Global Localization using Distinctive Visual Features*

Stephen Se
MD Robotics
9445 Airport Road
Brampton, Ontario L6S 4J3, Canada
sse@mdrobotics.ca

David Lowe, Jim Little
Department of Computer Science
University of British Columbia
Vancouver, B.C. V6T 1Z4, Canada
{lowe,little}@cs.ubc.ca

Abstract

We have previously developed a mobile robot system which uses scale invariant visual landmarks to localize and simultaneously build a 3D map of the environment In this paper, we look at global localization, also known as the kidnapped robot problem, where the robot localizes itself globally, without any prior location estimate. This is achieved by matching distinctive landmarks in the current frame to a database map. A Hough Transform approach and a RANSAC approach for global localization are compared, showing that RANSAC is much more efficient. Moreover, robust global localization can be achieved by matching a small sub-map of the local region built from multiple frames.

1 Introduction

Accurate localization is a prerequisite for building a good map, and having an accurate map is essential for good localization. Therefore, Simultaneous Localization And Mapping (SLAM) is a critical underlying factor for successful mobile robot navigation.

Many early successful approaches [1] utilize artificial landmarks to achieve SLAM, and do not function properly in beacon-free environments. Vision-based approaches using natural landmarks in unmodified environments are highly desirable for many applications.

There are two types of localization: local and global. Local techniques aim at compensating odometry errors. They require that the initial location of the robot is approximately known and they typically cannot recover if they lose track of the robot's position.

Global techniques can localize a robot without any prior knowledge about its position, i.e., they can handle the *kidnapped robot* problem, in which a robot is kidnapped and carried to some unknown location. Global localization techniques are more powerful than local ones and can cope with situations in which the robot is likely to experience serious positioning errors.

Markov localization was employed by various teams with success [10, 14]. For example, the Deutsches Museum Bonn tour-guide robot RHINO [2] utilizes a metric version of this approach with laser sensors. However, it needs to be supplied with a manually derived map, and cannot learn maps from scratch.

Unlike RHINO, the latest museum tour-guide robot MINERVA [15] learns its map and uses camera mosaics of the ceiling in addition to the laser scan occupancy map. It uses the EM algorithm to learn the occupancy map and the Markov localization with filter techniques for global localization.

The Monte Carlo Localization method based on the CONDENSATION algorithm was proposed in [4]. This vision-based Bayesian filtering method uses a sampling-based density representation and can represent multi-modal probability distributions. Given a visual map of the ceiling obtained by mosaicing, it localizes the robot globally using a scalar brightness measurement. [8] proposed some modifications for better efficiency in large symmetric environments.

Since the sensor information (sonar, laser or brightness measurements) only provides very low feature specificity, these methods are probabilistic and require the robot to move around, while the probabilities gradually converge towards one localized peak.

Learning natural visual features for pose estimation is proposed in [13]. Landmark matching is achieved using principal components analysis and a tracked landmark is a set of image thumbnails detected in the learning phase, for each grid position in pose space.

Using a panoramic image-based model for robot localization is proposed in [3]. A panoramic model is constructed with depth and 3D planarity information. The matching is based on the planar patches.

We have proposed a vision-based SLAM algorithm [12] by tracking SIFT (Scale Invariant Feature Transform) landmarks and building a 3D map simultaneously on our mobile robot equipped with Triclops, a trinocular stereo system.

In this paper, we consider global localization as a recognition problem, by matching the distinctive SIFT features detected in the current frame to the pre-built SIFT database map. A Hough Transform approach and a RANSAC approach are described and compared. Moreover, we consider global localization using features from multiple frames.

*Our work has been supported by the Institute for Robotics and Intelligent System (IRIS III), a Canadian Network of Centres of Excellence.

