and m_s^j is estimated using closet point distance(known) d_{CS}, where d_{CS} is computed as the length of the hypotenuse of a right triangle $\triangle_{m_s^j m_c^i m_P}$ and d_{m_i} is the size of the voxel.

the points around the ground truth point m_T are continuously distributed. However, the points are discretized with equal distance d_{m_i} in our case.

To model the true positional distance d_{TS} between m_r^i and m_s^j using closet point distance d_{CS}, the paper assumes the registration positional error is independent of local uncertainty and equal in three directions: $E(d_{TS}^2) = \sigma_{TS}^2 = \frac{1}{3}\sigma_{TSx}^2 = \frac{1}{3}\sigma_{TSy}^2 = \frac{1}{3}\sigma_{TSz}^2$. Assuming the scene point m_s^j is uniformly distributed on the surface of a sphere with the radius of d_{TS}, the probability density function is denoted as $f(d_{CS}^2|d_{TS}) = 1/(4\pi d_{TS}^2)$.

In Eq.(17), the conditional expectation of d_{CS}^2 given d_{TS} is formulated and calculated by integrating in the spherical coordinate system. d_{CS}^2 is computed as the length of the hypotenuse of a right triangle $\triangle_{m_s^j m_c^i m_P}$ as shown in Fig. 7.

$$E(d_{CS}^2|d_{TS}) = \int_S d_{CS}^2 \cdot f(d_{CS}^2|d_{TS})ds \approx \frac{1}{3}d_{TS}^2 \quad (17)$$

In conditional expectation, $E[E(X|Y)] = E(X)$, then the expectation of d_{TS}^2 can be calculated as:

$$E(d_{TS}^2) = E(3E(d_{CS}^2|d_{TS})) \approx 3d_{CS}^2 \quad (18)$$

Since mean square error (MSE) is the global estimate of d_{CS}, then $d_{CS} \approx MSE$. As each dimension is identical, we get $\sigma_{TSx}^2 \approx MSE^2$

4) *Overall Weights:* With the uncertainties of occupancy probability, structural information and positional error estimated in Eq.(15), (16), (18), the weights ω_o, ω_τ defined in Eq.(13), (14) are denoted as:

$$\omega_o = k_1 \cdot \frac{\sigma_{TSx}}{H_m} \approx k_1 \cdot \frac{MSE}{H_m} \quad (19)$$

$$\omega_\tau = k_2 \cdot \sigma_{TSx} \cdot \lambda_\tau \approx k_2 \cdot MSE \cdot \lambda_\tau \quad (20)$$

where k_1 and k_2 are the normalization factors. As the registration iteratively minimizes the error function, MSE is non-increasing. Since weights ω_o and ω_τ are constant in the matching process, the MSE should tend to zero as the
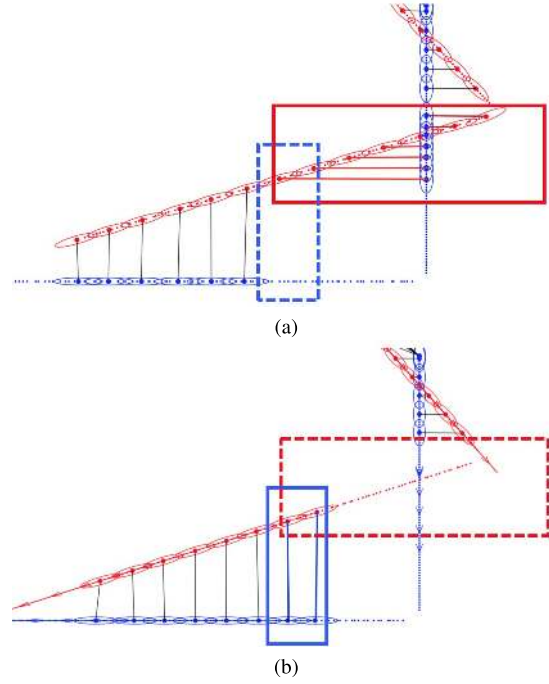

Fig. 8. An example of data association established between two L-shape datasets. In Fig. 8a, it only considers the Euclidean distance. While in Fig. 8b, the 7D augmented descriptor is applied. The full line red box shows the wrong data association estimated in Fig. 8a, which is removed in Fig. 8b. While the black box in the upper image shows the missed data association, which is found and connected with 7D descriptor in Fig. 8b. (a) Data association using Euclidean distance. (b) Data association using 7D augmented distance.

matching converges to a perfect match. When the positional error is large, we should rely more on occupancy probability value and geometric information (only for the first level). As the two maps get closer, the algorithm tends to rely more on positional value.

To show the superiority of our 7D distance metric compared to purely Euclidean distance, an example of data association between two L-shape edges is presented in Fig. 8. In the L-shape datasets, the black and red color denotes two different datasets. As can be seen, the data association using only Euclidean distance based closet neighborhood search falls into the false match, which might lead to wrong convergence in the optimization process. In contrast, the 7D distance metric combined with the augmented information filters out the wrong correspondences and finds correct pairs. The results prove that combining edge information, occupancy probability, and point coordinate tend to find accurate data association.

C. Error Metric Optimization

Based on the two-level structure, the registration strategy starts from the geometry structural edge layer and proceeds to the dense grid voxel layer. The optimization problem is to minimize the error function defined in Eq.(12) with the data associations estimated.

Starting with the edge optimization, robust correspondences have been established, and fewer voxels will be optimized. Applying the prolonged shape of the local uncertainty covariance Σ_{g_i} located along the edge direction will penalize two
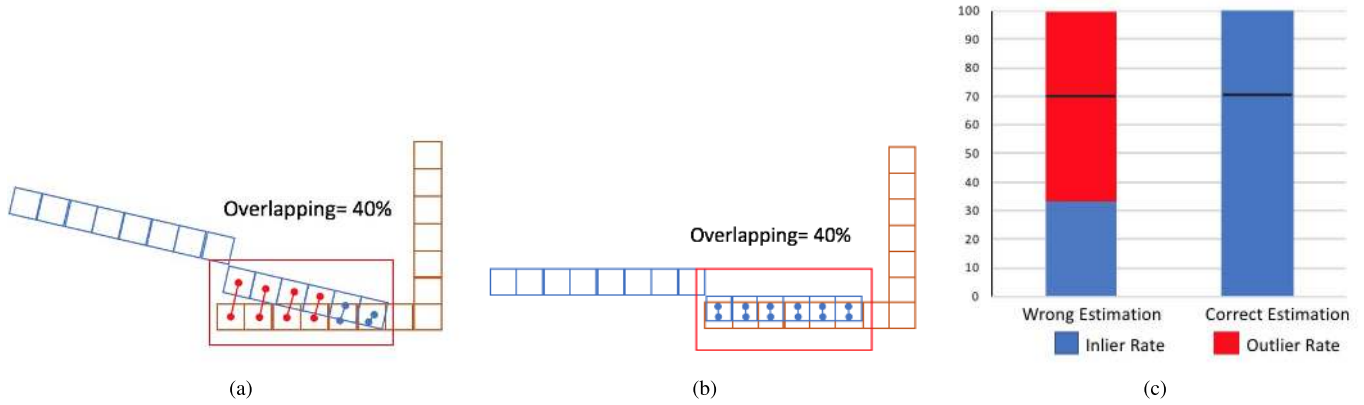
Fig. 9. Given the same datasets with two different rigid transformations. Statistical testing is performed to decide whether to accept the transformation. The inliers are denoted with black, while the outliers are red. In Fig. 9c, the acceptance rate 70% is computed using the overlapping parameter. (a) Wrong rigid transformation. (b) Correct rigid transformation. (c) The plot of inlier and outlier rate, where the acceptance is 70%.

