The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Coarse-to-Fine Global Localization for Mobile Robots with Hybrid Maps of Objects and Spatial Layouts

Soonyong Park, Howon Cheong, and Sung-Kee Park

Abstract— This paper proposes a novel global localization approach that uses hybrid maps of objects and spatial layouts. We model indoor environments using the following visual cues from a stereo camera: local invariant features for object recognition and their 3D positions for object location representation. We also use a 2D laser range finder. Therefore, we can build a hybrid local node for a topological map that is composed of an object location map and a spatial layout map. Based on this modeling, we suggest a coarse-to-fine strategy for the global localization. The coarse pose is obtained by means of object recognition and point cloud fitting, and then its fine pose is estimated with a probabilistic scan matching algorithm. With real experiments, we show that our proposed method can be an effective global localization algorithm.

I. Introduction

One of the most important functions of a robotic map is to provide the robot with clues to estimate its pose. The properties of the map vary according to the sensing modality. Consequently, depending on the type of map, the localization methods should be different. Over the past years, 2D range sensors, such as laser range finder, sonar, and IR sensors, have been used extensively for robotic mapping. These maps are simple representations containing only geometrical information, such as evenly-spaced grids, corners, and edge features. When a robot estimates its pose, however, it can be difficult to detect correspondences in geometrically non-distinctive environments. This problem is due to the fact that most range sensor based localization methods have employed simple feature matching or map matching approaches [1, 2, 3].

In contrast to the range sensors, a vision sensor provides richer and more intuitive sensing information, in that it can give both geometrical data, such as stereo depth, and other visual cues, such as object recognition and place recognition. In addition, it has been shown that vision-based mapping and localization is more stable in dynamic environments than other approaches using range sensors [4]. This is because the local invariant features in captured images are full of information, and so are more discriminative than scanned range data. Therefore, a vision sensor can be a useful sensing modality to deal with both map representation and localization. Most of the recent studies on vision-based mapping and localization have been based on either a place recognition approach or object recognition approach. Most place recognition-based systems [5, 6, 7, 8] represent the environment as representative views captured from the environment, or as a set of specific types of visual features, such as local invariant features [9], extracted from the environmental images. Then, the localization problem can be solved with a place recognition method: matching the visual features detected in the query image to those contained on a pre-built visual feature database, or finding the reference image in a representative image database that is the most similar in appearance to the query image. On the other hand, object recognition-based systems [10, 11, 12] represent a specific place by means of the object entities that exist in that place. The localization is then performed by verifying that the recognized object is contained in the place.

In this paper, we propose a complete global localization system: hybrid maps of objects and spatial layouts and coarse-to-fine global localization method. The map consists of global topological map and local hybrid map, in which each local space is connected as a topological graph and its detailed information is represented with objects and spatial layout. The objects found in each local space give the spaces a unique identification, and their positions are described with respect to each local space's reference frame. The spatial layout describes an empty area and a geometric shape of local spaces, and provides metric information to compute the spatial relationships between neighboring spaces. Contrast to the conventional maps, such integration of object and spatial layout into a topological map can give the proposed map properties of human cognitive map [13] and semantic map [14]. Moreover, the computational complexity of metric maps can be reduced by separating the environment representation into local maps that are connected by a topological graph. The localization process consists of three stages: perception, coarse pose estimation, and fine pose estimation. The perception is performed with object recognition. It allows a robot to infer the candidate nodes where the robot is expected to be located in. The coarse pose estimation is to compute the robot pose relative to each candidate node. It is conducted with point cloud fitting. The fine pose estimation is carried out with a probabilistic scan matching which enables both the determination of correct node among the candidate nodes and the computation of robot pose relative to the correct node.

In the present paper, we use a mobile robot equipped with a multi-sensor system composed of two 180° laser range finders to extract a spatial layout and a stereo camera to perform object recognition. In addition, we apply two kinds of visual

This research was performed for the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Knowledge Economy of Korea.

Soonyong Park, Howon Cheong, and Sung-Kee Park are with the Center for Cognitive Robotics Research at the Korea Institute of Science and Technology (KIST), 39-1 Hawolgok-dong, Sungbuk-gu, Seoul, Korea.
{scipio77,dest2,skee}@kist.re.kr

features for the object recognition and the pose estimation: a PCA feature [15] and stereo depth. The PCA feature represents a normalized SIFT descriptor by applying the principle components analysis, and this representation significantly improves SIFT's matching performance.

The rest of this paper is organized as follows. Section 2 presents a method to represent the proposed map. Our global localization process and detailed algorithms will be presented in section 3. Experimental verifications are presented in section 4 and some concluding remarks are given in section 5.

II. Map Representation

The environment is considered to be represented as a set of local spaces and the environmental map is constructed by human augmented mapping [16]; a user brings the robot to some memorable places and commands it to build a map around there. The proposed map is composed of a global topological map and local hybrid maps. The global topological map consists of nodes and arcs as shown in Fig. 5 (b). It represents the spatial relationship between the local spaces in terms of an adjacent matrix. The local spaces form nodes of the topological map. Our previous work [17] presents the details of the global topological map and map building procedure. In addition, the navigation methodology using the proposed map is also presented in [17]. The local hybrid map describes the detailed information of a local space in terms of objects found in the space and the spatial layout of the space. It is composed of a spatial layout map and an object location map.

