Three practical localization issues towards dependable navigation

Chang-bae Moon, Woojin Chung*, Kyoung-rok Kim and Jae-Bok Song
Department of Mechanical Engineering
Korea University, Seoul, KOREA
smartrobot@ korea.ac.kr

Abstract— **A mobile robot localization problem can be classified into three sub-problems: A sensor model, a motion model and a filtering technique. So far, we have developed the range sensor based, integrated localization scheme, which can be used in human-coexisting real environment such as a science museum and office buildings. From those experiences, we found out that there are several significant issues to be solved. In this paper, we focus on three key issues, and then illustrate our solutions to the presented problems. Three issues are listed as follows:**

(1) Investigation of design requirements of a desirable sensor model, and performance analysis of our design

(2) Performance evaluation of the localization result by computing the *matching error*

(3) The *semi-global localization* scheme to deal with localization failure due to abrupt wheel slippage

In this paper, we show the significance of each concept, developed solutions and the experimental results. Experiments were carried out in a typical modern building environment, and the results clearly show that the proposed solutions are useful to develop practical, integrated localization schemes.

I. Introduction

A mobile robot localization problem can be classified into three sub-problems: A sensor model, a motion model and a filtering technique. A sensor model design includes a selection of sensor types, map representations and map-matching functions. A motion model design is involved with the uncertainty representation of robot's motion. A filtering technique is in charge of mathematical representation of probabilities and the localization synthesis.

Recently, there have been many research activities on the mobile robot localization problem. One typical example is Monte Carlo Localization (MCL) in [6]. MCL uses probabilistic representation of positional uncertainties, and the scheme is widely used due to its superior robustness, capability to deal with both global and local localization problems, and a simple mathematical representation. Owing to a lot of recent research achievements, the autonomous navigation problem can be more easily solved in a variety of real environments. Comparison of some algorithms can be seen in [7].

However, developing a localization scheme for an inexperienced newcomer is still a difficult problem, mainly because it is hard to find out a structural, quantitative design strategy in existing works. In addition, we noticed that appropriate performance evaluation of the localization results in real time is essential to survive in dynamic environment.

So far, we have developed range sensor based, integrated localization scheme, which can be used in human-coexisting real environment such as a science museum and office buildings. Our previous works can be seen in [2] and [1]. We have proposed the MCL based robust mapping, localization and synthesis scheme. From those experiences, we found out that there are several significant issues to be solved. In this paper, we focus on three key issues, and then illustrate our solutions to the presented problems. Three issues are listed as follows:

(1) Investigation of design requirements of a desirable sensor model, and performance analysis of our design.

(2) Performance evaluation of the localization result by computing the *matching error*.

(3) The *semi-global localization* scheme to deal with localization failure due to abrupt wheel slippage.

The above first issue is on the structural design of a sensor model. Appropriate sensor model design leads to achieving the fast convergence speed and the robustness under various uncertainties, which are trade-off characteristics. Those performances are dependent upon selection of the map-matching and probability mapping functions. We first establish design requirements of the sensor model, and then present developed map-matching and probability mapping functions. The usefulness of the proposed design is verified through the presented experiments.

The above second issue deals with the question "Is the computed localization result believable?" If we use the previously developed localization schemes, localization performance is satisfactory in most "normal" conditions. However, there exist a lot of exceptional situations such as too much sensor data corruption by visitors, or abrupt wheel slippage at bumps. In those cases, sensor data should be neglected in computations. From our experience, we noticed that the evaluation of the computed result is much more significant than the positional accuracy itself. We propose the concept of *matching error* to decide the localizer status *i.e.*, "*localizer success*" or "*localizer warning*."

The third issue to propose is the *semi-global localization* scheme. Once the localization fails, samples should be redistributed over the entire region to globally localize the robot, in conventional approaches as in [3]. However, the computational cost becomes extremely high. In our approach, the sample distribution area can be limited into a small region by exploiting the history of localization status.