edges not lying in the same direction. The applying of local uncertainty covariance reduces the number of iterations compared to Euclidean distance based optimization algorithm. Then it forces the two edges to have the same direction with positional coordinate aligned, which enhances the convergence basin to inaccurate initial estimation and improves overall registration results.

$$T = \min \sum_j (g_r^j - T \oplus g_s^{c_j})^T (\Sigma_{g_r^j} + R^T \Sigma_{g_s^j} R)^{-1} (g_r^j - T \oplus g_s^{c_j})$$
$$(21)$$

Based on the edge registration outputs, the dense voxel-wise correspondences will be established to refine the estimation. Here, the uncertainty model involved is the individual voxel covariance Σ_{p_i}. Since the covariance is equal in three directions, rotation matrix has no impact on Σ_{p_i}. Then the error function is simplified to the norm of Euclidean distance with the scale factor (see Eq.(22)). Since not all the corresponding error terms should have the same weight, larger weight is assigned to the associations with higher occupancy probability.

$$T = \min \sum_i (p_{m_r^i}^{-1} + p_{m_s^{c_i}}^{-1})^{-1} ||m_r^i - T \oplus m_s^{c_i}||^2 \quad (22)$$

V. TRANSFORMATION EVALUATION AND PROBABILITY MERGING

In this section, the statistical testing of accepting or rejecting the transformation is presented. Then, a relative entropy filter is proposed to measure the map dissimilarities and merge individual maps into a global map.

A. Transformation Evaluation

Although the registration algorithm will achieve convergence, there is no guarantee that a global minimum will be found. Besides, there is a high chance that the algorithm would fall into the local minimum.

In [43], an environment measurement model (EMM) is proposed to verify the transformation by calculating the Mahalanobis distance between matched points. Here, we assume map M_s has been transformed to the coordinate frame of M_r and is denoted as $^r M_s$. Since the edge matching is

a middle process, the evaluation is focused on the voxel-wise correspondences between full map voxels. The statistical evaluation of map M_r given $^r M_s$ is formulated in Eq.(23).

$$p(M_r | ^r M_s) = \prod_i^{N_{m_r}} p(m_r^i | ^r m_s^{c_i})$$
$$(23)$$

A proper inconsistency distance should be defined to describe the discrepancy between submaps. As illustrated in Eq.(22), the probability of each correlated pair can be computed as follows:

$$p(m_r^i | ^r m_s^{c_i}) = exp\left(-\frac{(m_r^i - ^r m_s^{c_i})^T \cdot (m_r^i - ^r m_s^{c_i})}{p_{m_r^i}^{-1} + p_{m_s^{c_i}}^{-1}} \right) \quad (24)$$

The inliers are decided by setting a threshold of Mahalanobis distance. Then, a confidence level is applied as the criteria for rejecting the wrong estimates by computing the overall fraction of inliers. The critical thing is to set a proper threshold for the ratio to accept the transformation. To address this issue, the parameter Θ is introduced to measure the overlapping between the two registered maps. The value is calculated by dividing the number of matched pairs N_{match} by the number of voxels on two maps.

$$\Theta = \frac{1}{2} \cdot \left(\frac{N_{match}}{N_{m_s}} + \frac{N_{match}}{N_{m_r}} \right) \quad (25)$$

Since accepting a wrong rigid transformation will lead to a diverged global map, a conservative acceptance rate is set to be 60%. Besides, as the overlapping decreases, the acceptance rate is enhanced with special care for extreme low overlap cases. The relationship between the overlapping metric Θ and the acceptance threshold of the transformation is defined as follows:

- $\Theta < 20$ %, the inlier percentage should > 80 %
- 20 % $< \Theta < 60$ %, the inlier percentage should > 0.9-$\frac{\Theta}{2}$ %
- $\Theta > 60$ %, the inlier percentage should > 60 %

This simple relationship leads to satisfactory performance in our experiments. An example in Fig. 9 shows the applying of statistical testing efficiently rejects high erroneous estimates that would largely undermine the quality of the fused map.

Notation	Location	Value
κ	Alg. (1)	st as 0.5
$d(g_r^i, g_s^j)$	Eq.(13)	set as 2
$d(m_r^i, m_s^j)$	Eq.(14)	set as 0.5
inlier percentage	Eq.(24)	depending on overlapping metric Θ
$KL(o_{m_r^i}||o_{m_s^{c_i}})$	Eq.(27)	set as 0.1

In the example, the overlapping area is denoted with the red box, and overlapping metric Θ is 40%. Then the inlier threshold should be $0.9 - \frac{\Theta}{2} = 0.7$. In both scenarios, there are six corresponding pairs established. In Fig. 9a, there are four outliers and two inliers, while the six matching pairs are all inliers in Fig. 9b. Since the inlier acceptance rate is 70%, Fig. 9c shows the final results of the transformation evaluation.

B. Relative Entropy Filter

Under sensor noise and motion uncertainty, the same object may have different probabilities in separated maps. Hence it is vital to come up with a strategy to consider the dissimilarities and fuse them into the global probability map.

Kullback-Leibler (KL) divergence is a commonly used relative entropy filter to measure the difference between two random variables. The original form is:

$$KL(P||Q) = \sum_{x \in X} p(x) \log \frac{p(x)}{q(x)} \tag{26}$$

We introduce $o_{m_r^i}$ as a discrete random variable (RV) with two states : occupied and free, where $p(o_{m_r^i} = \text{occupied}) = p_{m_r^i}$, $p(o_{m_r^i} = \text{free}) = 1 - p_{m_r^i}$, and the difference between the two RVs are computed as:

$$KL(o_{m_r^i}||o_{m_s^{c_i}}) = p_{m_r^i} \log \frac{p_{m_r^i}}{p_{m_s^{c_i}}} + (1 - p_{m_r^i}) \log \frac{(1 - p_{m_r^i})}{(1 - p_{m_s^{c_i}})} \tag{27}$$

$KL(o_{m_r^i}||o_{m_s^{c_i}})$ represents the difference between the two probabilities. If KL is within a certain threshold, the dissimilarity is seemed as random errors and the corresponding pairs are fused with weights

$$p_{fuse}^i = \frac{p_{m_r^i}}{p_{m_r^i} + p_{m_s^{c_i}}} p_{m_r^i} + \frac{p_{m_s^{c_i}}}{p_{m_r^i} + p_{m_s^{c_i}}} p_{m_s^{c_i}} \tag{28}$$

If KL is beyond the threshold with $p_{m_r^i} > p_{m_s^{c_i}}$, it means the voxel with lower probability $p_{m_s^{c_i}}$ does not have enough information for the environment and will be rejected, then we only trust the voxel with higher probability and let $p_{fuse}^i = p_{m_r^i}$.