The spatial layout map represents a geometric shape and an empty area of each node, and it is described as a laser range scan (Fig. 1 (c)). We assume that the geometrical characteristics of a local space can be described from a 360° laser range scan obtained at one position. A laser range finder can gather high quality range scan data and it suffers from very small number of specular reflections. The angular uncertainty of the laser sensor is very small and, therefore, it can provide a very fine description of the surroundings to the robot. The range scan data also provides metric information to compute the relative position and orientation displacements between neighboring nodes. Various scan matching methods can be applied for the purpose of calculating the displacements [2, 3].

The object location map is composed of specific objects that characterize each node, and it is generated from omni-directional environmental images and depth information from the stereo camera. The object location map provides two kinds of information about objects: appearance and location. Appearance information describes the appearance properties of objects seen at a node point in terms of PCA feature models for object recognition. Here the origin point of each node's reference frame is denoted as the node point. The location information describes the locations of objects with respect to the nodes' reference frames, and it is represented as a point cloud corresponding to the 3D coordinates of each object's PCA features (Fig. 1 (b)).

In this paper, the PCA feature models for the objects in each node are built manually. In other words, the training images used to build the PCA feature models are selected manually from among the omni-directional environmental images, and we consider the training images that contain objects that are potentially interesting to humans and recognizable by humans. Among these potentially interesting objects included in the training images, those objects that have more PCA features than a specific threshold are defined as the visual landmarks in the object location map. In this paper, the threshold number was set to 10.

In the context of localization, the object location map can provide efficient means to avoid the problem with symmetries in geometrically non-distinctive environments. Since most local spaces in indoor environments have unique object sets which characterize the spaces, and if these objects can be

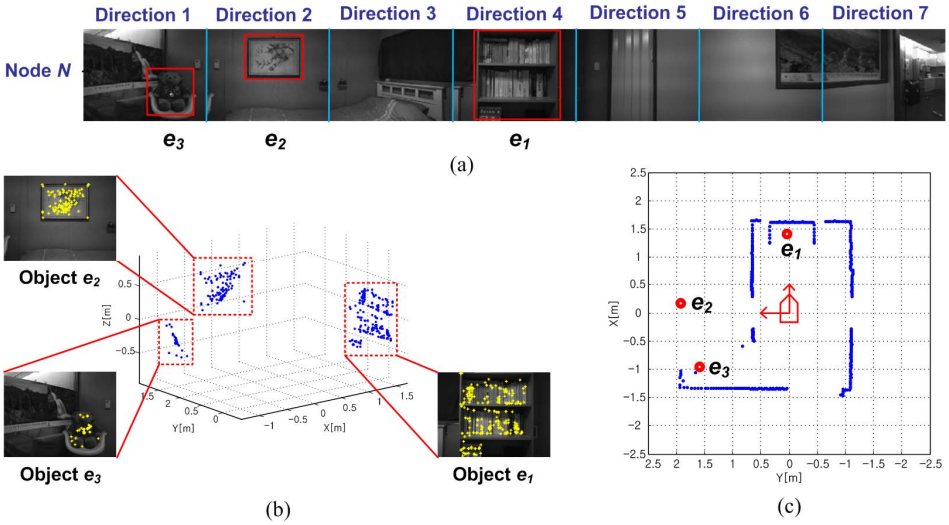


Fig. 1. Example of a local hybrid map. (a) Omni-directional environmental images of node *N*. Direction 4 coincides with the front of the robot, i.e., the direction of the x axis. (b) Example of an object location map. (c) Example of a spatial layout map with one point locations of each object.

recognized, the robot is able to determine its location.

Fig. 1 shows an example of a local hybrid map. Fig. 1 (a) shows omni-directional environmental images taken from node *N*. Images in which useful objects are seen are adopted for object model views. Node *N* includes objects e_1, e_2, and e_3. From the model views of directions 1, 2, and 4, each object's PCA features are extracted and registered as object feature models. These feature models are used for object recognition in the localization process. In addition, stereo depth information, which is associated with each object's PCA features, is extracted and transformed into 3D coordinates relative to the node's reference frame. Thus, the transformed 3D coordinates form a point cloud. Fig. 1 (b) shows the object location map built from Fig. 1 (a). The spatial layout map of node *N* is shown in Fig. 1 (c). We can also identify the one point locations of the objects, which are mean values of each object's 3D point cloud.

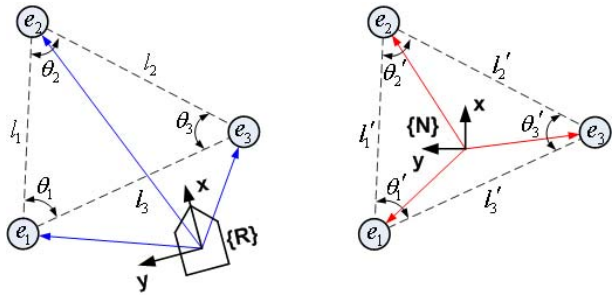
III. Global Localization

Global localization is to estimate a robot's pose (position and orientation) in a previously learned map when the robot's initial pose is unknown. An accurate estimate of pose improves the initialization process and reduces the chance of having a "kidnapped robot." The proposed global localization process is performed in three stages: perception, coarse pose estimation, and fine pose estimation. In the first stage, the robot perceives objects from the acquired images of its environment and determines the candidate nodes where it is expected to be located. Once the candidate nodes are identified, the coarse poses with respect to these nodes are computed by using 3D point cloud fitting. In the final stage, the correct node is determined and the fine pose relative to this correct node is estimated by means of a Monte Carlo method based on probabilistic scan matching.

