A Robust Localization Algorithm for Mobile Robots with Laser Range Finders[*]

Hee Jin Sohn and Byung Kook Kim

Department of Electrical Engineering & Computer Science
Korea Advanced Institute of Science and Technology
373-1 Guseong-dong, Yuseong-gu, Daejon, 305-701, Republic of Korea
hjsohn@rtcl.kaist.ac.kr, bkkim@ee.kaist.ac.kr

Abstract** – **We present a robust localization algorithm for mobile robots with laser range finders, which takes feature-based approach for reliable matching process as well as point-based approach for accurate and stable pose calculation. An efficient sequential segmentation algorithm is suggested, which also performs least square fitting processes simultaneously. Hence robust line segment features can be obtained with much less computation time. A novel cost function for pose calculation is suggested and formulated, which makes it possible to compute the location and orientation accurately and reliably. We examined the validity of the proposed algorithm with various simulations and experiments, and revealed its robustness and accuracy compared to other typical localization algorithms. The results show that the localization error of the proposed algorithm is 40% to 80% less than conventional localization algorithms.

Index Terms** – **mobile robot, localization, laser range finder.

I. INTRODUCTION

In order to accomplish given tasks, a mobile robot should be able to determine its pose (orientation and location), i.e. it should be able to localize itself. In map-based localization, a mobile robot decides its pose from the correlation between a given global environment map and measurement data from exteroceptive sensor(s) such as video camera [1], ultrasonic range finder [2][3], infrared range finder [4], laser range finder, and so on.

Among these sensors, laser range finder (LRF) has been most widely used in the localization due to its high accuracy and reliability [5][6][7][8][9][10][11]. Also a number of localization algorithms for LRF have been developed. We can classify them according to used data types: point-based localization algorithms [4][5][7][12] and feature-based localization algorithms [6][12][13][15][16][17][18].

A point-based localization algorithm compares two consecutive scan data and finds point-to-point correspondences. Then the algorithm computes least-squared solution by minimizing total squared sum of point-to-point distances. However, it is hard to find correct matching point pairs since the correspondence criteria only considers point property: The 'shape' is not considered which is a feature with a group of points.

On the other hand, a feature-based localization algorithm extracts several types of features from a set of raw scan data. Typical features may be line segments or distinct points such as corners. Various precise algorithms solve the localization problem by finding the orientation and location which minimize a certain cost function. However, they are less robust than point-based ones because the features they use may not reflect all the properties of scan data. Also the reliability of features may not be guaranteed. Beveridge's algorithms [18] are developed for image processing. They use two end points of each data line segment, hence the other points constituting the line segment are not considered. Borges' algorithm [16] uses both line and point features, but actual data describing a line feature is only one point. One problem is that this algorithm is very sensitive to the accuracy of features used.

In this paper, we propose a robust map-based localization algorithm for mobile robots with LRFs. The proposed algorithm combines raw measurement data with abstracted feature data effectively by defining a novel cost function, which leads to a robust algorithm against unexpected sensor noise and mismatch errors.

The proposed algorithm uses all the matched points in localization process. For matching process, it utilizes least square fitted line segments that are obtained by a sequential segmentation process, hence it is possible to match scan data with map reliably. For pose calculation, a novel cost function is used to reflect all the aspects of matched scan data, so the robustness is improved compared to other existing localization algorithms.

The rest of the paper is organized as follows. Section 2 proposes a robust localization algorithm. Sections 3 and 4 present simulation and experimental results respectively, and Section 5 concludes with some remarks.

II. ROBUST LOCALIZATION ALGORITHM

The proposed algorithm assumes that a mobile robot has a predefined global map which is composed of line segments, and equipped with a LRF for sensing external environment as well as encoders attached to driving motors for sensing the number of revolutions of each wheel.

The localization process is divided into three phases: line segment extraction from measured data, line segment matching between the global map and sensor data, and pose calculation from the matching result.

A. Line segment extraction

In order to segment laser scan data into piecewise linear sections, many researchers have used an iterative segmentation algorithm which originates from [20]. However, the iterative method spends unnecessary time in processing duplicated operations.