VI. EXPERIMENTAL RESULTS

Experiments conducted in simulated and real-world environments are summarized in this section. Both qualitative and quantitative results are presented in the following aspects:

a) Evaluation of how the proposed approach improves the accuracy in both translation and rotation estimation compared with the baseline algorithms, where fewer iterations required to achieve the same accuracy.

b) Exploration of the effect of reducing the overlap between the two maps showing that extracting the structural information increases convergence.

c) Showing that the efficiency of the proposed algorithm is significantly improved compared to the baseline algorithms, proving that semi-real time performance is guaranteed.

d) Presenting that our algorithm increases the robustness under initial perturbations, where statistic results show the improved performance on accuracy and convergence.

e) Evaluation of how the statistical testing accepts the correct transformation and merges the map using relative entropy filter.

As mentioned before, the research is focused on the fusion stage with an initial relative transform estimation. The initial relative transformation between the coordinate frames of different robots can be extracted during the deployment phase of the mission if such information is available. Else it can be extracted later on through coordinated rendezvous actions [49].

To simulate the noisy initial relative transformations, two maps with different initial transformations both in translation and rotation were generated.

A. Evaluation Protocol

1) Experiment Environment: Experiments conducted using simulated and real data are compared by operating the robot to collect data and generate maps in three different scenarios in Nanyang Technological University (see Tab.II). Each robot was equipped with a Hokuyo Laser range finder for pose estimation using Gmapping [50], Asus Xtion Pro RGB-D camera for 3D perception. In the simulation dataset Willow, the robot was teleoperated to explore environment twice from the same starting point. The Willow is a standard ROS/ Gazebo office world with varying room sizes and corridors. In the real environments Corridor and WKW, the recorded datasets were divided into two segments to generate two different maps with some overlap. The corridor is a long corridor in the building, while the WKW is the lobby of a building with walking people and clutter furniture/objects. The ground truth transformation between the maps was acquired by recording the experimental setup. The resolution of the 3D occupancy grid map was set to be 0.1m.

2) Error Metric: The paper compares the estimated transformation T_e to the ground truth T_g. Then the error $\triangle T$ is calculated as $\triangle T = T_e \cdot T_g^{-1} = \{\triangle R, \triangle t\}$. The 3D translation error $e_t = ||\triangle t||$ and 3D rotation error $e_r = ||\triangle R|| = \arctan \triangle R$ are the Euclidean norm of and Euler norm of the difference between the ground-truth and the output. Another metric MSE (Mean Squared Error) is the mean squared Euclidean distance between the matched pairs.

3) Comparison Baseline: To demonstrate the superiority of the two-level PMM algorithm, the comparison is conducted on two levels. Firstly, the ICP based edge matching is compared against the first level of PMM. Then, the full
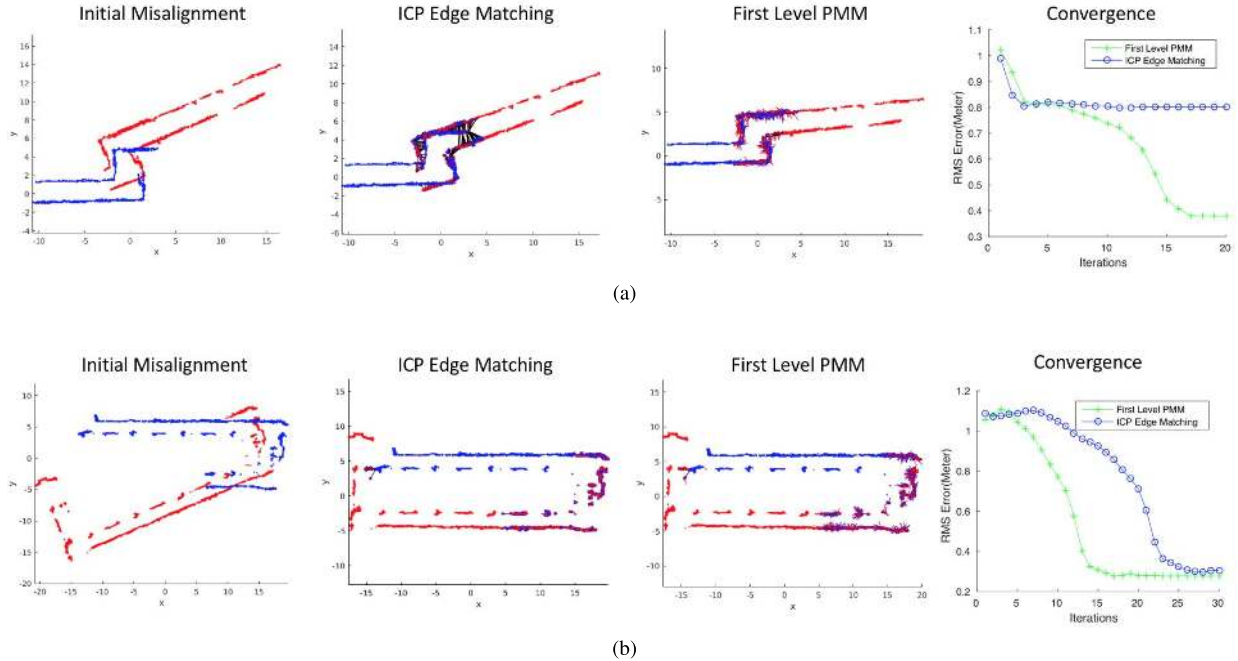
(a)

(b)

Fig. 10. Comparison between edge registration algorithms, where first level of PMM is compared against ICP based edge registration algorithm. (a) Registration results of the edges extracted from Corridor dataset. The black lines show the correspondences established between the two edge maps. (b) Registration results of the edges extracted from WKW dataset. Both algorithms converge to the correct position, where the first level of PMM converges much faster.

TABLE II
SUMMARIZE OF THE EXPERIMENT ENVIRONMENT

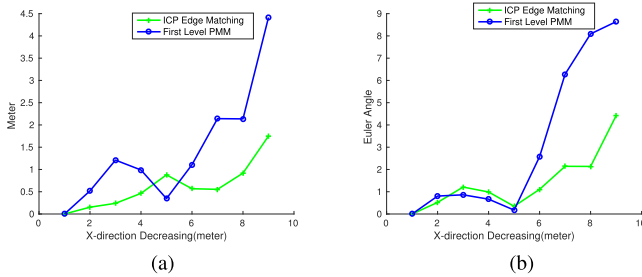| Dataset	Dimension(x*y*z)	Map Size	Edge Size	Environment Type
Corridor Dataset	20m*5m*3m	46000 voxels	**3500** voxels	Structural, static
WKW Dataset	40m*15m*3m	67071 voxels	**4300** voxels	Semi-stru, dynamic
Willow Dataset	50m*40m*3m	76000 voxels	**3200** voxels	Structural, static

Fig. 11. Sensitivity to the change of overlap in WKW dataset, where ICP edge matching and first level PMM are compared. (a) Translation Error. (b) Rotation Error.