The rest part of this paper is organized as follows: Design requirements and the proposed design of a sensor model are presented in section 2. Section 3 proposes the *semi-global localization*. Experimental results are presented in section 4. Finally some concluding remarks are given in section 5.

II. SENSOR MODEL DESIGN

A. *Sensor Model Requirements*

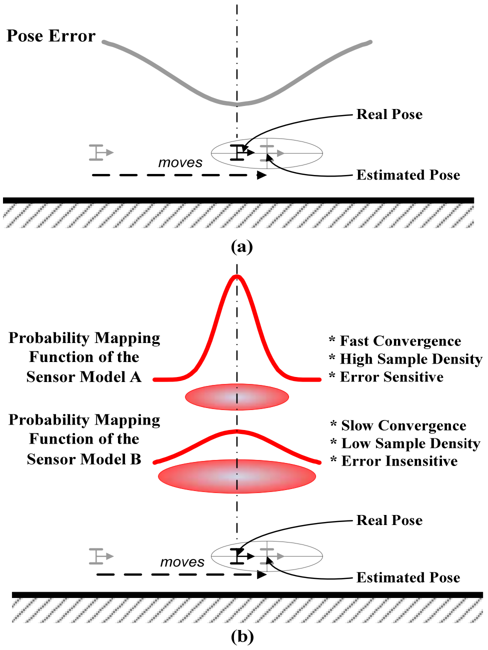


Fig. 1. (a) Illustration of the pose error distribution (b) Conceptual illustration of trade-off characteristics in a sensor model design

The first step of the sensor model design is to determine the error function of map-matching. It is natural that the map-matching error should be minimized at the real pose, and the error should increase if the pose of the computing reference sample goes far from the real position. This is a fundamental requirement for successful localization. The next step is to design a mapping function from the errors to the probability distribution. A lot of candidates can be considered for the mapping, and the convergence performance is affected by the choice of the mapping function.

If the probability mapping is defined as $p(E)$, we focus on the sensitivity $\partial p(E)/\partial E$. If the sensitivity is high, the convergence speed is fast. However, robustness with respect to uncertainties might be low. If the sensitivity is low, the convergence speed might be slow. However, the MCL solution might converge even when positional uncertainties are distributed over the wide area, which implies robustness with respect to uncertainties. This fact implies that the convergence property is dependent upon the choice of $p(E)$. Therefore, the sensor model should be carefully designed by considering the convergence property. The conceptual illustration of $p(o_t | s_{i,t}, m)$ design is shown in Fig. 1. From our experiences, we summarize the design requirements of the sensor model as the following three issues:

(a) The map-matching error should be minimized at the real pose, and the error should increase if the pose of the computing reference sample goes far from the real position.

(b) The probability mapping function should be carefully designed to achieve desirable convergence properties. The convergence property indicates the convergence speed and the area of convergence.

(c) The designed error function should be insensitive with respect to the partial corruption of sensor data. The partial corruption takes place when dynamic obstacles exist around the robot.

B. *Sensor Model Design*

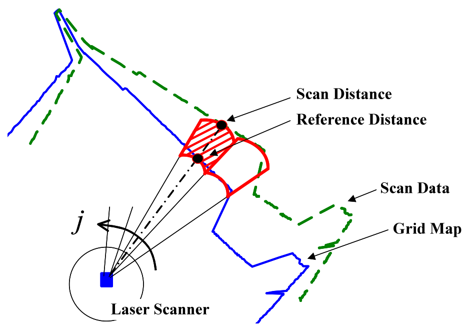


Fig. 2. Definition of the Matching Error

Fig. 2 illustrates a designed map-matching error function. Basically, the error E is defined as the area difference between the free space of scanned image and the expected reference image. The expected reference is computed from a grid map. *Scan Distance* corresponds to $j-th$ distance from the scan data. *Reference Distance* is the reference distance from a map. We denote the error as the *Matching Error*, and define as following equations:

$$ReferArea_j = ReferDist_j^2 \Delta\theta \,, \tag{1}$$

$$ScanArea_j = ScanDist_j^2 \Delta\theta \,, \tag{2}$$

$$DelArea_j = \left| ReferArea_j - ScanArea_j \right| \,, \tag{3}$$

$$MatchingError(\%) = \frac{\sum_j DelArea_j}{\sum_j (ReferArea_j + ScanArea_j)} \times 100(\%) \quad (4)$$