A. Perception

Object recognition is performed by matching the PCA features extracted from the scene images (i.e. the omni-directional images for localization) to each of the object feature models stored in all of the object location maps. By doing so, each node can obtain the set of PCA features that are matched with the PCA features of the objects recognized by the robot. These sets of matched PCA features are then used to compute a coarse pose relative to each candidate node.

In this paper, we employ the spectral matching method for the object recognition [18]. This graph-based matching approach presents an efficient spectral method for finding consistent correspondences between two sets of features. This spectral matching method differs from existing approaches based on similar graph-based matching methods [19], in that it has a much better computational complexity, which allows it to scale much better to large data sets, while being robust to noise and outliers. Compared to approaches based on voting and the Hough transform, such as [9], it uses all of the available data at once as opposed to generating transformation hypothesis from subsets correspondences. Also, it does not make an explicit, strong assumption of affine mapping between the scene and model.



(a) (b)

Fig. 2. Distance and angular relationships between adjacent objects. (a) Objects recognized by the robot. (b) Identical objects included in a node.

Candidate nodes are determined based on the object recognition results. A candidate node contains one or more objects identical to the object recognized by the robot. In addition, the set of identical objects included in each candidate node should have the same configuration as the set of recognized objects. For each candidate node, we can define a vicinity probability that the robot would be located in that node. This probability is applied to remove the useless candidate nodes. If the probability of a candidate node is under a threshold value, the node is dropped out from the candidate nodes. Experimentally, the threshold value was set to 0.5. In the following, we will explain how to compute the vicinity probability by using the example shown in Fig. 2.

As shown in Fig. 2 (a), if the objects e_1, e_2, and e_3 are recognized, the position vectors of these objects are computed by averaging the 3D coordinates of each object's PCA features. Each object has a distance relationship l_i [mm] and an angular relationship θ_i [°] with only two neighboring objects. We consider that the vicinity probability should be a function of several factors. First, it should depend on the number of common objects between the recognized objects and the identical objects included in each candidate node. Second, it should depend on the spatial relationship between the recognized objects and the identical objects: the distance and angular relationships. Therefore, the vicinity probability should be increased based on an increase in the number of common objects and a decrease in differences of the spatial relationship. The vicinity probability is then computed by

$$p(z \mid N, m) \leftarrow \frac{1}{1+\lambda \cdot z} \tag{1}$$

where $p(z \mid N, m)$ is the vicinity probability that the robot is located in candidate node *N*, and *m* denotes the object location map of node *N*. λ is a weighting factor for the number of common objects, and is defined as $\lambda = \exp(n_c)^{-0.5}$. n_c denotes the number of common objects between the recognized objects and identical objects included in node *N*. *z* is a similarity measure that compares the distance and angular relationships between the recognized objects and the identical objects. The

similarity measure is therefore defined as follows:

$$z = \alpha \sum_{i=1}^{n_c} | l_i - l_i' | + \beta \sum_{i=1}^{n_c} | \theta_i - \theta_i' | \qquad (2)$$

where α and β are weighting factors due to scale differences in the distance (millimeters) and angle (degrees). l_i' and θ_i' are the distance and angular relationships of each of the identical objects in Fig. 2 (b), respectively. When $n_c = 2$, the angular difference is ignored. However, we cannot compute the vicinity probability for a candidate node that includes only one identical object, since it is impossible to define the distance and angular relationships with only a single object. Because of this, the node that includes only one identical object is retained as a candidate node.

B. Coarse Pose Estimation

The coarse poses are computed relative to all candidate nodes. This is performed by fitting the 3D point cloud of the objects being recognized to the corresponding 3D point cloud of identical objects included in each candidate node.

As a first step, the 3D coarse pose is described in terms of a homogeneous transformation matrix [20] such as ${}^{N}_{R}\mathbf{T}$. This matrix describes the robot reference frame {R} with respect to the node reference frame {N}, and consists of a rotation matrix and an origin vector. The rotation matrix denotes the orientation of the moving frame {R} with respect to the fixed frame {N}. The origin vector is a position vector that locates the origin of the frame {R}.

Fig. 3 shows an example of computing the transformation matrix. In Fig. 3 (a), ${}^{R}\mathbf{p}_i$ are 3D position vectors of the PCA features of the recognized objects, and are defined with respect to the robot's reference frame {R}. These vectors form a point cloud. ${}^{N}\mathbf{v}_i$ in Fig. 3 (b) are 3D position vectors of the PCA features of identical objects included in a candidate node. They correspond to the PCA features of objects recognized by the robot in Fig. 3 (a). ${}^{N}\mathbf{v}_i$ are defined with respect to the node reference frame {N} where $i = 1 \sim n$. n is the number of matched PCA features between recognized objects and identical objects included in the candidate node. Consequently, ${}^{N}_{R}\mathbf{T}$ fits ${}^{R}\mathbf{p}_i$ into ${}^{N}\mathbf{v}_i$ as shown in Fig. 3 (c), and this transformation is described by the expression

$${}^{N}\mathbf{v}_i = {}^{N}_{R}\mathbf{T}\,{}^{R}\mathbf{p}_i \qquad (3)$$

In this paper, we employ the parameter estimation method used in [21] to compute ${}^{N}_{R}\mathbf{T}$. This method is based on the singular value decomposition and provides a least-squares estimate of the rigid body transformation parameters. The homogeneous transform matrix is typically described in a pair of the rotation matrix ${}^{N}_{R}\mathbf{R}$ and origin vector ${}^{N}\mathbf{q}_R$ as follows:

$${}^{N}_{R}\mathbf{T} = \begin{bmatrix} & {}^{N}_{R}\mathbf{R} & & {}^{N}\mathbf{q}_R \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & q_x \\ r_{21} & r_{22} & r_{23} & q_y \\ r_{31} & r_{32} & r_{33} & q_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (4)$$