PMM is compared against ICP [15], point-to-plane (P2P) ICP [20], Generalized ICP (GICP) [17] and Normal Distribution Transform (NDT) [22] based dense registration algorithm. The implementation of the compared algorithms are from the standard Point Cloud Library [51].

B. Edge Matching Analysis

One of the novelties of the proposed first level PMM comes at the extraction of structural information and modeling into a probabilistic registration problem. An alternative way to register extracted edges is to directly apply point-to-point ICP algorithm, which ignores the local uncertainty of each edge voxel. Here, the first level of PMM is compared against the ICP edge matching on two datasets. The initial perturbations were randomly generated for the two experiments, and the registration results of convergence rate and matching error are exhibited.

1) Registration Accuracy: The experiment was conducted in two datasets by extracting the edges from the full 3D map. Fig. 10a shows the ICP edge matching trapped into local minimum after several iterations. The reason is due to the wrong data association established, as shown in the black line in the figure. Therefore, the algorithm fails to converge to the right global minimum. Compared with ICP edge matching, the first level of PMM makes full use of the constraint on x and y directions and combines the local uncertainty information. Hence, the correct correspondences are established and gradually converged to the desired position.

In the WKW dataset, both algorithms converge to the correct minimum (see Fig. 10b). However, the algorithm proposed requires fewer iterations compared with ICP edge matching. In the experiment, our algorithm converges to the local minimum after around 15 iterations, while the classical ICP requires more than 25 iterations. In summary, our
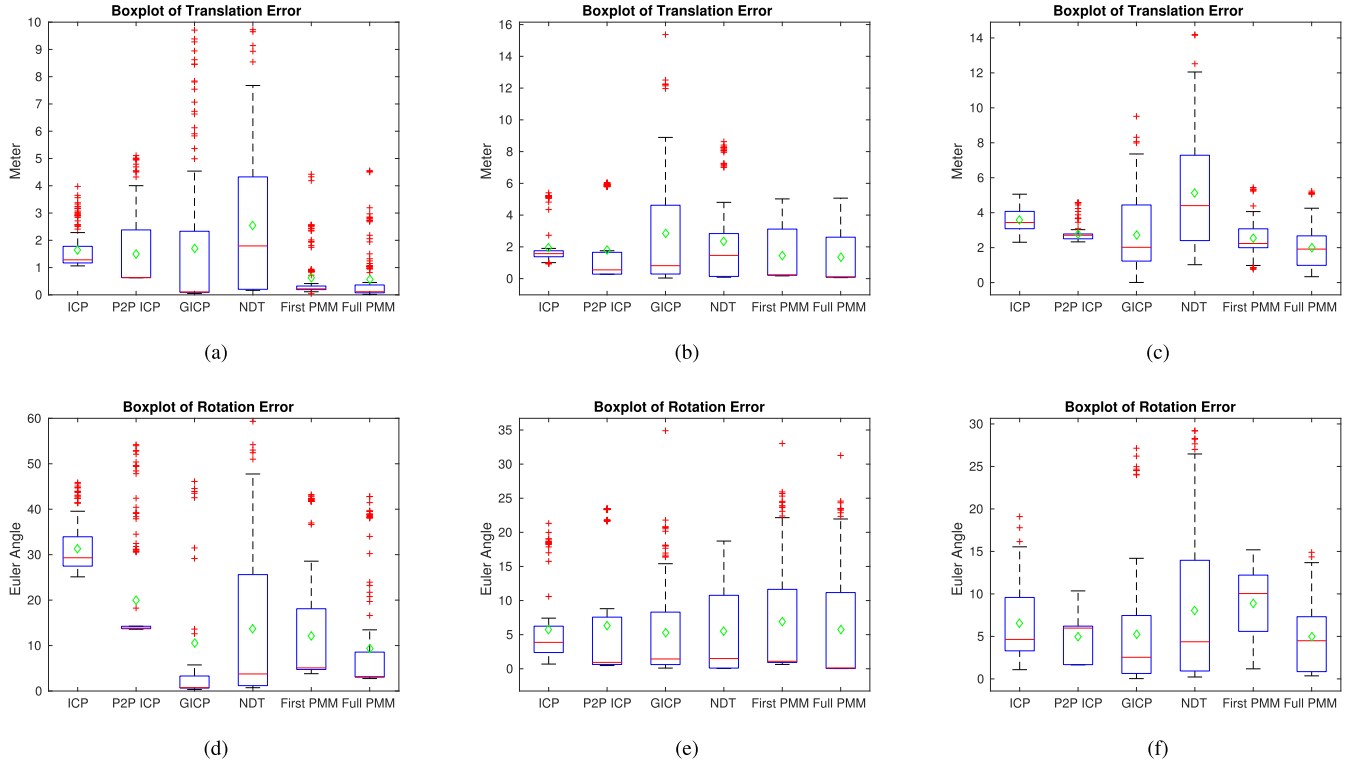
Fig. 12. Summary of convergence of the evaluated methods under 125 initial perturbations. Translation error (top) and rotation error (bottom). (a) Corridor dataset translation error. (b) WKW dataset translation error. (c) Willow dataset translation error. (d) Corridor dataset rotation error. (e) WKW dataset rotation error. (f) Willow dataset rotation error.

algorithm increases the efficiency of convergence speed by nearly 40%. The reason is that the proposed optimization algorithm penalizes more on the offset between the edge direction and increases the convergence speed.

2) Sensitivity to Overlapping: Since the maps are view-points independent and the assumption of large overlapping in dense registration does not hold true here. The influence on different overlapping parameters is also tested. The original overlapping rate between the two maps in Fig. 10b is 45%. Then, fixing the scene map to remove part of the map data by decreasing $1m$ gradually on x-direction of the scene map for nine iterations ($x = -9m$). The removing of the map data leads to the decrease of the overlapping rate to 10% when $x = -9m$. In Fig. 11, the change of rotation and translation error with the dropping of the overlapping rate is summarized, where our algorithm shows a larger convergence basin. In contrast, the classical ICP based approach has a large sliding error in the x-direction. The result shows proposed algorithm can make the most use of the map information and increase the region of attraction of the correct minimization direction.

C. Full Map Matching Analysis

For the three experiments above, an exhaustive compar-ison between ICP map matching, P2P ICP map matching, GICP map matching, NDT map matching, first level PMM and full PMM has been carried out to test the results of registration accuracy and sensitivity to different initial inputs.

The results of registration on three different scenarios are compared against the ground truth relative transformation. The scene map was transformed away from the ground truth and used as input to test the convergence. For each testing pairs, the offsets between $-2m$ and $2m$ in steps of 1m (along x and y-axes) were introduced. In addition, the maps were rotated from $-15°$ to $15°$ (in steps of $7.5°$). Thus, a total of 125 test transformations were generated and applied to the maps. For all algorithms, the threshold for nearest neighbor search is $2m$, and the number of iterations is set to be 30.

1) Sensitive to Initial Perturbations: The boxplot presented in Fig. 12 supports that the full PMM algorithm outperforms to those of ICP, P2P ICP, GICP, and NDT. The boxplot encloses the middle 50% of the data, with the median value marked by a line and the mean value marked by a green rhombus. The red cross marks indicate outliers, i.e., data values have greater distance from the standard deviation above and below the mean of the data.