Equation (1) and (2) imply the area of free spaces. Equation (4) is a definition of the *matching error*. The proposed *matching error* is advantageous in that the error is defined by the area, rather than the individual range data.

In a conventional approach in [5], the error is independently evaluated for each range data. Therefore, if there is any partial corruption of a sensor data, the resultant localization performance is greatly affected. The resultant probability is a multiplication of individual probabilities. Therefore, corrupted data should be filtered out by using additional filtering techniques. On the other hand, the proposed *matching error* is less sensitive with respect to the partial data corruption.

Once the *matching error* is defined, a designer should decide the probability update rule, which is represented by $p(o_t | s_{i,t}, m) \leftarrow matching\ error$. One simple example is to use the reciprocal of the error function as in [1]. However, it is difficult to estimate resultant performances by simply using reciprocals.

Our suggestion is to use the saturated Gaussian for probability update. By monitoring variances, a designer could have quantitative insights into the convergence properties. Furthermore, excessive convergence can be prevented by using saturation. Since there are intrinsic errors in a grid map and sensor measurement, too much iterative convergence computations are not desirable. Excessive iteration might shrink pose distribution too much, and which disturbs uncertainty handling. Furthermore, the convergence region can be easily found out by monitoring a variance. The designed probability updating rule is shown as following equations:

$$\eta = 1 \Big/ \int_0^{\mu_{i,t}^s + 3\sigma_{i,t}^s} \frac{1}{\sqrt{2\pi\sigma_{i,t}^{s\,2}}} \exp\left(-\frac{(x - bias)^2}{2\sigma_{i,t}^{s\,2}} \right) dx \quad (5)$$

$$p(x) = \frac{\eta}{\sqrt{2\pi\sigma_{i,t}^{s\,2}}} \quad ; x \le bias \quad (6)$$

$$p(x) = \frac{\eta}{\sqrt{2\pi\sigma_{i,t}^{s\,2}}} \exp\left(-\frac{(x - bias)^2}{2\sigma_{i,t}^{s\,2}} \right) \quad (7)$$

$; bias < x \le \mu_{i,t}^s + 3\sigma$

$$p(x) = 0 \ ; \ x > \mu_{i,t}^s + 3\sigma \quad (8)$$

Equation (6) implies a saturation region. It means that the probability is saturated if the matching error is smaller than the bias. Equation (8) indicates that the probability is 0 if the matching error is larger than $\mu + 3\sigma$, which is the convergence region limit. Equation (7) is a Gaussian probability mapping rule.

C. Localizer State Estimation

The objective of estimation is to decide whether the current localization status is "*localizer success*" or "*localizer warning*." We use *matching error* to estimate the localizer failure since the *matching error* represents a pose error that is computed based on a robot pose. The idea is that the *matching error* is larger than a specific threshold when localization is failed. The major difficulty lies in the determination of the threshold. The threshold can not be a single constant. The *matching error* is non-zero due to various errors even for the case of successful localization. The *matching error* varies a lot according to the environment. Therefore, it is difficult to estimate localizer status by the absolute value of the *matching error*.