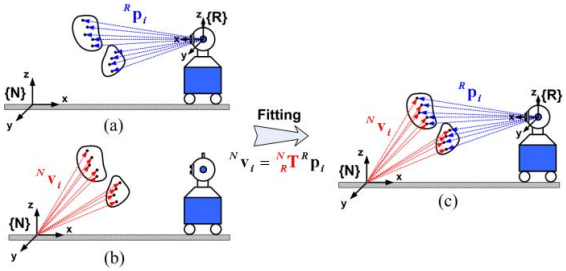


Fig. 3. (a) Point cloud of recognized objects. (b) Point cloud of objects included in candidate node. (c) Point cloud fitting between recognized objects and identical objects included in candidate node.

From the elements of rotation matrix ${}^{N}_{R}\mathbf{R}$ and origin vector ${}^{N}\mathbf{q}_R$ in (4), we can compute the 3D robot pose. The robot pose is expressed by 6 DOF (x, y, z, θ_x, θ_y, θ_z) relative to {N}, where (x, y, z) are position coordinates and the same as ${}^{N}\mathbf{q}_R$, and (θ_x, θ_y, θ_z) are rotation angles relative to the x axis, y axis, and z axis, respectively. From (4), we can compute the rotation angles as follows [20]:

$$\begin{aligned} \theta_y &= \mathrm{A}\tan 2\left(\sqrt{r_{31}^2 + r_{32}^2}\ ,\ r_{33}\right) \\ \theta_z &= \mathrm{A}\tan 2\left(r_{23}/\sin(\theta_y)\ ,\ r_{13}/\sin(\theta_y)\right) \\ \theta_x &= \mathrm{A}\tan 2\left(r_{32}/\sin(\theta_y)\ ,\ -r_{31}/\sin(\theta_y)\right) \end{aligned} \qquad (5)$$

Note the computed pose is coarse, since it is not only inaccurate due to the measurement errors of vision sensors and matching errors between PCA features; it is also not yet possible to establish the correct node.

C. Fine Pose Estimation

In this section, our methodology for estimating 2D fine pose and determining the correct node is discussed. This methodology involves using a Monte Carlo method [22, 23] based on a probabilistic scan matching in conjunction with laser range scan and coarse pose information. The Monte Carlo algorithm is widely accepted as an inference method that can cope well with decision making in the context of multimodal uncertainty.

Fig. 4 shows the graphical representation of the fine pose estimation strategy. Let a laser range scan $S_R = \{{}^{R}\mathrm{d}_1, \ldots, {}^{R}\mathrm{d}_n\}$ be a set of n points (${}^{R}\mathbf{d}_i = [\,{}^{R}x_i\,,\,{}^{R}y_i\,]$) acquired in the robot location {R}, and $S_{N_k} = \{\,{}^{N_k}\mathbf{r}_1, \ldots, {}^{N_k}\mathbf{r}_m\,\}$ another set of m points acquired in the node point of $\{\mathrm{N}_k\}$. Let $\mathrm{x} = \{x, y, \theta\}$ be the relative pose of {R} with respect to $\{\mathrm{N}_k\}$. The approximate pose $\hat{\mathbf{x}}_0$ is known from the coarse pose information. The problem is therefore to estimate the robot pose x such that maximizes the overlap between S_R and S_{N_k}. The overlap is a set of correspondences between points of S_R and S_{N_k}.

The basic approach is to perform iterative generation of random samples (i.e. the hypotheses for robot pose) according to their weights until all the samples converge from their initial positions, so that the estimated pose maximizes the overlap between the spatial layout map of the correct node and the laser range scan acquired by the robot. In this paper, we employ the

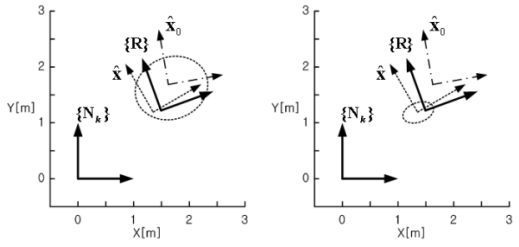


Fig. 4. Graphical representation of fine pose estimation process. $\hat{\mathbf{x}}_0$ denotes the approximate pose which is used as the seed position for initial random sampling. $\hat{\mathbf{x}}$ represents the estimated pose after sample convergence. (a) Initial boundary of random samples (dotted ellipse). (b) Boundary of random samples after convergence (dotted ellipse).

sampling importance resampling (SIR) [22, 23] approach to construct samples at every iteration.