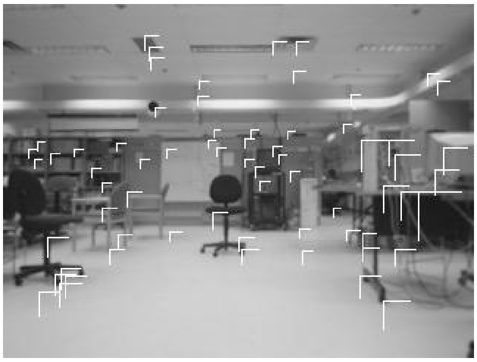


Figure 1: *SIFT stereo matching result, where horizontal and vertical lines indicate the horizontal and vertical disparities respectively.*

2 Mobile Robot Localization

Our vision-based mobile robot localization and mapping system uses SIFT visual landmarks in unmodified environments. By keeping the SIFT landmarks in a database, we track the landmarks over time and build a 3D map of the environment, and use these 3D landmarks for localization at the same time.

2.1 SIFT Stereo

SIFT was developed by Lowe [9] for image feature generation in object recognition applications. The features are invariant to image translation, scaling, rotation, and partially invariant to illumination changes and affine or 3D projection. Previous approaches to feature detection, such as the widely used Harris corner detector [6], are sensitive to the scale of an image and therefore are not suitable for building a map that can be matched from a range of robot positions.

SIFT keys are selected at maxima and minima of a difference of Gaussian function applied in scale space. At each feature location, an orientation is selected by determining the peak of a histogram of local image gradient orientations. A subpixel image location, scale and orientation are associated with each SIFT feature.

In our Triclops system, we have three images at each frame. In addition to the epipolar constraint and disparity constraint, we also employ the SIFT scale and orientation constraints for matching the right and left images. These resulting matches are then matched with the top image similarly. We can then compute the 3D world coordinates relative to the robot for each feature. They can subsequently serve as landmarks for map building and tracking. Figure 1 shows the SIFT stereo results for an image of resolution 320x240.

2.2 SLAM

To build a map, we need to know how the robot has moved between frames in order to put the landmarks together coherently. The robot odometry can only give a rough estimate and it is prone to error such as drifting, slipping, etc. To find matches in the second view, the odometry allows us to predict the region to search for each match more efficiently.

Once the SIFT features are matched, we can use the matches in a least-squares procedure to compute a more accurate camera ego-motion and hence better localization. This will also help adjust the 3D coordinates of the SIFT landmarks for map building.

We build a 3D map when the robot moves around in our lab environment. Figure 2 shows the bird's eye view of the map after 435 frames and there are 2783 SIFT landmarks in the database. The system currently runs at 2Hz on a Pentium III 700MHz processor. Readers are referred to [12] for further details.

2.3 Global Localization

Global localization is similar to a recognition problem where the robot tries to match the current view to a previously built map. The SIFT features used here were originally designed for object recognition purposes, therefore these visual landmarks are very suitable for global localization.

Apart from the scale and orientation, the local image region is described in a manner invariant to various image transformations [9] and we obtain enough measurements for high specificity. Robustness to local geometric distortion can be obtained by representing the local image region with multiple images representing local image gradients at a number of orientations.

We use 4 orientations, each sampled over a 2x2 grid of locations. The total number of samples in each SIFT local characteristic vector is $4 \times 2 \times 2$ or 16 elements. This vector size is sufficiently discriminating for this application and can be increased if necessary. Using this local image vector metric, we can simply compute the Euclidean distance between the vectors of two features to check how well they match.

We consider matching a set of SIFT landmarks to the database, using the distinctive visual information to localize a robot globally from the current view. Both the geometric and photometric information of the landmarks are utilized to facilitate matching.

We will describe two algorithms developed for global localization next, namely the Hough Transform approach and the RANSAC approach, followed by a comparison of the two approaches.

3 Hough Transform Approach

Given a set of current SIFT features and a set of SIFT landmarks in the database, we search for the robot position that would have brought the largest number of landmarks into close alignment, treating global localization as a search problem.