The *matching error* threshold value should be changed according to the environmental changes, the characteristic of the sensor and dynamic obstacles. In this paper, the threshold is computed by the *matching error* history and the *threshold* is the upper limit of the *matching error* of *localization success*.

$$threshold = \mu_{i-n:i} + 3\sigma_{i-n:i} + bias \quad (9)$$

Equation (9) shows how to compute the threshold. $\mu_{i-n:i}$ is the mean value of the matching error from $i-n$ to $i-th$ step. $\sigma_{i-n:i}$ is the standard deviation of the matching error.

The *matching error* can exceed the *threshold* by dynamic obstacles or a partial failure of the sensor. In many cases, the conventional MCL is able to deal with instantaneous failures. Hence, we use a median filter to eliminate the short-term outlier values. A median filter removes the outliers by replacing a data with the median value of the *n* surrounding points (here; *n* = 15). If the computed value of the *matching error* by the median filter is larger than *threshold*; the localization status is switched into "*localizer warning.*"

III. THE SEMI-GLOBAL LOCALIZATION

We propose the *semi-global localization* scheme to deal with exceptional localization situation. Since we have a localization status history of past movement, we can find out the moment when the robot fall into the *localization warning*. Therefore, we can specify the reachable region of the robot. The *semi-global localization* implies that the sample distribution can be limited to robot's reachable region. This approach is extremely useful when there is abrupt wheel slip. We noticed that there are a lot of irregular ground conditions in indoor environment. In such a case, excessive wheel slip takes place, which cannot be compensated by conventional MCL technique.

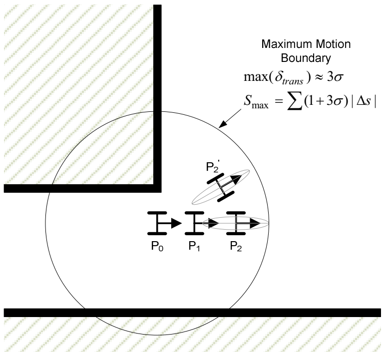


Fig. 3. *Maximum Motion Boundary Model*(P_0: last localization success pose, P_1: slip position, P_2: predicted robot pose by odometry, P_2': robot pose by wheel slippage)

Fig. 3 shows a *maximum motion boundary* region as a circle when the localizer is failed to estimate a robot pose. Suppose that a robot starts navigation from P_0 and a localizer status is *localization success*. While the robot moves to P_2, the robot's wheel slipped at P_1. As a result of wheel slippage, a robot is located at P_2'. P_2 is a pose that calculated by a wheel encoder. In that case, a maximum reachable region is a circle represented in fig. 3.

$$S_{\max} = \sum_i (1+3\sigma)\,|\Delta s| \tag{10}$$

Equation (10) shows how to calculate $S_{\max}$ that means maximum moving distance calculated using the encoder. σ is the standard deviation of the motion model, Δs is monitored by encoders. Using a motion model that has the Gaussian probability distribution, we assume maximum moving distance by abrupt wheel slippage as $\max(\delta_{trans}) \approx 3\sigma$.

IV. EXPERIMENTS

A. Experimental Environments

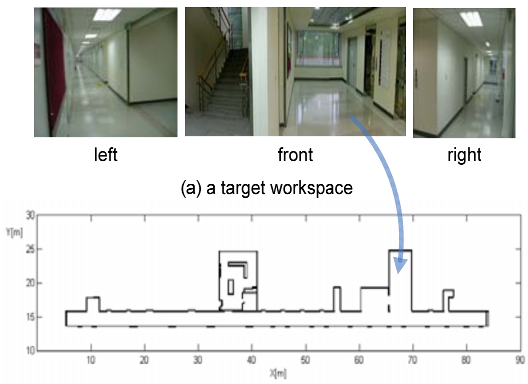


Fig. 4. Experimental environments

The proposed sensor model and *semi-global localization* experiments are conducted in a large building which is shown in fig.4. Fig.4 (a) shows a target workspace. The environment contains some difficulties. There exists a mirror-like surface around the elevator hall, which is about $x = 64m \sim 70m$. There also exists a stair, which is an open space. Fig.4 (b) shows the environment as a grid map with grid size $10cm \times 10cm$.