Establishing the correspondences between two scans is crucial problem in this process. This is difficult when using a raw scan data in the presence of large errors in the approximate pose and noises on the scan data. For solving the correspondence problem more efficiently, we transform each scan into a binary grid image which partitions the 2D space into grid pixels. Each pixel has a binary occupancy value; "1" for occupied with the scan data and "0" for free. Then the correspondences can be defined as occupied pixels in the overlap between two grid images. The proposed basic procedure to estimate the fine pose is as follows:

(1) Initialization

The initial random samples are evenly generated around the coarse poses relative to each candidate node under the assumption that the true position {R} is included in the boundary of the samples (Fig. 4 (a) and Fig. 9 (first)):

$$\mathbf{s}_0^{(n)} = \hat{\mathbf{x}}_0^{(N_k)} + \rho \cdot \mathbf{B}\mathbf{w}_0^{(n)} \tag{6}$$

where $\hat{\mathbf{x}}_0^{(N_k)} = (x^{(N_k)}, y^{(N_k)}, \theta_z^{(N_k)})$ is 2D coarse pose relative to the reference frame of candidate node N_k. $\mathbf{w}^{(n)}$ is a vector of standard normal random variables, and $\mathbf{B}$ is a 3×3 diagonal matrix of which non-zero elements correspond to process noises (e_x, e_y, e_θ), and ρ is a noise scale factor for initial random sampling. n denotes the number among the random samples, $n = 1, \ldots, n_s$. For this paper, we set $e_x = e_y = 100$mm, $e_\theta = 5°$, and $\rho = 10$.

(2) Iteration

From the old sample set $\{\mathbf{s}_{t-1}^{(n)}, \pi_{t-1}^{(n)}, c_{t-1}^{(n)}, n = 1, \ldots, n_S\}$ at iteration step t-1, a new sample set $\{\mathbf{s}_t^{(n)}, \pi_t^{(n)}, c_t^{(n)}\}$ is constructed for iteration step t. Here, $\pi_t^{(n)}$ is the weight of the sample $\mathbf{s}_t^{(n)}$ at iteration step t, and $c_t^{(n)}$ is the cumulative weight. The *n-th* of n_S new samples is constructed in a four step approach.

Step – 1. Select a sample $\mathbf{s}_t'^{(n)}$ as follows:

(a) generate a random number $u \sim \mathcal{U}_{[0,1]}$

(b) find the smallest j for which $c_{t-1}^{(j)} \geq u$

(c) set $\mathbf{s}_t'^{(n)} = \mathbf{s}_{t-1}^{(j)}$

Step – 2. Propose new poses of the samples as follows:

$$\mathbf{s}_t^{(n)} = \mathbf{s}_t'^{(n)} + \mathbf{B}\mathbf{w}_t^{(n)} = \begin{bmatrix} x_t^{(n)} & y_t^{(n)} & \theta_t^{(n)} \end{bmatrix}^T \tag{7}$$

Step – 3. Compute weights of the new samples:

$$\pi_t^{(n)} = \frac{\exp\left\{-G \times \left(\omega_{\max} - \omega_t^{(n)}\right)\right\}}{\sum_n \exp\left\{-G \times \left(\omega_{\max} - \omega_t^{(n)}\right)\right\}} \tag{8}$$

where $\omega_t^{(n)} = \dfrac{N\left(m^{(N_k)}, m_t^{(n)}\right)}{\sum_n N\left(m^{(N_k)}, m_t^{(n)}\right)}$, $\omega_{\max} = \max_n(\omega_t^{(n)})$, $G = \dfrac{K}{\omega_{\max}}$

We define $N(m^{(N_k)}, m_t^{(n)})$ as the number of occupied grid pixels in the overlap between the grid images $m^{(N_k)}$ of S_{N_k} and $m_t^{(n)}$ of $S_R^{(n)}$. K is a weighting factor for sample convergence (in this case $K = 5$). For constituting $m_t^{(n)}$, we first transform the scan data S_R with respect to the location of sample $\mathbf{s}_t^{(n)}$ by assuming that the data are obtained at the sample location as follows:

$$\begin{aligned} {}^R\mathbf{d}_i^{(n)} &= \mathbf{R} \cdot {}^R\mathbf{d}_i + \mathbf{t} \\ &= \begin{bmatrix} \cos\theta_t^{(n)} & -\sin\theta_t^{(n)} \\ \sin\theta_t^{(n)} & \cos\theta_t^{(n)} \end{bmatrix} \cdot \begin{bmatrix} {}^R x_i \\ {}^R y_i \end{bmatrix} + \begin{bmatrix} x_t^{(n)} \\ y_t^{(n)} \end{bmatrix} \end{aligned} \tag{9}$$

where $\mathbf{R}$ and $\mathbf{t}$ represent the coordinate transformation matrix from the reference frame of node N_k to the assumed robot pose $\mathbf{s}_t^{(n)}$. The transformed scan data is denoted as $S_R^{(n)}$. $m_k^{(n)}$ is then obtained by transforming $S_R^{(n)}$ into the grid image. For the next iteration, $(\mathbf{s}_t^{(n)}, \pi_t^{(n)}, c_t^{(n)})$ are stored where

$$\begin{aligned} c_t^{(0)} &= 0\,, \\ c_t^{(n)} &= c_t^{(n-1)} + \pi_t^{(n)} \end{aligned} \tag{10}$$

Step-4 : Repeat Step 1 through Step 3 until all of the samples converge into a candidate node. In addition, the sample distribution range should satisfy a boundary condition such that the mean distance between each sample position and the average position of the samples is less than a threshold value (Fig. 4 (b) and Fig. 9 (third)):