The Hough Transform [7] with a three-dimensional discretized search space (X, Z, θ) is used, where X is the sideways translation, Z is the forward translation and θ is the orientation. The algorithm is as follows [11]:

- For each SIFT feature in the current frame, find the set of N potential SIFT landmarks in the database that match, using the local image vector and the height as the preliminary constraints.

- For each of the potential matches, compute all the possible poses and vote the corresponding Hough bins. As robot pose cannot be uniquely determined from just one match, multiple bins are voted, covering an arc of robot poses at the estimated distance from the landmark.
- Vote also the neighbouring bins within the uncertainty region based on the landmark covariance.
- Select the top K bins and carry out least-squares minimization with outlier removal to obtain pose estimates. Select the one with maximum number of matches and lowest least-squares error. This corresponds to a robot pose which can best match the most features to the database.

4 RANSAC Approach

Global localization is performed by finding the robot pose supported by the most landmarks. This can be formulated as a hypothesis testing problem, where multiple pose hypotheses are considered and the best one corresponds to the pose which can match the most features in the current frame to the database.

RANSAC [5] has been used in many applications for model fitting, hypothesis testing and outlier removal. We employ RANSAC for global localization to test the pose hypotheses and find the inlier landmarks.

4.1 Tentative Matches

Firstly, we create a list of tentative matches from the features in the current frame to the landmarks in the database. For each feature in the current frame, we find the landmark in the database which is closest in terms of the local image vector and has similar height.

4.2 Computing the Alignment

Next, we randomly select 2 tentative matches from the list and compute the alignment parameters (X, Z, θ) from them. Two tentative matches are required in this case, since for each match, we can obtain 2 equations with 3 unknowns:

$$X = X_i - X'_i \cos\theta - Z'_i \sin\theta \qquad (1)$$

$$Z = Z_i - Z'_i \cos\theta - X'_i \sin\theta \qquad (2)$$

where (X_i, Y_i, Z_i) is the landmark position in the database and (X'_i, Y'_i, Z'_i) is the feature position in the current frame in camera coordinates.

Therefore, we need two matches, i and j. By equating the equations, we have:

$$A\cos\theta + B\sin\theta = C \qquad (3)$$

$$B\cos\theta - A\sin\theta = D \qquad (4)$$

where $A = X'_i - X'_j$, $B = Z'_i - Z'_j$, $C = X_i - X_j$, $D = Z_i - Z_j$. If the two tentative matches are correct, the distance between two landmarks should be invariant for the Euclidean transformation, so the following constraint is applied to each sample selection: $A^2 + B^2 \approx C^2 + D^2$. This efficiently eliminates many samples containing wrong matches from further consideration.

Solving Equations 3 and 4, we obtain:

$$\theta = \tan^{-1}\frac{BC - AD}{AC + BD}$$

and substituting this into Equations 1 and 2 gives an alignment.

4.3 Seeking Support

Now we check all the tentative matches which support this particular pose (X, Z, θ).

Firstly, we compute the landmark position for each match k relative to this pose:

$$\begin{aligned} X_p &= (X_k - X)\cos\theta - (Z_k - Z)\sin\theta \\ Y_p &= Y_k \\ Z_p &= (X_k - X)\sin\theta + (Z_k - Z)\cos\theta \end{aligned}$$

and then compute the image position (r_p, c_p) and disparity d_p for this landmark at this pose.

Match k supports this pose if (r_p, c_p) and d_p are close to the measured image position (r_m, c_m) and disparity d_m for the feature in the current frame, i.e., $|r_p - r_m| < \Delta_r$ and $|c_p - c_m| < \Delta_c$ and $|d_p - d_m| < \Delta_d$ (currently $\Delta_r = 5$, $\Delta_c = 5$, $\Delta_d = 2$).