For the first scenario, the translation and rotation errors of the algorithms except full PMM present a significant variance, indicating that the optimization process often falls into the local minimum. The reason is that the corridor environment has low translational features in the y-direction, where other algorithms fail to utilize the y-direction constraint in the optimization process. In the second environment, all algorithms present acceptable performance. Thanks to the abundant translational features on x-direction and y-direction, all the algorithms can relatively converge to the correct transformation. In the third environment, large translation

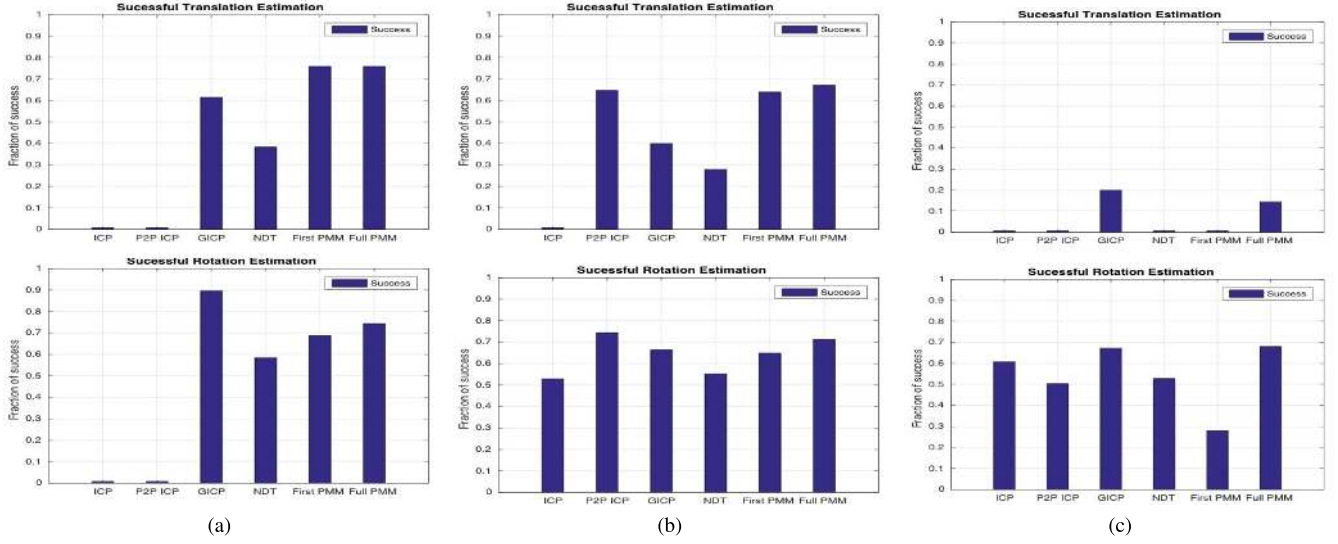| Dataset	ICP		P2P ICP		GICP		NDT		First PMM		Full PMM	
	Euler	Meter	Euler	Meter	Euler	Meter	Euler	Meter	Euler	Meter	Euler	Meter
Corridor	31.3896	1.6349	19.9685	1.5073	10.5811	1.7058	13.6265	2.5550	12.0577	0.6182	**9.3664**	**0.5719**
WKW	5.7712	1.9461	6.3146	1.7797	5.3157	2.8408	**5.5135**	2.3245	6.8964	1.4672	5.6950	**1.3436**
Willow	6.5412	3.5984	**4.9689**	2.7840	5.2842	2.7154	8.0725	5.1337	8.8821	2.5391	4.9850	**1.9978**

Fig. 13. Summary of success rates of the evaluated matching methods over all datasets. Translation estimation (top) and rotation estimation (bottom). (a) The success rate in Corridor dataset. (b) The success rate in WKW dataset. (c) The success rate in Willow dataset.

errors are found for all the algorithms. This is due to locally similar geometries, where the small office rooms are very similar. The rotation errors are much lower, suggesting that in the structured environment the translation errors are the primary concern.

2) Convergence Analysis: Tab.III summarizes the mean error in both rotation/translation estimations, the full PMM outperforms other algorithms in most of the cases. The exception is the rotation estimation of P2P ICP slightly exceeds full PMM in Willow dataset, and NDT slightly exceeds full PMM in WKW dataset. Besides the performance of full PMM, the GICP achieves the second best results. Since not all the outputs are successful, the criteria for a correct registration was chosen as a translation error of $0.6m$ or less and a rotation error of $6°$ or less.

The performance summarized in Fig. 13 exhibits the successful fraction of all three datasets. The trend is clear that the smallest errors in Tab.III exhibit the highest percentage of success. Fig. 13a shows poor performance on ICP and P2P ICP, and the most likely cause is the wrong data association due to the low constraint on y-direction. Fig. 13b shows that P2P ICP, GICP, and full PMM achieve comparable convergence basin. In the Willow dataset (see Fig. 13c), all the algorithms witness a large offset on translation estimation, which is caused by the highly symmetric structure of the small office rooms. Overall, Full PMM demonstrates superior or comparable performance over existing algorithms.

3) Efficiency Analysis: The overall computation time of full PMM is compared against the ICP, P2P ICP, GICP and NDT

TABLE IV

THE EFFICIENCY COMPARISON OF THE ALGORITHMS (SECONDS)

Datasets	ICP	P2P ICP	GICP	NDT	Full PMM
Corridor	3.14	2.90	24.08	136.42	**2.13**
WKW	10.35	8.13	46.66	233.33	**5.31**
Willow	5.59	4.47	40.27	217.38	**3.83**

methods and results are shown in Tab. IV. The experiments were executed in a desktop PC with an Intel Core i7-6700HQ CPU @2.60GHz, 24 GB RAM, running Ubuntu 14.04 in a standard C++ environment. The results indicate that ICP, P2P ICP, and full PMM require a significantly lower computation time compared to NDT and GICP. The high computation time of NDT is due to computing the required normal distributions from the octomap input in addition to the fusion step. GICP adopting a more similar approach to PMM, generates the second best map registration accuracy of all methods, however it's higher computation time can be attributed to the use of only the voxel level registration requiring more computation efforts compared to our proposed two level PMM. Therefore, it can be claimed that the high efficiency of the full PMM compared to all other methods stems from the use of the two-level structure, where edges are registered in the first level with fast convergence with high accuracy which causes the voxel based registration at the second level to converge quickly as well, making the entire process more efficient.