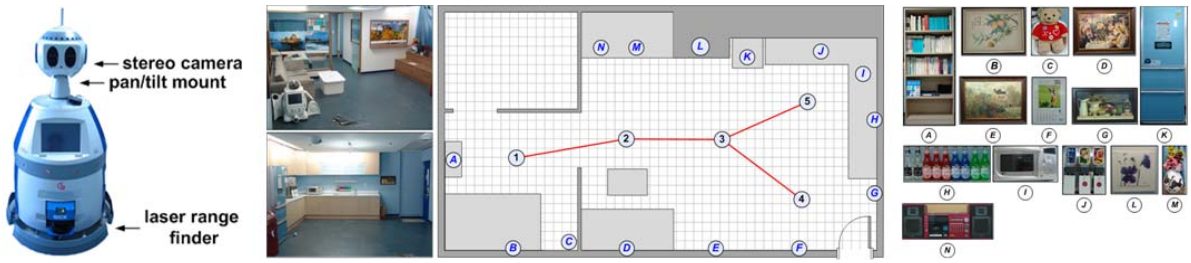
$$\frac{\sum_{n=1}^{n_s} \sqrt{(x_n - \bar{x})^2 + (y_n - \bar{y})^2}}{n_s} < D_{threshold} \tag{11}$$

where (x_n, y_n) represents the position of the *n-th* sample and $(\bar{x}, \bar{y})$ denotes the average position of the samples. The threshold value is defined as $D_{thresold} = 150$ mm.

(3) Estimation

After the iteration process, the correct candidate node, determined from the convergence of all the samples, is established. The fine pose is then estimated as follows:

$$\varepsilon[\mathbf{x}_t] = \sum_{n=1}^{n_S} \left(\pi_t^{(n)} \times \mathbf{s}_t^{(n)}\right) \tag{12}$$

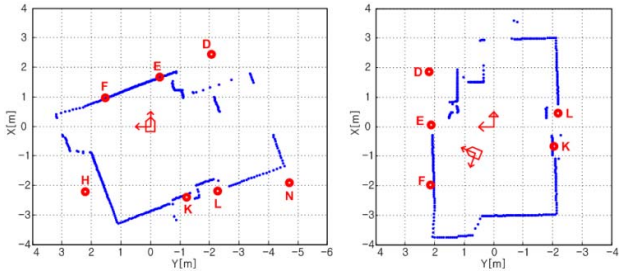


(a) (b) (c)

Fig. 5. (a) Infotainment robot, a mobile robot equipped with a stereo camera fixed on a pan/tilt mount and two laser range finders. (b) Layout of the experimental environment. The map of the environment is composed of five nodes. (c) Objects used as visual landmarks.

Fig. 6. Omni-directional environmental images of each node. Each object's feature models for object recognition are learned from these images. The objects enclosed with the red box belong to each node where the images are captured. The horizontal depth map of each node is also built from the disparity images corresponding to the above images. Direction 4 coincides with the front of the robot, i.e. direction of the x axis.

Fig. 7. Omni-directional environmental images for localization and recognized objects. Direction 4 coincides with the front of the robot, i.e. the direction of the x axis.



(a) (b)

Fig. 8. (a) Laser range scan data for localization and the positions of recognized objects. (b) Spatial layout map of node 3 and positions of objects included in node 3.

IV. Experimental Results

The global localization strategy described above were implemented in our mobile robot (Fig. 5 (a)) and tested in an indoor, home-like environment (Fig. 5 (b)). The robot was equipped with a stereo camera and two laser range finders. The experimental environment was 10m × 5m in size and contained some household objects. The map was composed of five nodes. As shown in Fig. 5 (c), there were fourteen objects used as visual landmarks, and their positions are shown in Fig. 5 (b). Fig. 6 shows the environmental images gathered at each node point. The PCA feature models for object recognition are trained by supervised learning from all images in which any objects used as visual landmarks are seen.

The localization experiments were carried out with the obtained map. To analyze error distributions, ground truth poses were measured with a robot pose estimation system that used laser range data [2, 3]. Three types of experiments were conducted. The first evaluated the proposed global localization algorithm in detail. The second evaluated the performance of the global localization algorithm in a dynamic situation. The third evaluated the accuracy of the fine pose estimation.

Table 1. Result of coarse pose estimation

Candidate nodes	Common objects	$p(z \mid N,m)$	2D coarse pose		
			X [mm]	Y [mm]	θ [°]
2	D, L, N	0.66	-2701.68	1008.14	65.13
3	D, E, F, K, L	0.82	-897.08	613.88	68.80
4	E, F, H	0.69	-1178.31	203.15	-105.65
5	H, K	0.61	-1621.92	1283.42	162.05

The first experiment consisted of performing global localization at a random location. The omni-directional environmental images for localization were captured at a random position and the images are shown in Fig. 7. Seven objects were recognized by the object recognition process: D, E, F, H, K, L, and N. The laser range scan data from the test position was also obtained. Fig. 8 (a) shows the laser range scan data for localization and the positions of the recognized objects.

Nodes 2, 3, 4, and 5 were selected as candidate nodes. Table 1 shows the common objects from those recognized by the robot and the identical ones included in each candidate node. The vicinity probability in (1) and 2D coarse poses relative to each candidate node are also reflected in the table. Among the candidate nodes, node 3 obtained the largest vicinity probability that the robot would be located in that node, and the number of common objects was the greatest.

Fig. 9. Converging samples into correct node during fine pose estimation. At the beginning (first), after 3rd iteration (second), and 8th iteration (third).