4.4 Hypothesis with Most Support

This random selection, alignment computation and support seeking process is repeated m times. The pose with most support is our hypothesis. We then proceed with least-squares minimization for the inliers which support this hypothesis and obtain a better estimate for the final pose.

The probability of a good sample τ for RANSAC [5] is given by:

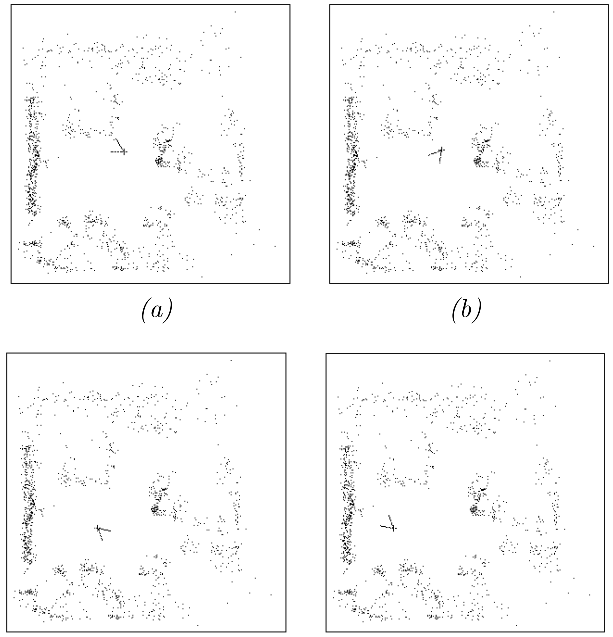
$$\tau = 1 - (1 - (1 - \epsilon)^p)^m \qquad (5)$$

where ϵ is the contamination ratio (ratio of false matches to total matches), p is the sample size and m is the number of samples required.

In this case, $p = 2$ as two matches are required to compute the alignment. The contamination ratio depends on how distinctive the features are, the database size as well as the environment. We will compare the effect of various contamination ratios in Section 6.

5 Experimental Results

Using the database map built earlier covering a 10m by 10m area, we test the robot at various positions. Both approaches give similarly good results. The following pose results (X, Z, θ) are obtained using the RANSAC approach with $m = 50$, where X and Z are in cm and θ is in degrees:



(a) *(b)* *(c)* *(d)*

Figure 2: *Global localization results with RANSAC. The vee indicates the robot field of view. (a) Case L1. (b) Case L3. (c) Case L5. (d) Case L8.*

Case	Measured Pose	Estimated Pose	Match
L1	(-10,120,-60)	(-13.3,127.6,-60.5)	35
L2	(50,210,-25)	(54.3,208.9,-25.6)	17
L3	(-15,130,-140)	(-16.0,134.9,-140.5)	32
L4	(-80,60,-150)	(-75.7,68.8,-148.6)	23
L5	(-100,0,130)	(-105.0,7.6,130.9)	50
L6	(30,-70,40)	(31.3,-64.9,38.5)	11
L7	(-170,20,-125)	(-175.2,21.8,-124.4)	52
L8	(-210,0,-50)	(-207.6,8.3,-49.0)	18

Measured pose is the approximate ground truth measured manually. The average Euclidean translation error is 7cm and the average rotation error is around 1° for these 8 cases. These errors could be further reduced by using higher image resolution but they are sufficient for our navigation requirement.

We currently set a minimum of 10 matches for a reliable estimation. Figure 2 shows some of these results visually, indicating the robot location and orientation relative to the database map.

Global localization fails when the robot is facing some landmarks which were previously viewed from very different directions during map building. Therefore, extensive landmarks all over the environment should be observed from multiple views during map building, to obtain a richer database map.

6 RANSAC versus Hough Transform

We would like to compare the computational efficiency of these two approaches of global localization. The following run-time results are based on a Pentium III 700MHz processor.

6.1 Hough Transform

In this approach, we have a 3-D Hough space for (X, Z, θ) where q, n and l are the number of bins for X, Z and θ respectively. For each of the potential matches, we need to vote in l bins since there are multiple robot poses that could have observed this landmark. The main computation includes computing the poses to vote and finding the peaks in the Hough space.