We present an efficient sequential segmentation algorithm. With N being the number of laser scan data, the sequential algorithm has an $O(N)$ computation time which is less than that of the iterative algorithm – $O(N \cdot \log N)$. Moreover, the sequential segmentation algorithm performs sequential least square fitting along with segmentation process by which we can obtain robust line segments with much less computation time and can also segment laser scan data more reliably.[†]

Fig. 1 shows how the segmentation process is accomplished. Laser scan data of N points are defined as

$$D = \{D_i = (x_i, y_i) \mid i = 1...N\}. \tag{1}$$

They are segmented into

$$G = \{G_i \mid i = 1...N_G\} \tag{2}$$

where G_i is a group of i-th segmented scan data and has N_i points:

$$G_i = \{D_{ij} \mid j = 1...N_i, D_{ij} \in D\}. \tag{3}$$

On the other hand, each G_i is sequentially least square fitted and they are grouped into L:

$$L = \{L_i \mid i = 1...N_G\} \tag{4}$$

where

$$L_i = \{\theta_i^G, \rho_i^G, (x_i^s, y_i^s), (x_i^e, y_i^e)\}. \tag{5}$$

θ and ρ are line parameters when the line is described as

$$x \cdot \cos\theta + y \cdot \sin\theta = \rho \tag{6}$$

and (x_i^s, y_i^s), (x_i^e, y_i^e) are start and end points of L_i respectively, which are the projection of D_{i1} and D_{iNi} on the line.

Let $G_i = \{D_s,...,D_j\}$ be a previously-identified segment, and L_i be the least square fitted line from G_i. The line parameters of L_i are given by (a_j, b_j) as in the APPENDIX. In order to check whether D_{j+1} can be included in G_i, r_{j+1} - the perpendicular distance from D_{j+1} to L_i - is tested whether it is within the predefined threshold. If r_{j+1} is within the range, D_{j+1} is included in G_i, the parameters of L_i is updated to (a_{j+1}, b_{j+1}) by sequential least square fit as described in (26) and (27), and the next point D_{j+2} is checked for inclusion. If not, segmentation process for G_i is finished. The line parameters (a_j, b_j) of L_i are converted to (θ_i^G, ρ_i^G) format and new segmentation process begins from D_{j+1}.

After all, we can get a group of segments G and corresponding lines L. L is used in line segment matching process and G is used in pose calculation process.

B. Matching line segments

We set three coordinate systems for localization process: the global coordinates Σ_W on which global map is based, the local coordinates Σ_R attached to the real robot pose, and the local coordinates Σ_E related to the robot pose estimated by odometry.
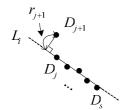

Fig. 1. Sequential segmentation example.

We assume that the error of odometry is small enough within short moving distance. In this case, the robot pose estimated by odometry has also a small error compared to the real robot pose, which means the line segments in the global map transformed to Σ_E are close to the corresponding data line segments L based on Σ_R. Hence it is enough to use a simple matching algorithm.

Let M be a group of N_M global map line segments based on Σ_E:

$$M = \{M_k \mid k = 1...N_M\}. \tag{7}$$

where M_k includes its line parameters with start and end points:

$$M_k = \{\theta_k^M, \rho_k^M, (x_k^s, y_k^s), (x_k^e, y_k^e)\}. \tag{8}$$

Matching sensor data to a global map means finding the corresponding map line segment for each data line segment. When M_k and G_i satisfy the following equations

$$|\theta_i^G - \theta_k^M| \le \sigma_\theta \tag{9}$$

$$|\rho_i^G - \rho_k^M| \le \sigma_\rho \tag{10}$$

where σ_θ and σ_ρ are given thresholds selected heuristically, they are considered to be a matching pair.

C. Pose calculation

From the matching result, the robot pose can be calculated by minimizing the following cost function.

Fig. 2 describes a matching pair (G_i, M_k). Let d_{ij} be the perpendicular distance from a data point of G_i to the map line segment M_k, l_i be the length of the data line segment L_i, and σ_i^2 be the variance of the distances between data points G_i and the data line segment L_i. When ϕ and $P = [p_x \ p_y]^T$ are defined as rotation angle and translation vector respectively, d_{ij} is a function of ϕ and P. The proposed cost function using $d_{ij}(\phi, P)$ is

$$E(\phi, P) = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} d_{ij}^2(\phi, P) \tag{11}$$

where N_S is the number of matching pairs.