Fig. 5. Infortainment robot

Fig.5 shows a two wheel differential drive mobile robot. The robot is equipped with two SICK laser scanners in front and rear. The laser range finder scans 180° with 1° angular resolution and the maximum range was set to 9m.

B. Sensor model

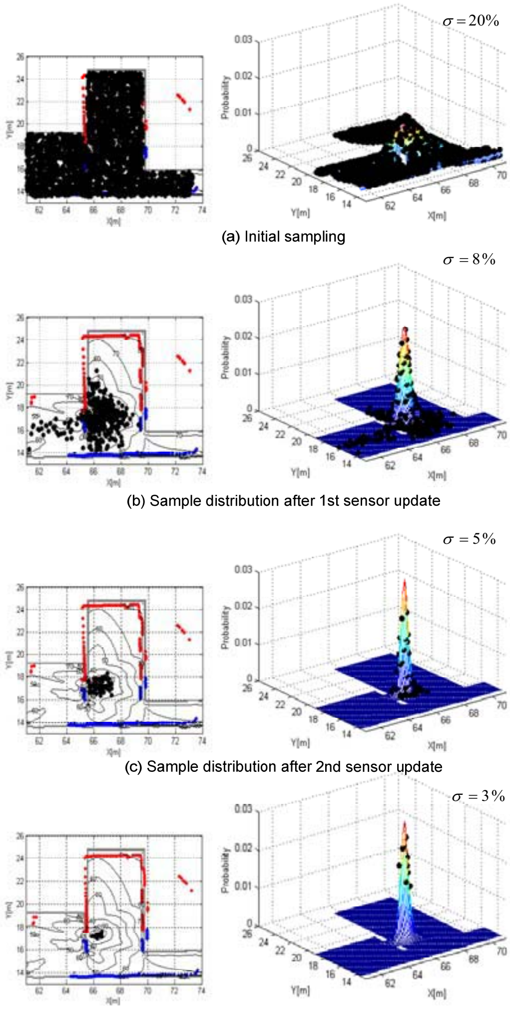
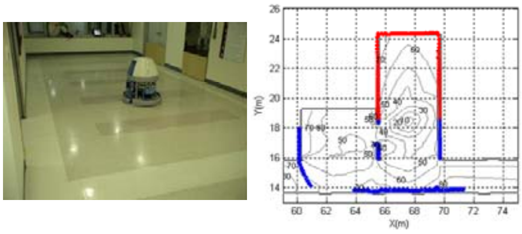
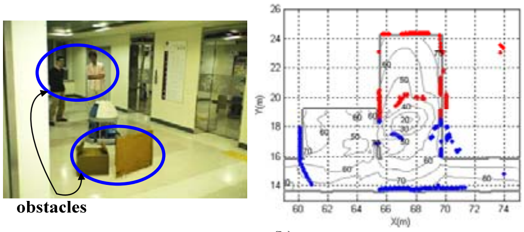


Fig. 6. Convergence of samples by MCL

Fig. 6 shows a localization result using the proposed *matching error*. Fig. 6 shows *matching error* contour in left side and probability distribution in right side which is changed by sample's *matching error* distribution. Fig. 6 (a) shows first sampling result that samples are randomly distributed. After 2 steps of sensor update, samples are converged to a region that *matching error* is smaller than 60%. After 3 steps of sensor update, samples converge to 10% region. Conducted experiment shows that samples are converged to a robot pose while sensor data is updated.