Node 3 was therefore determined to likely be the correct node.

After the coarse pose estimation, the correct node and fine pose were determined by using the probabilistic scan matching algorithm. Fig. 9 shows how the samples converged during the fine pose estimation process. Three snapshots were selected for explanation. The sample distribution was presented at each step. In the beginning, each of the 200 samples was uniformly distributed over each candidate node. The 2D coarse pose (X, Y, θ) in Table 1 was selected as the seed pose for random sampling. After the third iteration, most samples were concentrated on node 3 (second snapshot). The final snapshot shows all samples converged into node 3 when the robot uniquely determined the correct node and fine pose. Therefore, node 3 was determined as the correct node. The fine pose was (X, Y, θ) = (-852.35mm, 610.14mm, 71.15°), and the ground truth pose was (-848.65mm, 612.38mm, 70.98°). The fine pose was defined relative to the reference frame of the correct node and it is visually described in Fig. 8 (b).

The second experiment focused on the aspect of global localization performance in a dynamic situation: the robustness of the pose estimation in the presence of people and after removing some of the objects included in each node. For this experiment, we captured the omni-directional environmental images for localization at the same location as used in the first experiment. Fig. 10 shows the omni-directional environmental images, in which three people are shown in directions 3, 6, and 7. Furthermore, we can only find object E when comparing with Fig. 7. Some of the objects were occluded by the people and the others were removed from their positions.

The object recognition process recognized object E. The laser range scan data from the test position was also acquired. Fig. 11 shows the laser range scan data and the position of the recognized object. Since only one object

Fig. 10. Omni-directional environmental images for localization in a dynamic situation. The images were captured at the same location as Fig. 7.

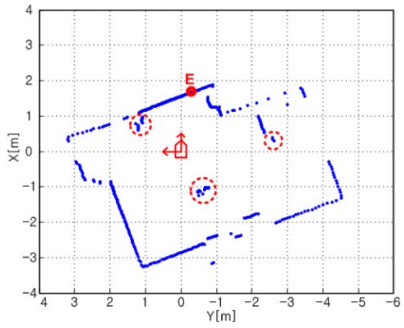


Fig. 11. Laser range scan data for localization in a dynamic situation and the position of the object being recognized. Dotted circles indicate people moving in the robot's surroundings.

Table 2. Results of pose estimation relative to node 3 in a dynamic situation.

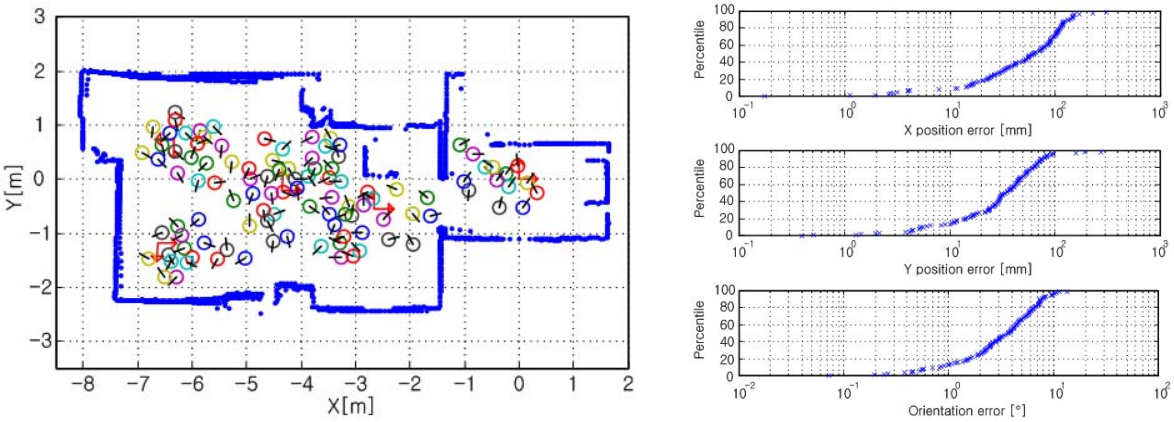
	x [mm]	y [mm]	θ_z [°]	z [mm]	θ_x [°]	θ_y [°]
Coarse pose	-861.40	621.48	70.11	15.35	1.17	-2.49
Fine pose	-855.03	618.66	70.71			
Ground truth	-848.65	612.38	70.98			
Error	-6.38	-6.28	0.27			

was recognized, we could not define the vicinity probability in (1). Nodes 3 and 4 were selected as candidate nodes, and node 3 was finally determined as the correct node. Table 2 shows the pose estimation results relative to node 3 and the error between the ground truth and fine pose.

The third experiment focused on verifying the accuracy of the fine pose estimation in the home-like environment. For this experiment, a total of 150 different test positions were randomly chosen from the whole experiment environment. Unlike conventional Monte Carlo localization [1], the robot did not need additional any motion behavior to localize. Fig. 12 (a) shows the localization results for the test positions, indicating the robot's estimated position and orientation relative to its environment. Fig. 12 (b) shows error distribution diagrams, in which the mean and median errors were less than (39.1mm, 40.21mm, 4.05°) and (30.72mm, 32.48mm, 3.71°), respectively. As these figures demonstrate, the proposed localization algorithm gave a very satisfactory performance in experimental testing.