Minimizing the cost function $E(\phi, P)$ means minimizing the sum of "distances" between scan data and the corresponding map line segments. The "distance" defined in this cost function is the sum of squared perpendicular distances from all matched data points to the corresponding map line segment. By considering all matched scan data, we can improve localization robustness better than using only one or two data points, which is the difference compared to other existing cost functions. Moreover, a G_i with more points gets higher weight, which means that a segment with more data is more reliable. The value l_i/σ_i^2 gives additional weight to G_i which means that a segment with longer length and less variance is trustier. l_i can be obtained from end points of each L_i, and σ_i^2 is natural by-product of sequential segmentation algorithm. See APPENDIX.
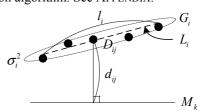

Fig. 2. Definition of a cost function.

The following localization formula is similar to Beveridge's [18], but he inserted a scale factor s which is usual case for image processing, but not for LRF data. $d_{ij}^2(\phi,P)$ in (11) can be expanded as

$$d_{ij}^2(\phi,P) = (m_i \cdot (\Delta_{ij} \cdot \Phi + P) - \rho_i^M)^2 \qquad (12)$$

where

$$m_i = [\cos\theta_i^M \quad \sin\theta_i^M], \quad \Delta_{ij} = \begin{bmatrix} x_{ij} & -y_{ij} \\ y_{ij} & x_{ij} \end{bmatrix}, \quad \Phi = \begin{bmatrix} \cos\phi \\ \sin\phi \end{bmatrix}. \quad (13)$$

Expanding (11) with (12) results in the following equation:

$$E(\Phi,P) = \Phi^T \cdot A \cdot \Phi + 2P^T \cdot B \cdot \Phi + P^T \cdot C \cdot P \\ - 2U^T \cdot \Phi - 2V^T \cdot P + k \qquad (14)$$

where

$$A = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} \Delta_{ij}^T \cdot m_i \cdot m_i^T \cdot \Delta_{ij}, \quad B = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} m_i \cdot m_i^T \cdot \Delta_{ij}$$

$$C = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} m_i \cdot m_i^T, \quad U = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} \rho_i^M \cdot \Delta_{ij}^T \cdot m_i \qquad (15)$$

$$V = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} \rho_i^M \cdot m_i, \quad k = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} (\rho_i^M)^2.$$

$E(\Phi,P)$ may have multiple minima. After finding all local minima, we can select the one satisfying physical criterion. All local minima of $E(\Phi,P)$ satisfy the following set of equations:

$$\frac{\partial E(\phi,P)}{\partial \Phi} = \frac{\partial E(\phi,P)}{\partial P} = 0. \qquad (16)$$

From (16), it is possible to express P as a function of Φ:

$$\frac{\partial E}{\partial P} = 2B\Phi + 2CP - 2V = 0$$

$$P = C^{-1}(V - B\Phi). \qquad (17)$$

Further development of (14) with (17) results as:

$$E(\Phi) = \Phi^T F \Phi + 2W^T \Phi + k' \qquad (18)$$

where

$$F = A - B^T (C^{-1})^T B$$
$$W = B^T (C^{-1})^T V - U \qquad (19)$$
$$k' = k - V^T (C^{-1})^T V.$$

Now E becomes a second-order polynomial of $\cos\phi$ and $\sin\phi$ with the constraint $\cos^2\phi + \sin^2\phi = 1$. According to Lagrange's theorem [19], E can be converted into a fourth-order equation of $\cos\phi$ or $\sin\phi$. By the way, $\sin\phi$ is a monotonically increasing function in the range of $|\phi| \leq \pi/2$ which is a common situation for localization. Hence further expansion of the above equations for $\sin\phi$ results in

$$\lambda^4 + a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0 \qquad (20)$$

where $\lambda = \sin\phi$, and $a_3,\ldots,a_0$ are constants obtained from the expansion. The solutions for (20) are given in closed forms:

$$\lambda_1 = -a_3/4 + k_1/2 + k_2/2$$
$$\lambda_2 = -a_3/4 + k_1/2 - k_2/2$$
$$\lambda_3 = -a_3/4 - k_1/2 + k_3/2 \qquad (21)$$
$$\lambda_4 = -a_3/4 - k_1/2 - k_3/2$$

where

$$k_1 = \sqrt{\frac{1}{4}a_3^2 - a_2 + u}$$

$$k_2 = \sqrt{\frac{3}{4}a_3^2 - k_1^2 - 2a_2 + \frac{1}{4}(4a_3a_2 - 8a_1 - a_3^3)k_1^{-1}} \qquad (22)$$

$$k_3 = \sqrt{\frac{3}{4}a_3^2 - k_1^2 - 2a_2 - \frac{1}{4}(4a_3a_2 - 8a_1 - a_3^3)k_1^{-1}}$$

and u is a constant originated from solving (20).