(a)

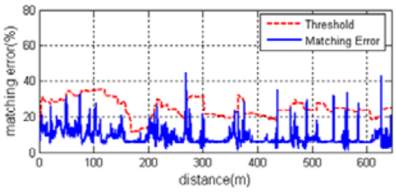


(b)

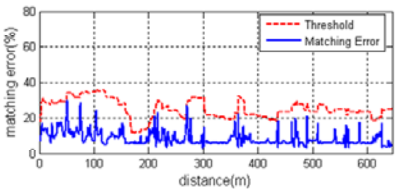
Fig. 7. *Matching error* contour by corruption of sensor data by obstacles. (a) *Corruption ratio* 0% (b) *Corruption ratio* 30%

Fig.7 shows the *matching error* contour by different corruptions by obstacles. The *corruption ratio* means the percent of corrupted sensor reading that a range error is greater than 1m. As shown in fig.7, even obstacles block the sensor; it is clear that the contour shape does not change a lot. This fact implies that the convergence property is maintained under the existence of dynamic obstacles.

C. Localizer state estimation



(a)



(b)

Fig. 8. *Matching error* during navigation (a) without median filter, (b) with median filter

Fig. 8 (a) shows measured *matching error* while a robot navigates through a corridor about 670m. A navigation experiment is conducted in the environment that was shown in fig. 4 and the localizer was in success at all times. It means that localization status was *localizer success* during the experiment. Even the status of localizer is *localizer success*, the *matching error* is non-zero due to various errors. Fig. 8 (a) shows the *matching error* exceeds the *threshold*. This fact implies that the localizer estimation result might return "*localization warning*," which is wrong estimation, without a median filter. Fig. 8 (b) shows *matching error* with median filter. As shown in fig. 8, using median filter, some outlier values are filtered out. Therefore, the result of estimation was always success, which matches with the experimental result.

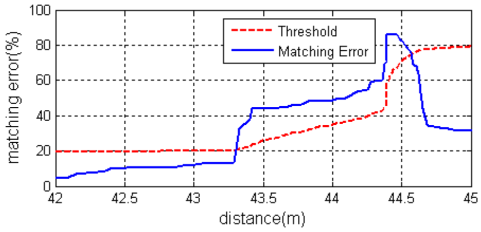


Fig. 9. *Matching error* when localization is failed

Fig.9 shows the localization failure case by the abrupt wheel slippage at 43.3m, and the *matching error* exceeds the threshold. The localization status is switched into localizer warning. Please notice that the *matching error* clearly exceeds the threshold for the case of serious failure.

D. Semi-Global localization

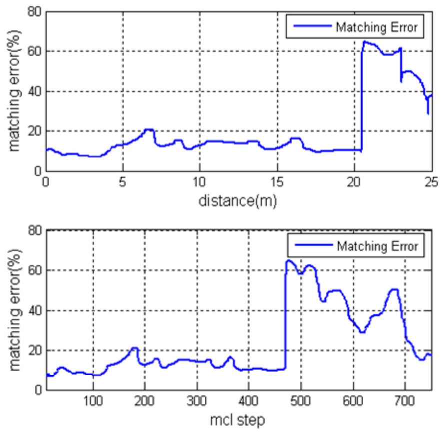


Fig. 10. *Matching error* by conventional MCL

Fig. 10 shows *matching error* during navigation using the conventional MCL. The localizer is completely failed by abrupt wheel slippage at 21m. As shown in Fig.10 the *matching error* cannot be recovered from the localization failure.

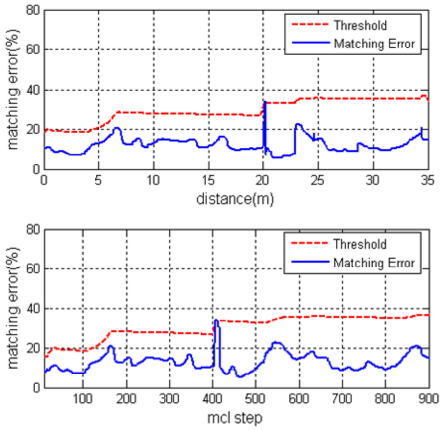


Fig. 11. *Matching error* by the proposed *semi-global localization*

Fig. 11 shows different result with fig. 10. At 20m the *matching error* exceeds the threshold by abrupt wheel slippage; therefore the localization status is switched into localizer warning. Hence, a robot tries to recover from localizer warning using the *semi-global localization* algorithm. As shown in Fig.11, the *matching error* is recovered after 21m.