V. Conclusion

In this paper, a new approach is presented for an object and spatial layout based hybrid map representation and global localization of mobile robots. The proposed map has a hybrid structure which includes global topological and local hybrid maps. And, on the basis of the map, it is possible to reliably



(a) (b)

Fig. 12. (a) Localization results for different test positions in the home-like environment. (b) Error distribution for the localization results.

estimate the pose of the robot. Global localization is performed in three stages: perception, coarse pose estimation and fine pose estimation. In extensive experiments carried out on a real robot and in an indoor environment, we showed that the proposed methods can be an effective and accurate global localization process.

In this paper, the map construction was not carried out fully autonomously. A future work is therefore planned involving the study of an object and spatial layout based SLAM system, i.e. how a mobile robot by itself can make the type of map described in this paper. The autonomous node position selection can be realized by using the vertices of the generalized Voronoi graph (GVG) [24]. In addition, the Voronoi edges can be used as paths to move between adjacent nodes. Based on the GVG, the robot can explore an unknown environment with a topological exploration strategy [24, 25].

REFERENCES

[1] D. Lee and W. Chung, "Discrete-Status-Based Localization for Indoor Service Robots," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 5, pp. 1737-1746, 2006.

[2] F. Lu and E. Milios, "Robot Pose Estimation in Unknown Environment by Matching 2D Range Scans," *IEEE Conf. on Computer Vision and Pattern Recognition*, 1994, pp. 935-938.

[3] G. Weiss, and E. Puttkamer, "A Map Based on Laserscans without Geometric Interpretation," *Proc. IAS-4*, 1995, pp. 403-407.

[4] G. N. Desouza and A. C. Kak, "Vision for Mobile Robot Navigation: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 237-267, 2002.

[5] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647-665, 2008.

[6] S. Se, D. Lowe and J. Little, "Mobile Robot Localization and Mapping with Uncertainty using Scale-Invariant Visual Features," *The International Journal of Robotics Research*, vol. 21, no. 8, pp. 735-758, 2002.

[7] R. Sim, P. Elinas, and J. J. Little, "A Study of the Rao-Blackwellised Particle Filter for Efficient and Accurate Vision-Based SLAM," *International Journal of Computer Vision*, vol. 74, no. 3, pp. 303-318, 2007.

[8] J. Košecká, L. Zhou, P. Barber, and Z. Duric, "Qualitative Image Based Localization in Indoors Environments," *IEEE Conf. on Computer Vision and Pattern Recognition*, 2003, pp. 3-10.

[9] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journa. of Computer Vision*, vol. 60, no. 2, pp. 91-110, 2004.

[10] M. Tomono and S. Yuta, "Indoor Navigation based on an Inaccurate Map using Object Recognition," *IEEE/RSJ Int. Conf. on Intelligent Robots an Systems*, 2002, pp. 399-405.

[11] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart, "Cognitive Maps for Mobile Robots – An Object based Approach.," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 359-371, 2007.

[12] A. Ranganathan and F. Dellaert, "Semantic Modeling of Places Using Objects," *Proceedings of Robotics: Science and Systems (RSS)*, 2007.

[13] W. K. Yeap and M. E. Jefferies, "On early cognitive mapping," *Spatial Cognition and Computation*, vol. 2, pp. 85-116, 2000.

[14] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J.A. Fernández-Madrigal and J. González, "Multi-Hierarchical Semantic Maps for Mobile. Robotics", *IEEE/RSJ Int. Conf. Intelligence Robots and Systems*, 2005, pp. 3492-3497.

[15] Y. Ke and R. Sukthankar, "PCA-SIFT: A More Distinctive Representation for Local Image Descriptors," *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2004, pp. 506-513.

[16] P. Althaus and H. I. Christensen, "Automatic Map Acquisition for Navigation in Domestic Environments," *IEEE Int. Conf. on Robotics and Automation*, 2003, pp. 1551-1556.

[17] S.-K. Park, M. Kim, and C. Lee, "Mobile Robot Navigation Based on Direct Depth and Color-based Environment Modeling," *IEEE Int. Conf. on Robotics and Automation*, 2004, pp. 4253-4258.

[18] M. Leordeanu and M. Herbert, "A Spectral Technique for Correspondence Problems using Pairwise Constraints," *IEEE Int'l Conf. on Computer Vision*, 2005, pp. 1482-1489.

[19] A. C. Berg, T. L. Berg, and J. Malik, "Shape Matching and Object Recognition using Low Distortion Correspondences," *IEEE Conf. on Computer Vision and Pattern Recognition*, 255, pp. 26-33.

[20] John J. Craig, *Introduction to Robotics: Mechanics and Control (3rd Edition)*, Prentice Hall, 2005.

[21] H. Zhuang and R. Sudhakar, "Simultaneous Rotation and Translation Fitting of Two 3-D Point Sets," *IEEE Trans. Syst., Man, and Cybernetics – Part B : Cybernetics*, vol. 27, pp. 127-131, 1997.

[22] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An Introduction to MCMC for Machine Learning," *Machine Learning*, vol. 50, pp. 5-43, 2003.

[23] I. Isard and A. Blake, "CONDENSATION – conditional density propagation for visual tracking," *Int. J. Computer Vision*, vol. 29, no. 1, pp. 5-28, 1998.

[24] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 2, 125-137, 2001.

[25] H. Cheong, S. Park, and S.-K. Park, "Topological Map Building and Exploration Based on Concave Nodes," *Int. Conf. on Control, Automation and Systems*, 2008, pp. 1115-1120.