D. Statistical Testing and Map Merging

In the standard library, GICP and NDT cannot provide point-wise correspondence. We only discuss ICP, P2P ICP and
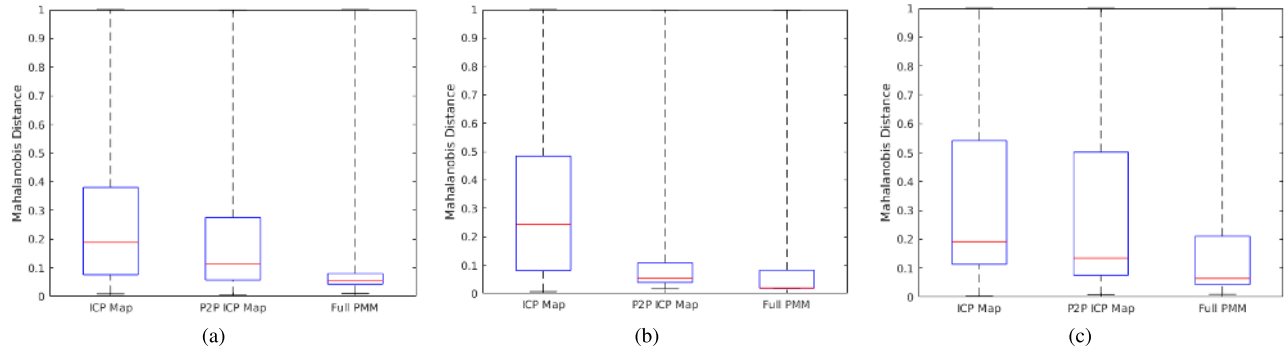
Fig. 14. The distribution of the average Mahalanobis distance between the matched pairs in all three experiments. (a) Corridor dataset Mahalanobis distance. (b) WKW dataset Mahalanobis distance. (c) Willow dataset Mahalanobis distance.
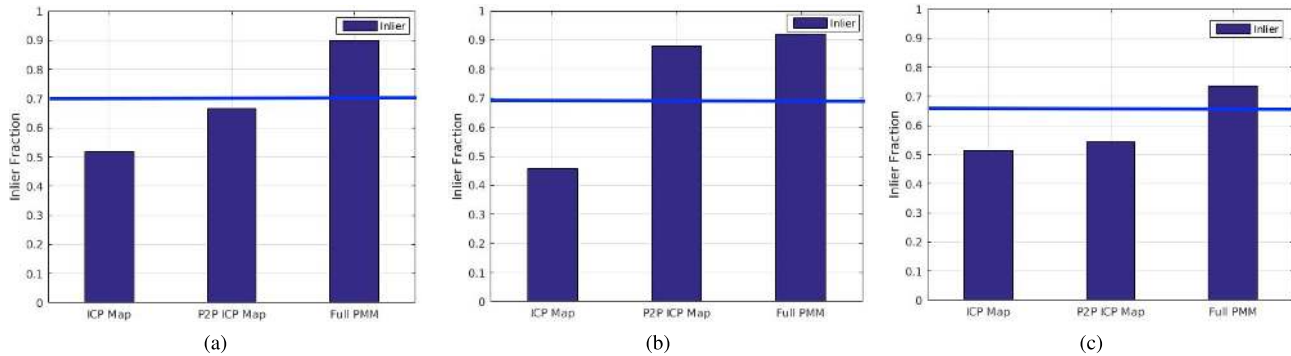

Fig. 15. The fraction of inliers and the threshold of accepting the estimated transformation. (a) Corridor dataset inlier rate. (b) WKW dataset inlier rate. (c) Willow dataset inlier rate

full PMM in the following experiments. After computing the transformation, the accuracy is evaluated by calculating the percentage of inliers and merged based on relative entropy filter.

1) Statistical Testing and Transformation Evaluation: Fig. 14 illustrates the comparison of the average Mahalanobis distances among P2P ICP map matching, ICP map matching and full PMM in the three experiment settings. The figure shows the box plots that enclose the middle 50% of the data with the median value marked by a line. It is apparent that our algorithm has lower variance and mean compared with others.

Since ICP is deterministic with only coordinate position, the covariance of each point is denoted as the identity matrix $I_{3\times3}$. For the three experiments, a strict criterion is set to guarantee the accuracy: the inliers threshold is set as 0.2 Mahalanobis distance, and the fraction of rejecting a transformation is set according to the overlapping rate. The overlapping rate for Willow dataset, WKW dataset, and Corridor dataset are 40%, 42%, and 48%, then, the acceptance is calculated as 70%, 69%, and 66% respectively.

In Fig. 15, the acceptance threshold is denoted with a black line, where the inlier rate higher than the threshold is accepted. In Corridor and Willow datasets, the inlier percentages of ICP and P2P ICP are lower than full PMM since they fail to converge to perfect alignment. Based on the auto-tuned thresholds, full PMM transformation generates accurate and reliable map registration result compared with ICP and P2P

TABLE V
AVERAGE ENTROPY OF FUSED MAP FOR DIFFERENT UPDATE RULE.
(THE BEST PERFORMANCE IS DENOTED IN BOLD)

| Dataset | Directly Stitch | | Taking Average | | Relative Entropy | |
	Size	Entropy	Size	Entropy	Size	Entropy
Corridor	40763	0.2061	**32787**	0.2141	**32787**	**0.2002**
WKW	101364	0.2374	**82765**	0.2371	**82765**	**0.2317**
Willow	95596	0.1583	**75442**	0.1680	**75442**	**0.1526**

ICP based map matching approach. Alternatively, both full PMM and P2P map matching algorithms produce acceptance results in the WKW environment.

2) Relative Entropy Filter Based Map Merging: After determining the correct transformation for map registration, a relative entropy filter is applied to achieve map merging by considering the dissimilarities between matched voxels. The relative entropy threshold is set as 0.1 in KL-divergence. The entropy of the resultant maps after applying the KL-divergence based merging algorithm is summarized in Tab. V. In comparison, the merging strategy in [7] is performed by directly taking the average of probabilities for all the matched voxels. Besides, the algorithm is also compared with directly stitching the two maps without any post-processing.

The results are summarized in Tab.(V). It is shown that the relative entropy filter based map merging algorithm reduces the size of the fused map and mean of the entropy. Both relative entropy filter and taking an average algorithm to lead to compression for the size of the map compared with directly
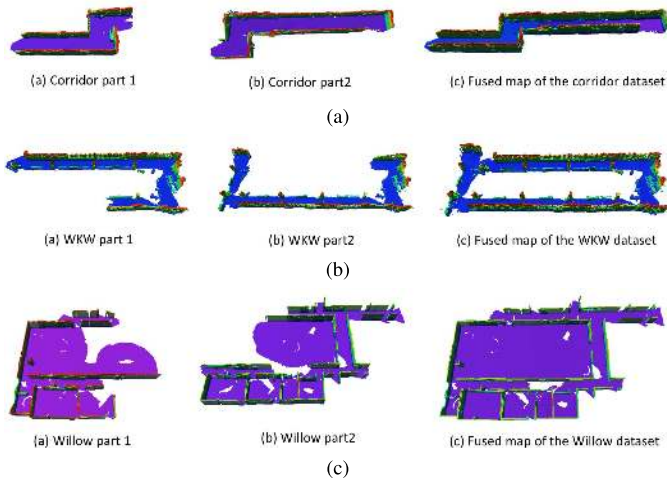
Fig. 16.　The results of fused maps for all the three experiments. (a) The fused map of Corridor dataset. (b) The fused map of WKW dataset. (c) The fused map of Willow dataset.

stitching. The entropy of WKW and Corridor environment is higher than Willow due to real-world noise and uncertainty. The experiment indicates that the proposed map fusion strategy can combine probabilities of both maps effectively to decrease the total uncertainty. Fig. 16 shows the final results of the fused 3D occupancy grid maps of three environments.