The pose computation time for one potential match for all features in the current frame at all orientations is t_1. As we find the best N matches for each feature, the pose computation takes Nt_1.

It takes t_2 to find the highest peak in the Hough space, which is proportional to the map dimension. Optimally, we can maintain a heap to avoid going through the Hough bins repeatedly. Then, at each of the K times, it will take only logarithmic time to retrieve the next peak instead of linear time. For now, we just simply going through the bins K times, so the time required is Kt_2. There is some overhead of t_3 as well and the total time taken is $Nt_1 + Kt_2 + t_3$.

For our experiment, the discretization used is (10cm,10cm,2°) with a Hough space of $q\,n\,l$ bins (currently $q = 100$, $n = 100$, $l = 180$), $t_2 = 0.05$. With $K = 10$, $N = 5$, $t_1 = 0.025$, $t_3 = 0.1$, the total time taken in this case is around 0.725 second.

6.2 RANSAC

For the RANSAC approach, the computational cost is affected greatly by how many times we need to sample, which depends on the contamination ratio, to achieve a certain probability of a good sample.

With $p = 2$ and $\epsilon = 1 - c/f$ where f is the number of features in the current frame and c is the number of correct matches, we can re-write Equation 5 as:

$$m_1 = \frac{\log(1-\tau)}{\log(1-c^2/f^2)} \approx -\frac{f^2}{c^2}\log(1-\tau) \qquad (6)$$

using Taylor's expansion as approximation.

For each random selection, we need to check the support from all the f tentative matches, as the tentative matches are obtained by considering each feature in the current frame one by one. The time required is $(f_t + t_4)$ where f_t is the time to check for support from f tentative matches and t_4 is a fixed overhead.

Therefore, the total cost is $(f_t + t_4)m_1 + t_5$ where t_5 is the time to create the list of tentative matches, which depends on the number of features in the current frame and the database size.

In our case, $f_t = 1.4 \times 10^{-5}$, $t_4 = 10^{-5}$, $t_5 = 0.02$, the total time is therefore $(1.4\times10^{-5} + 10^{-5})m_1 + 0.02$. Assuming a contamination ratio of 0.70, to achieve 99% probability of a good sample, m_1 is 50 and the time is around 0.02 second. RANSAC is much more efficient than the Hough Transform.

For larger values of the contamination ratio, a larger m_1 is required to maintain the probability of getting a good sample. The following table shows the number of samples and the time required for various contamination ratios. We can see that the required time is still quite short even when the contamination is high.

Contamination	Samples	Time (sec)
0.70	50	0.021
0.90	460	0.031
0.95	1840	0.064
0.98	11500	0.296

6.3 Discussion

With highly distinctive SIFT features, either the Hough Transform or the RANSAC approach will give a good estimate, with the RANSAC approach being more efficient. The computational cost increases linearly with the database size for both approaches.

When non-specific features are used, we need to consider all the possible matches between the current frame and the database landmarks. Therefore, using SIFT features is much more efficient as good matches found based on the SIFT distinctiveness facilitate the process considerably.

Moreover, when using less specific features, global localization is more difficult to achieve by just using information from one frame, because multiple possible robot poses may not be reliably differentiated. For sonar data in [14] and brightness measurements in [4], stochastic localization methods are required to localize the robot gradually while it moves around.

7 Map Alignment

Instead of using only the current frame for global localization, we now build a small sub-map of a local region from multiple frames and then align this sub-map to the database map. This approach is more robust for scenarios where the robot is facing a scene with very few SIFT landmarks.

To align two maps, we employ an algorithm very similar to the global localization algorithm above. Either the Hough Transform approach or the RANSAC approach can apply, but we consider RANSAC here due to its efficiency.