We can verify the physical solution from (21) using several constraints. Rotation angle can be solved from the selected λ as

$$\phi = \sin^{-1}\lambda \qquad (23)$$

and translation vector P can be calculated by inserting the obtained ϕ into (17).

III. SIMULATIONS

We have performed various simulations with three different scenarios to compare the performance of our algorithm with existing algorithms. The scenario outline is as follows: a virtual robot moves along a given path in a virtual environment. The robot with LRF conducts scanning 50 times along the path. At each step, localization algorithms to be compared perform individual localization processes under exactly the same condition and store the localization errors (rotation and translation errors) in memory. After 50 times of scanning have finished, localization errors at every step are accumulated and compared.

Three algorithms are selected to be compared with the proposed algorithm: Two algorithms proposed by Beveridge[18] and one from Borges[16], since they show relatively accurate localization results compared to other algorithms, and approaches taken by them are similar to the proposed algorithm – they obtain localization results by minimizing their own cost functions. The cost functions for each algorithm are listed in TABLE I. *Proposed* means the proposed algorithm, *Beveridge-2P* and *Beveridge-ISPD* are Beveridge's two algorithms where *2P* implies the algorithm that uses two end points of each data line segment and *ISPD* indicates the algorithm that utilizes integrated squared perpendicular distance, and *Borges* stands for the Borges' algorithm.

TABLE I
LOCALIZATION ALGORITHMS TO BE COMPARED

Algorithm	Cost function
Proposed	$E_P(\phi,P) = \sum_{i=1}^{N_S} \frac{l_i}{\sigma_i^2} \sum_{j=1}^{N_i} d_{ij}^2(\phi,P)$
Beveridge - 2P	$E_{2P}(\phi,P) = (\frac{1}{L_D}) \sum_{i=1}^{N_S} \frac{l_i}{2}(v_{i1}^2 + v_{i2}^2)$
Beveridge - ISPD	$E_{ISPD}(\phi,P) = (\frac{1}{L_D}) \sum_{i=1}^{N_S} \frac{l_i}{3}(v_{i1}^2 + v_{i1}v_{i2} + v_{i2}^2)$
Borges	$E_{BO}(\phi,P) = \sum_{i=1}^{N_S} \mu_i \cdot d_i^2(\phi,P)$

TABLE II
LIST OF SCENARIOS FOR SIMUNATION

Scenario	Explanation
Scenario 1	[Normal situation] The variances of all data line segments are the same and perfectly matched with map line segments.
Scenario 2	[Different wall reflectance] The data variances are different by wall characteristics.
Scenario 3	[Mismatch error] Some line segments from obstacles are matched.

The simulation was performed in a 6mX6m virtual squared room by three scenarios explained in TABLE II. Scenario 1 is for normal situation. Scenario 2 is for when there exist walls that show different reflectance. Scenario 3 is for when mismatch errors occur. MATLAB is chosen for simulation tool.

A. Scenario 1

Fig. 3 shows a sample scan data and map for scenario 1. A virtual robot is placed at the center of the room and rotates counterclockwise. It scans $180°$ range in the front with $0.5°$ resolution and conducts all localization algorithms in TABLE I. After line segment extraction and line segment matching, localization algorithms run under the same condition. All localization errors are accumulated respectively and compared according to their relative magnitude. See TABLE III. Except Borges' algorithm, all the other algorithms show similar results.

B. Scenario 2

Scenario 2 uses the same environment as scenario 1 except scan data from two walls show five times more variances. These two walls are marked with dotted ovals in Fig. 4. Because of their bigger variance, scan data from them are less reliable than those from the other walls. This aspect is reflected on the proposed cost function.

The proposed algorithm shows better performance than other algorithms. Particularly, the rotation error of proposed algorithm is about 60% less than Beveridge's algorithm as shown in TABLE IV.