VII. CONCLUSION AND FUTURE WORK

This paper addressed the hierarchical probabilistic fusion framework for 3D occupancy maps, which was factorized into three modules: uncertainty modeling, map matching, and map merging. The paper first clarified the difference between raw sensor data registration and map matching in-depth. Then, the local uncertainty of maps was modeled by combing the individual voxel and structural edge covariances. To calculate the accurate relative transformation between maps, a two-level probabilistic map matching algorithm was proposed to combine structural information and voxel information. For the first level, 3D edges extracted from the maps were aligned based on the probabilistic data association strategy. Then, the full maps were correlated on voxel-wise to refine the result and derived the corresponding pairs for probability merging. Since the transformation was only a local minimum, statistical testing was performed to evaluate the quality of the transformation by rejecting low inlier fraction. The final map merging was achieved with a relative entropy filter which integrated the measurements from both maps and decreased the uncertainties.

Experiments on simulated and real-world environments were conducted, and the evaluations confirmed the efficacy of the proposed approach in generating more accurate relative transformation estimates between maps with a faster convergence rate and a larger convergence basin. The proposed algorithm was able to produce more consistent global 3D occupancy grid maps compared with existing methods.

Although a substantially improved performance is exhibited in the proposed algorithm, several challenges stills remain to be addressed. Mostly, being an iterative optimization algorithm, it still tends to trap into a local minimum. Thus, more improvements on finding the initial inputs as well as a strategy to reject the wrong estimation should be explored. The extension of the work to real setting multi-robot missions need to consider the following issues: a). Localization uncertainties: Under the framework of the probabilistic map matching framework, this type of localization uncertainty can be integrated into the matching process. b). Heterogeneous sensor: A multiple probabilistic map matching algorithm can be developed to deal with this issue. c). Communication issues: in the real-setting, the system will face narrow bandwidth and limited computational power under periodic communication. A distributed communication strategy to share map information among robots should be provided.

REFERENCES

[1] S. Saeedi, M. Trentini, M. Seto, and H. Li, "Multiple-robot simultaneous localization and mapping: A review," *J. Field Robot.*, vol. 33, no. 1, pp. 3–46, 2016.

[2] D. Scaramuzza *et al.*, "Vision-controlled micro flying robots: From system design to autonomous navigation and mapping in GPS-denied environments," *IEEE Robot. Autom. Mag.*, vol. 21, no. 3, pp. 26–40, Sep. 2014.

[3] N. Michael *et al.*, "Collaborative mapping of an earthquake-damaged building via ground and aerial robots," *J. Field Robot.*, vol. 29, no. 5, pp. 832–841, 2012.

[4] A. T. Rashid, M. Frasca, A. A. Ali, A. Rizzo, and L. Fortuna, "Multi-robot localization and orientation estimation using robotic cluster matching algorithm," *Robot. Auton. Syst.*, vol. 63, pp. 108–121, Jan. 2015.

[5] P. Trautman, J. Ma, R. M. Murray, and A. Krause, "Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation," *Int. J. Robot. Res.*, vol. 34, no. 3, pp. 335–356, 2015.

[6] T. M. Bonanni, B. D. Corte, and G. Grisetti, "3-D map merging on pose graphs," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 1031–1038, Apr. 2017.

[7] J. Jessup, S. N. Givigi, and A. Beaulieu, "Robust and efficient multirobot 3-D mapping merging with octree-based occupancy grids," *IEEE Syst. J.*, vol. 11, no. 3, pp. 1723–1732, Sep. 2017.

[8] S. Carpin, "Fast and accurate map merging for multi-robot systems," *Auton. Robots*, vol. 25, no. 3, pp. 305–316, 2008.

[9] S. Saeedi, L. Paull, M. Trentini, M. Seto, and H. Li, "Map merging for multiple robots using Hough peak matching," *Robot. Auto. Syst.*, vol. 62, no. 10, pp. 1408–1424, 2014.

[10] S. Saeedi, L. Paull, M. Trentini, M. Seto, and H. Li, "Group mapping: A topological approach to map merging for multiple robots," *IEEE Robot. Autom. Mag.*, vol. 21, no. 2, pp. 60–72, Jun. 2014.

[11] J. Saarinen, H. Andreasson, T. Stoyanov, and A. J. Lilienthal, "Normal distributions transform Monte–Carlo localization (NDT-MCL)," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Nov. 2013, pp. 382–389.

[12] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Auto. Robots*, vol. 34, no. 3, pp. 189–206, 2013.

[13] J. Jessup, S. N. Givigi, and A. Beaulieu, "Merging of octree based 3D occupancy grid maps," in *Proc. 8th Annu. IEEE Int. Syst. Conf.*, Mar./Apr. 2014, pp. 371–377.

[14] Y. Yue, D. Wang, P. G. C. N. Senarathne, and D. Moratuwage, "A hybrid probabilistic and point set registration approach for fusion of 3D occupancy grid maps," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2016, pp. 001975–001980.

[15] P. J. Besl and D. N. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[16] S. Ying, J. Peng, S. Du, and H. Qiao, "A scale stretch method based on ICP for 3D data registration," *IEEE Trans. Autom. Sci. Eng.*, vol. 6, no. 3, pp. 559–565, Jul. 2009.

[17] A. V. Segal, D. Haehnel, and S. Thrun, "Generalized-ICP," *Robot. Sci., Syst.*, vol. 2, no. 4, p. 435, Jun. 2009.

[18] Y. Taguchi, Y.-D. Jian, S. Ramalingam, and C. Feng, "Point-plane SLAM for hand-held 3D sensors," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 5182–5189.

[19] A. Das, J. Servos, and S. L. Waslander, "3D scan registration using the normal distributions transform with ground segmentation and point cloud clustering," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 2207–2212.

[20] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image Vis. Comput.*, vol. 10, no. 3, pp. 145–155, 1992.

[21] A. Censi, "An accurate closed-form estimate of ICP's covariance," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2007, pp. 3167–3172.

[22] M. Magnusson, "The three-dimensional normal-distributions transform—An efficient representation for registration, surface analysis, and loop detection," Ph.D. dissertation, School Sci. Technol., Örebro Univ., Örebro, Sweden, 2009.

[23] G. Agamennoni, S. Fontana, R. Y. Siegwart, and D. G. Sorrenti, "Point clouds registration with probabilistic data association," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4092–4098.

[24] M. Velas, M. Spanel, and A. Herout, "Collar line segments for fast odometry estimation from velodyne point clouds," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2016, pp. 4486–4495.

[25] Y. He, B. Liang, J. Yang, S. Li, and J. He, "An iterative closest points algorithm for registration of 3D laser scanner point clouds with geometric features," *Sensors*, vol. 17, no. 8, p. 1862, 2017.

[26] J. Servos and S. L. Waslander, "Multi-channel generalized-ICP: A robust framework for multi-channel scan registration," *Robot. Auton. Syst.*, vol. 87, pp. 247–257, Jan. 2017.

[27] J. Liu, S. Du, C. Zhang, J. Zhu, K. Li, and J. Xue, "Probability iterative closest point algorithm for position estimation," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst.*, Oct. 2014, pp. 458–463.