The process is the same as in global localization, except that during the support seeking stage we now use the world positions of the landmarks to check for support, instead of the image coordinates.

In this experiment, when the robot wants to localize itself globally, it rotates a little bit, from -15 degrees to 15 degrees and builds a sub-map of this local region using information from multiple frames.

Figure 3 shows the various sub-maps built at several test positions. There are 411 landmarks, 207 landmarks, 383 landmarks and 270 landmarks in the sub-maps respectively. There are significantly more landmarks than in just one frame, typically around 70.

Map alignment using RANSAC is then carried out between these sub-maps and the database map, we obtain the following results $(Xcm, Zcm, \theta°)$:

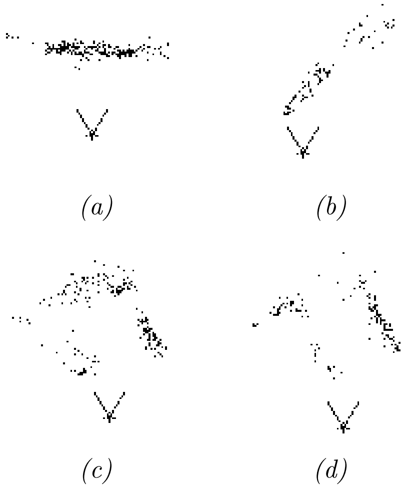


Figure 3: *Sub-maps built at test positions. (a) Case M1. (b) Case M2. (c) Case M3. (d) Case M4.*

Case	Measured Pose	Estimated Pose	Match
M1	(-110,30,-90)	(-104.8,30.5,-92.2)	191
M2	(-270,100,-45)	(-259.7,101.8,-43.5)	32
M3	(-130,100,-150)	(-125.9,90.3,-146.9)	143
M4	(60,310,-65)	(56.9,312.8,-63.5)	44

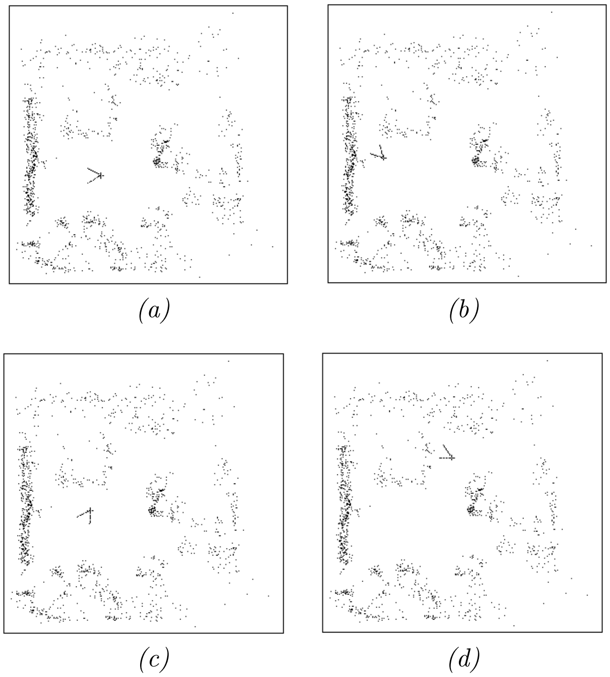
We can see that very good alignments are obtained with many matches in all cases. These global localization results are shown visually in Figure 4. If only the current frame is used for global localization here, there are insufficient matches in cases M2 and M4 for a reliable estimation.

8 Conclusion

In our previous work [12], we have built a database map with distinctive SIFT landmarks. We have developed a Hough Transform approach for global localization in [11]. In this paper, we proposed a RANSAC approach for matching SIFT features in the current frame to the database efficiently, to localize globally. We have demonstrated that the robot can globally localize itself well using SIFT features, even from just the current frame.