C. Scenario 3

In this scenario, some obstacles are positioned close to the walls resulting that the data segments from the obstacles are mismatched to the map line segments as depicted in Fig. 5. Obstacles marked by dotted circles are drawn with map line segments for convenience, but they are not included in the map. Scan data in a dotted rectangle are obtained from the rectangular obstacle in the north. They are matched with the nearby map line segment since they satisfy the matching criteria (9) and (10). As can be seen in TABLE V, the proposed algorithm shows great performance compared to the other algorithms. The translation error is 28%, and the rotation error is 41% of Beveridge's ISPD algorithm, which shows the robustness of the proposed algorithm against mismatch errors.
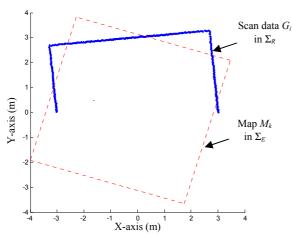

Fig. 3. Sample scan data and map line segments in scenario 1.

TABLE III
ACCUMULATED LOCALIZATION ERRORS IN SCENARIO 1

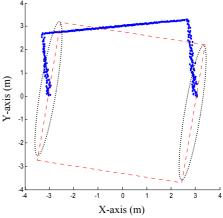| Algorithms	Translation error		Rotation error	
	Absolute (m)	Relative (%)	Absolute (rad)	Relative (%)
Proposed	0.083088	100	**0.024714**	100
Beveridge-2P	**0.081829**	98.48	0.025341	102.53
Beveridge-ISPD	0.082154	98.88	0.026066	105.47
Borges	2.375371	2858	0.362489	1466

Fig. 4. Sample scan data and map line segments in scenario 2.

TABLE IV
ACCUMULATED LOCALIZATION ERRORS IN SCENARIO 2

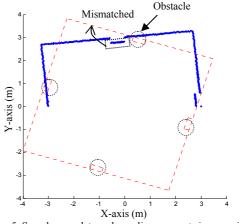| Algorithms	Translation error		Rotation error	
	Absolute (m)	Relative (%)	Absolute (rad)	Relative (%)
Proposed	**0.202310**	100	**0.037649**	100
Beveridge-2P	0.208115	102.87	0.063054	167.48
Beveridge-ISPD	0.207576	102.60	0.060877	161.70
Borges	3.023878	1494	0.773088	2053

Fig. 5. Sample scan data and map line segments in scenario 3.

TABLE V
ACCUMULATED LOCALIZATION ERRORS IN SCENARIO 3

Algorithms	Translation error		Rotation error	
	Absolute (m)	Relative (%)	Absolute (rad)	Relative (%)
Proposed	**0.239050**	100	**0.037770**	100
Beveridge-2P	0.884320	369.93	0.137570	364.23
Beveridge-ISPD	0.827993	346.37	0.091713	242.82
Borges	3.793253	1587	1.432189	3792

IV. EXPERIMENTAL RESULTS

The experimental evaluation allowed us to compare the performance of the algorithms in TABLE I in a real environment. We used an LRF of LMS 200 from SICK which is connected to Linux/RTAI (Real-Time Application Interface) based Pentium III-500MHz industrial PC through RS422 interface. The scan range of LMS 200 is 8 m, the scan angle range is 180°, and the angle resolution is set to 0.5°. Simple polygonal environment has been constructed with caution to take precise measurement.

The experiments have been performed with three scenarios - the same as the previous simulations. The localization errors of all algorithms are accumulated respectively during five times of measurements and compared each other.

A. Scenario 1

Fig. 6 shows sample scan data and map for scenario 1. Numbered solid line segments are obtained from scan data through the sequential segmentation algorithm and dotted line segments are from global map based on Σ_E.
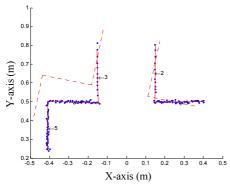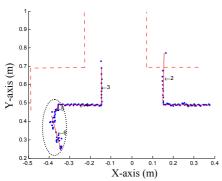

Fig. 6. Experimental scan data for scenario 1.

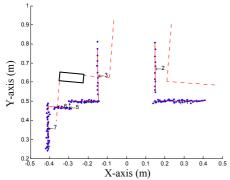Fig. 7. Experimental scan data for scenario 2.

Fig. 8. Experimental scan data for scenario 3.

Cumulative localization errors with respect to each algorithm are listed in TABLE VI.

As is the case in Section 3, proposed and Beveridge's algorithms show similar results, but Borges' algorithm exhibits localization errors two times larger than the others.