[28] S. Nobili, R. Scona, M. Caravagna, and M. Fallon, "Overlap-based ICP tuning for robust localization of a humanoid robot," in *Proc. IEEE Int. Conf. Robot. Autom.*, May/Jun. 2017, pp. 4721–4728.

[29] Q. Li, R. Xiong, and T. Vidal-Calleja, "A GMM based uncertainty model for point clouds registration," *Robot. Auto. Syst.*, vol. 91, pp. 349–362, May 2017.

[30] D. Viejo and M. Cazorla, "A robust and fast method for 6dof motion estimation from generalized 3D data," *Auto. Robots*, vol. 36, no. 4, pp. 295–308, Apr. 2014.

[31] J. Serafin and G. Grisetti, "Using extended measurements and scene merging for efficient and robust point cloud registration," *Robot. Auto. Syst.*, vol. 92, pp. 91–106, Jun. 2017.

[32] S. Carpin, "Fast and accurate map merging for multi-robot systems," *Auto. Robots*, vol. 25, no. 3, pp. 305–316, Oct. 2008.

[33] J. Oberländer, A. Roennau, and R. Dillmann, "Hierarchical SLAM using spectral submap matching with opportunities for long-term operation," in *Proc. 16th Int. Conf. Adv. Robot. (ICAR)*, Nov. 2013, pp. 1–7.

[34] S. Schwertfeger and A. Birk, "Map evaluation using matched topology graphs," *Auto. Robots*, vol. 40, no. 5, pp. 761–787, Jun. 2016, doi: 10.1007/s10514-015-9493-5.

[35] M. Bosse and R. Zlot, "Map matching and data association for large-scale two-dimensional laser scan-based slam," *Int. J. Robot. Res.*, vol. 27, no. 6, pp. 667–691, 2008.

[36] A. Cunningham, K. M. Wurm, W. Burgard, and F. Dellaert, "Fully distributed scalable smoothing and mapping with robust multi-robot data association," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2012, pp. 1093–1100.

[37] R. Aragues, C. Sagues, and Y. Mezouar, "Feature-based map merging with dynamic consensus on information increments," *Auto. Robots*, vol. 38, no. 3, pp. 243–259, Mar. 2015.

[38] A. Gawel, T. Cieslewski, R. Dubé, M. Bosse, R. Siegwart, and J. Nieto, "Structure-based vision-laser matching," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 182–188.

[39] P. C. Niedfeldt, A. Speranzon, and A. Surana, "Distributed map fusion with sporadic updates for large domains," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 2806–2813.

[40] M. J. Schuster, C. Brand, H. Hirschmüller, M. Suppa, and M. Beetz, "Multi-robot 6D graph SLAM connecting decoupled local reference filters," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep./Oct. 2015, pp. 5093–5100.

[41] C. Brand, M. J. Schuster, H. Hirschmüller, and M. Suppa, "Submap matching for stereo-vision based indoor/outdoor SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep./Oct. 2015, pp. 5670–5677.

[42] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3D feature matching," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 809–812.

[43] F. Endres, J. Hess, J. Sturm, D. Cremers, and W. Burgard, "3-D mapping with an RGB-D camera," *IEEE Trans. Robot.*, vol. 30, no. 1, pp. 177–187, Feb. 2014.

[44] Y. Yue, D. Wang, P. G. C. N. Senarathne, and C. Yang, "Robust submap-based probabilistic inconsistency detection for multi-robot mapping," in *Proc. Eur. Conf. Mobile Robots (ECMR)*, Sep. 2017, pp. 1–6.

[45] H. Carrillo, P. Dames, V. Kumar, and J. A. Castellanos, "Autonomous robotic exploration using occupancy grid maps and graph slam based on Shannon and Rényi entropy," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 487–494.

[46] S. Thrun, W. Burgard, D. Fox, and R. C. Arkin, *Probabilistic Robotics* (Intelligent Robotics and Autonomous Agents). Cambridge, MA, USA: MIT Press, 2005.

[47] K. Klasing, D. Althoff, D. Wollherr, and M. Buss, "Comparison of surface normal estimation methods for range sensing applications," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 3206–3211.

[48] G. C. Sharp, S. W. Lee, and D. K. Wehe, "ICP registration using invariant features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 1, pp. 90–102, Jan. 2002.

[49] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, "Distributed multirobot exploration and mapping," *Proc. IEEE*, vol. 94, no. 7, pp. 1325–1339, Jul. 2006.

[50] (2016). *ROS Package slam_gmapping*. [Online]. Available: www.ros.org/wiki/slam_gmapping

[51] R. B. Rusu and S. Cousins, "3D is here: Point cloud library (PCL)," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Shanghai, China, May 2011, pp. 1–4.

Yufeng Yue received the B.E. degree in automation from the Beijing Institute of Technology, Beijing, China, in 2014. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include collaborative mapping, multi-robot coordination, multi-robot information fusion, and reasoning.

P. G. C. Namal Senarathne received the B.Sc. degree in computer science and engineering from the University of Moratuwa, Sri Lanka, in 2009, and the Ph.D. degree from Nanyang Technological University in 2015. During his Ph.D. degree, he worked on improving 2-D exploration algorithms that use the frontier heuristic. He is currently researching the development of efficient algorithms for UGV and UAV exploration in 3-D environments, multi-robot collaborative exploration, 3-D perception for navigation of mobile robots, and 3-D path planning for UAVs.

Chule Yang received the B.Eng. degree in electrical engineering from Wuhan University, Wuhan, China, in 2014, and the M.Sc. degree in computer control and automation from Nanyang Technological University, Singapore, in 2015, where he is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering. His research interests include multimodal perception, probabilistic reasoning, and intelligent decision making for autonomous system.

Jun Zhang received the B.Eng. degree in mechanical engineering from the Huazhong University of Science and Technology (HUST) in 2012, and the M.Eng. degree in mechatronics from HUST, Wuhan, China, in 2015. He is currently pursuing the Ph.D. degree with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Currently, his research focuses on place recognition and localization, and multi-sensor calibration.

Mingxing Wen received the B.Eng. degree in automation from the Beijing Institute of Technology, Beijing, China, in 2015, and the M.Sc. degree in computer control and automation from Nanyang Technological University (NTU), Singapore, in 2016, where he is currently pursuing the Ph.D. with the School of Electrical and Electronic Engineering. He was a Research Associate with NTU. His research focuses on robot learning, and reinforcement learning applications in autonomous system.

Danwei Wang received the B.E. degree from the South China University of Technology, China, in 1982, and the M.S.E. and Ph.D. degrees from the University of Michigan, Ann Arbor, MI, USA, in 1989 and 1984, respectively. Since 1989, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Currently, he is a Professor and the Co-Director of the ST Engineering-NTU Corporate Laboratory. He is leading the Autonomous Mobile Robotics Research Group. His research interests include robotics, control theory, and applications. He is a Senator in the NTU Academics Council. He was a recipient of the Alexander von Humboldt Fellowship, Germany. He has served as the general chairman, technical chairman, and various positions in international conferences, such as ICARCV and IROS Conferences. He is an Associate Editor of the *International Journal of Humanoid Robotics* and served as an Associate Editor for the Conference Editorial Board, IEEE Control Systems Society, from 1998 to 2005.