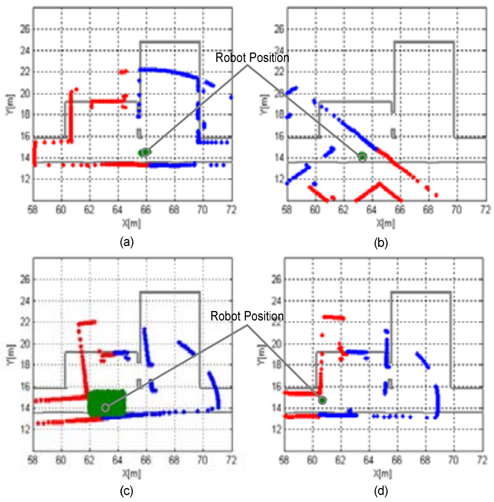


Fig. 12. *Semi-global localization* steps

Fig.12 shows *semi-global localization* steps by showing the grid map and scan data at 20m. Fig.12 (c) shows that the localizer starts the *semi-global localization* since the *matching error* exceeds the *threshold* by the wheel slippage. As shown in fig.12 (c), the sampling region is spread to the reachable region about 2.64m^2. As shown in fig.12 (d), the localizer recovers the localization failure after 10 sampling steps.

The conducted experiment shows that a robot can recover from localization failure by using the proposed scheme. In fig. 12 (c), the sampled area is 2.64m^2 while the total workspace is 290m^2. Hence, the computing region using the global localization to estimate a robot pose is 100 times larger than the region of *semi-global localization*. It is clear that the *semi-global localization* algorithm is faster and accurate than the conventional global localization.

V. CONCLUSION

In this paper, three practical issues of the reliable localization scheme are addressed, and then our solutions were presented. By using the proposed sensor model design, a performance evaluation strategy and the *semi-global localization*, a practical and reliable localization solution is developed. The presented approaches were successfully experimented in real environment. It is noteworthy that recognition of the localization status greatly helps to improve practical performances, even though the presented *matching error* concept is quite simple.

REFERENCES

[1] Dongheui Lee, Woojin Chung, "Discrete Status based Localization for Indoor Service Robots," *IEEE Transactions on Industrial Electronics*, Vol. 53, No.5, October 2006. (to appear)

[2] Dongheui Lee, Woojin Chung, "Dependable Localization Strategy in Dynamic Real Environments," *IEEE/RSJ International Conf. on Intelligent Robotics and System (IROS)*, 2005.

[3] Kai O. Arras, Jos. A. Castellanos, Roland Siegwart, "Feature-based Multi-Hypothesis Localization and Tracking for Mobile Robots Using Geometric Constraints," *Proceedings of the 2002 IEEE International Conference On Robotics & Automation*, Washington DC, May 2002.

[4] S. Thrun, D. Fox, W. Burgard, *Probabilistic Robotics*, The MIT Press, 2005, pp.153-169.

[5] Arnaud Doucet, Nando de Freitas, and Neil Gordon, *Sequential Monte Carlo Methods in Practice*, New York: Springer-Verlag, January 2001, pp.10-13.

[6] F. Dellaert, D. Fox, W.Burgard, S. Thrun, "Monte Carlo Localization for Mobile Robots," *IEEE Int. Conf. on robotics and Automation (ICRA)*, 1999

[7] J.-S. Gutmann, D. Fox, "An Experimental Comparison of Localization Methods Continued," *IEEE/RSJ International Conf. on Intelligent Robots and Systems (IROS)*, 2002.