B. Scenario 2

Fig. 7 shows the scan data when there is a wall that has different reflectance. The wall marked with a dotted oval in Fig. 7 is wrapped with a sheet of packing foam to endow unexpected reflectance. TABLE VII confirms this fact. The accumulated localization errors with respect to each algorithm are listed in TABLE VIII. The proposed algorithm shows the best performance.

C. Scenario 3

Fig. 8 shows the case when an obstacle is close to a wall. The line segment numbered 6 is obtained from an obstacle but matched to the nearby wall because it satisfies matching criteria. The accumulated localization errors with respect to each algorithm are listed in TABLE IX, telling us that the proposed algorithm shows the best performance.

Additional remarks: Experimental results are different from those of simulation because the environments for simulation and experiments are not the same.

TABLE VI
ACCUMULATED LOCALIZATION ERRORS IN SCENARIO 1

Algorithms	Translation error		Rotation error	
	Absolute (m)	Relative(%)	Absolute (rad)	Relative(%)
Proposed	**0.04689**	100	0.02620	100
Beveridge- 2P	0.04901	104.52	**0.02385**	91.03
Beveridge- ISPD	0.04955	105.67	0.02983	113.86
Borges	0.09763	208.2	0.08725	333.0

TABLE VII
VARIANCES ACCORDING TO LINE SEGMENTS

Number of line segment	Variance [x 10^{-3}]
1	1.328
2	1.265
3	0.088
4	0.643
5	**4.914**
6	**7.329**

TABLE VIII
ACCUMULATED LOCALIZATION ERRORS IN SCENARIO 2

Algorithms	Translation error		Rotation error	
	Absolute (m)	Relative(%)	Absolute (rad)	Relative(%)
Proposed	**0.06047**	100	**0.02459**	100
Beveridge- 2P	0.19379	320.45	0.18840	766.03
Beveridge- ISPD	0.18152	300.17	0.16508	671.21
Borges	0.29374	485.73	0.83810	3408

TABLE IX
ACCUMULATED LOCALIZATION ERRORS IN SCENARIO 3

Algorithms	Translation error		Rotation error	
	Absolute (m)	Relative(%)	Absolute (rad)	Relative(%)
Proposed	**0.04795**	100	**0.05081**	100
Beveridge- 2P	0.10824	225.74	0.15730	309.58
Beveridge- ISPD	0.10106	210.76	0.14012	275.77
Borges	0.15395	321.06	0.12206	240.22

V. Conclusions

We presented a robust map-based localization algorithm for mobile robots with LRFs. An efficient sequential segmentation algorithm for extracting line segments from scan data is suggested, which also performs sequential least square fit on the data. The proposed algorithm combines raw measurement data with extracted line segments effectively by defining a novel cost function, which leads to a robust algorithm against unexpected sensor noise and mismatch error. A robust localization algorithm without scale factor is formulated. Simulation and experimental results have verified the fact that the proposed algorithm is more robust than the compared algorithms in the occurrences of unexpected wall characteristics and mismatch errors.

Future works include experiments with proposed localization algorithm in real-world robot systems and finding a solution for SLAM problem showing improved robustness with this robust localization algorithm.

Appendix
Sequential Least Square Fit

A group of points $\{ (x_i, y_i) \mid i=1..n \}$ is least square fitted to a line $y=a_n x + b_n$ where the parameters are given by

$$b_n = \frac{\sum_{i=1}^{n} x_i y_i - n \bar{x}_n \bar{y}_n}{\sum_{i=1}^{n} x_i^2 - n \bar{x}_n} \tag{24}$$

$$a_n = \bar{y}_n - b_n \bar{x}_n .$$

and $\bar{x}_n$, $\bar{y}_n$ are defined as

$$\bar{x}_n = \sum_{i=1}^{n} x_i$$
$$\bar{y}_n = \sum_{i=1}^{n} y_i . \tag{25}$$

When a point (x_{n+1}, y_{n+1}) is added, the line parameters are updated as follows:

$$b_{n+1} = \frac{\sum_{i=1}^{n+1} x_i y_i - (n+1) \bar{x}_{n+1} \bar{y}_{n+1}}{\sum_{i=1}^{n+1} x_i^2 - (n+1) \bar{x}_{n+1}} \tag{26}$$

$$a_{n+1} = \bar{y}_{n+1} - b_{n+1} \bar{x}_{n+1}$$

where

$$\bar{x}_{n+1} = \frac{n \bar{x}_n + x_{n+1}}{n+1}$$
$$\bar{y}_{n+1} = \frac{n \bar{y}_n + y_{n+1}}{n+1}$$
$$\sum_{i=1}^{n+1} x_i y_i = \sum_{i=1}^{n} x_i y_i + x_{n+1} y_{n+1} \tag{27}$$
$$\sum_{i=1}^{n+1} x_i^2 = \sum_{i=1}^{n} x_i^2 + x_{n+1}^2 .$$