The contribution of this work includes the use of distinctive visual 3D landmarks for mobile robot localization, which has primarily been tackled in the past using 2D maps obtained from laser or sonar sensors. Moreover, the comparison of the commonly-used Hough Transform with RANSAC shows the greatly improved efficiency of RANSAC when matching distinctive features. Building sub-maps and then using them for global localization provides more robustness.

These algorithms have been implemented on our mobile robot, who won the first prize in the AAAI Hors d'Oeuvres competition 2001. The robots need to serve appetizers to people and return to the home base for refill when the food on the tray runs out. Our robot can estimate where it is and find its way home.



(a) (b)

(c) (d)

Figure 4: *Global localization by map alignment. The vee indicates the robot field of view. (a) Case M1. (b) Case M2. (c) Case M3. (d) Case M4.*

When global localization with the current frame is not certain due to the lack of features, the robot should rotate or move around. Therefore, it builds a small sub-map of the local region to match to the database for more robustness.

When the robot is in an area not in the database map or if the area has changed substantially from the original map, it will not be able to localize globally. The new region should be mapped and integrated with the existing map.

Maps can now be re-used as the robot knows where it is, it can continue to improve and augment the previous map. Using the same database map, multiple robots can localize themselves individually with reference to the same coordinate frame based on the visual landmarks they are looking at. Knowing the relative positions of the robots from each other is crucial for multi-robot collaboration, such as navigation, map building and other higher-level tasks.

A comprehensive database map is important for global localization. We are currently investigating some mobile robot exploration strategies to build a good map for the environment, where the robot would observe objects from various viewpoints. Moreover, experiments in larger environments are required to evaluate the scalability of the system.

References

[1] J. Borenstein, B. Everett, and L. Feng. *Navigating Mobile Robots: Systems and Techniques*. A. K. Peters, Ltd, Wellesley, MA, 1996.

[2] W. Burgard, A.B. Cremers, D. Fox, D. Hahnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. The interactive museum tour-guide robot. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence (AAAI)*, Madison, Wisconsin, July 1998.

[3] D. Cobzas and H. Zhang. Cylindrical panoramic image-based model for robot localization. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1924–1930, Maui, Hawaii, October 2001.

[4] F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the condensation algorithm for robust, vision-based mobile robot localization. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'99)*, Fort Collins, CO, June 1999.

[5] M.A. Fischler and R.C. Bolles. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Commun. Assoc. Comp. Mach.*, 24:381–395, 1981.

[6] C.J. Harris and M. Stephens. A combined corner and edge detector. In *Proceedings of 4th Alvey Vision Conference*, pages 147–151, Manchester, 1988.

[7] P.V.C. Hough. Method and means of recognizing complex patterns, December 1962. U.S. Patent 306965418.

[8] P. Jensfelt, O. Wijk, D.J. Austin, and M. Andersson. Experiments on augmenting condensation for mobile robot localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, April 2000.

[9] D.G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh International Conference on Computer Vision (ICCV'99)*, pages 1150–1157, Kerkyra, Greece, September 1999.

[10] I. Nourbakhsh, R. Powers, and S. Birchfield. Dervish: An office-navigating robot. *AI Magazine*, 16:53–60, 1995.

[11] S. Se, D. Lowe, and J. Little. Local and global localization for mobile robots using visual landmarks. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 414–420, Maui, Hawaii, October 2001.

[12] S. Se, D. Lowe, and J. Little. Vision-based mobile robot localization and mapping using scale-invariant features. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2051–2058, Seoul, Korea, May 2001.

[13] R. Sim and G. Dudek. Learning and evaluating visual features for pose estimation. In *Proceedings of the Seventh International Conference on Computer Vision (ICCV'99)*, Kerkyra, Greece, September 1999.

[14] R. Simmons and S. Koenig. Probabilistic robot navigation in partially observable environments. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1080–1087, San Mateo, CA, 1995. Morgan Kaufmann.

[15] S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Minerva: A second-generation museum tour-guide robot. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA'99)*, Detroit, Michigan, May 1999.