Also the variance of the points which is used in the proposed cost function is obtained naturally and sequentially from the above equations:

$$\sigma_n = \frac{1}{n} \sum_{i=1}^{n} y_i^2 - \bar{y}_n . \tag{28}$$

Fitting to a line $x=a_n y + b_n$ has similar process. In this paper, both types of line equations are used with regard to their slopes. After a_n and b_n are determined, θ and ρ for (6) can be obtained from them.

References

[1] G. Dudek and C. Zhang, "Vision-based robot localization without explicit object models," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, pp. 76-81, 1996

[2] A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE Journal of Robotics and Automation*, vol. RA-3, no. 3, pp. 249-265, 1987

[3] G. C. Anousaki and K. J. Kyriakopoulos, "Simultaneous localization and map building for mobile robot navigation," *IEEE Robotics and Automation Magazine*, vol. 6, no. 3, pp. 42-53, 1999

[4] I. J. Cox, "Blanche-an experiment in guidance and navigation of an autonomous robot vehicle," *IEEE Transactions on Robotics and Automation: a publication of the IEEE Robotics and Automation Society*, vol. 7, no. 2, pp. 193-204, 1991

[5] T. Einsele, "Localization in indoor environments using a panoramic laser range finder," *Thesis for a Doctorate*, Technischen Universität, 2000

[6] G. S. Lionis and K. J. Kyriakopoulos, "A laser scanner based mobile robot SLAM algorithm with improved convergence properties," *Proceedings -IEEE International Conference on Intelligent Robots and Systems*, vol. 1, pp. 582-587, 2002

[7] J. G. Mota and M. I. Ribeiro, "Mobile robot localization on reconstructed 3D models," *Journal of Robotics and Autonomous Systems*, vol. 31, no. 1, pp. 17-30, 2000

[8] N. Ayache and O. Faugeras, "Hyper: a new approach for the recognition and positioning of 2D objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 44-54, 1986

[9] Y. D. Kwon and J. S. Lee, "A stochastic map building method for mobile robot using 2-D laser range finder," *Autonomous Robots*, vol. 7, no. 1, pp. 187-200, 1999

[10] C. Chang, C. Tsai, J. Hsu, and C. Lin, "Laser pose tracking for a mobile robot using fuzzy adaptive extended information filtering," *American Control Conference*, vol. 3, pp.2471-2476, 2003

[11] L. Zhang and B. K. Ghosh, "Line segment based map building and localization using 2D laser rangefinder," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2538-2543, 2000

[12] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans," *Journal of Intelligent & Robotic Systems*, vol. 18, no. 3, pp. 249-275, 1997

[13] R. Mázl, M. Kulich, and L. Preucil, "Range scan-based localization methods for mobile robots in complex environments," *Proceedings - IEEE Intelligent Transportation Systems Conference*, pp. 280-285, 2001

[14] C. Pradalier and S. Sekhavat, "Concurrent matching, localization and map building using invariant features," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 1, pp. 514-520, 2002

[15] X. Zezhong, L. Jilin, and X. Zhiyu, "Map building and localization using 2D range scanner," *Proceedings - IEEE Symposium on Computational Intelligence in Robotics and Automation*, vol. 2, pp.848-853, 2003

[16] G. A. Borges and M. J. Aldon, "Optimal mobile robot pose estimation using geometrical maps," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 1, pp. 87-94, 2002

[17] C. Chen and H. Wang, "Adaptive pose and location estimation for indoor mobile robot," *Proceedings - IEEE Intelligent Transportation Systems*, vol. 2, pp.1616-1621, 2003

[18] J. R. Beveridge and E. M. Riseman, "How easy is matching 2D line models using local search?," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 564-579, 1997

[19] K. P. Chong and H. Żak, *An Introduction to Optimization*, 2nd Edition, John Wiley & Sons, 2001

[20] T. Pavlidis and S. Horowitz, "Segmentation of planar curves," *IEEE Transactions on Computers*, vol. 23, pp. 860-870, 1974