PR-SLAM in Particle Filter Framework*

Gijeong Jang, Jun-Sik Kim, Sungho Kim and Inso Kweon

Department of Electrical Engineering & Computer Science
Korea Advanced Institute of Science and Technology (KAIST)
373-1, Guseong-dong, Yuseong-gu, Daejeon, Korea

gjjang@rcv.kaist.ac.kr, jskim@rcv.kaist.ac.kr, shkim@rcv.kaist.ac.kr, iskweon@kaist.ac.kr

Abstract **- Simultaneous Localization and Mapping is an important task for autonomous mobile robot. To let the robot explore a new environment without any prior map, real-time estimation of the geometrical relation between the robot and the environment is necessary.**

Extended Kalman Filter (EKF)-based approaches are the most common. However, they always have the risk of collapse where the assumption of Gaussian distribution is not applicable. It is well known that state estimation with a particle filter is very robust against clutter in dynamic and noisy environments because of its ability to represent non-Gaussian distributions. Unfortunately, particle-based posterior representation in high dimensions is extremely expensive.

We propose an approach, named Partitioned Recursive SLAM, that overcomes the complexity problem arising in adopting a particle filter in SLAM. By partitioning the state and alternating the turns for the state update, the computational capacity required to process SLAM is reduced to scale linearly with the number of landmarks in the map.

Index Terms - PR-SLAM, particle filter, Catadioptric vision, ORP

I. INTRODUCTION

SLAM is a chicken-and-egg problem. As the robot moves, its pose estimates are corrupted by error in the robot's motion. The perceived locations of objects in the environment are corrupted by both measurement noise and error in the estimated pose of the robot. In other words, because the robot's path and the map are correlated, error in one part propagates to the other alternately.

One approach to SLAM is to estimate the most likely robot pose and map using batch estimation algorithm similar to those used in the Structure From Motion (SFM) literature [1]. Another approach is solving the equations generated with the trigonometry. But these methods are not appropriate for online operation where the SLAM is conducted on a set of observations and controls that grow without bound. Furthermore, these algorithms generally do not estimate the uncertainty.

Thus, majority of SLAM method adopt probabilistic algorithms for online operation. Recently, Extended Kalman Filter (EKF) based approaches are most common, thanks to the compact state representation and relatively good performance [2][3][4][5][6][7]. An EKF based SLAM algorithm was proposed that alternately estimated the vehicle state and the depth of the environment [8]. But, EKF based methods always have the risk of collapse where the assumption of Gaussian distribution is not applicable.

It is well known that the state estimation with particle filter is very robust to clutter in dynamic and noisy environment. The estimation recovery from the false belief is most prominent characteristic of particle filter, thanks to the ability to represent non-Gaussian distribution. But, because the state dimension drastically increase in proportion to the map size, direct particle representation of density in very high dimension is almost impossible [9]. Recently, modified versions of particle filter in SLAM are introduced and show partial solutions for the problem [10].

In this paper, we introduce an efficient SLAM algorithm structured in particle filter. In the particle filter framework, all the distributions are represented by the sample densities. The algorithm runs in the form that the state is partitioned and they alternate the turns of update between vehicle and landmarks. We name it as Partitioned Recursive SLAM (PR-SLAM)

II. CATADIOPTRIC VISION SYSTEM

In the two-dimensional bearing-only SLAM, the catadioptric vision sensor (omnidirectional camera) is used as an input device with a very wide field of view [11]. Another advantage of this camera is that it provides direct reports on horizontal bearing measurements. These facts simplify the problem of bearing-only SLAM.

As shown in Fig. 1, if the surface of the mirror is formed by revolving a hyperbola around the Z axis, all the rays directed to the focal point of the mirror reflect on the mirror surface and turn to the principal point of the lens. Consequently, if the robot moves on a flat plane, the patterns on the height Z invariably appear on the same radius in the omnidirectional image, as shown in Fig. 2. The black circle is the horizontal line that is the projection of the horizontal plane in the omnidirectional image. As the robot begins to move and takes an image sequence, the points on the horizontal plane move only along the horizontal line.

* This research has been supported by NRL (code number M1-0302-00-0064) of MOST, Korea.

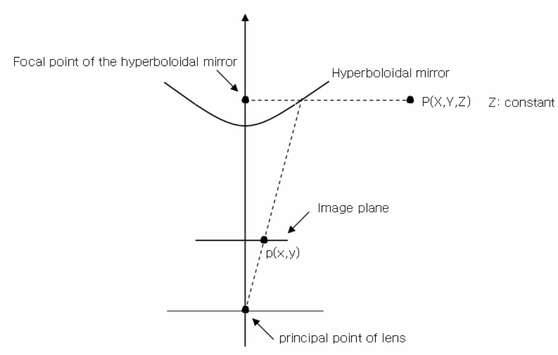


Fig. 1 The configuration of the omnidirectional camera

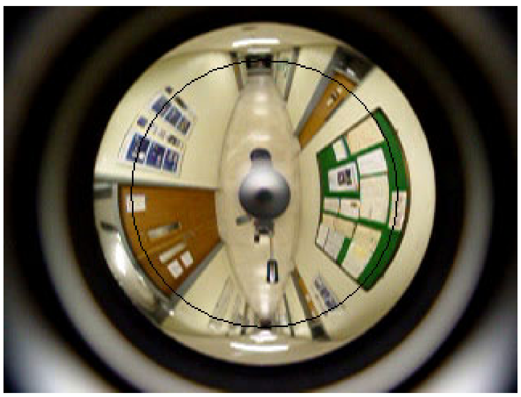


Fig. 2 An omnidirectional image and a horizontal line

Fig. 3 is a special case of an omnidirectional spatiotemporal image, an Omnidirectional Route Panorama (ORP), which is the sequential stacking of horizontal lines taken by the robot moving on the plane [12].

A simple real-time edge operation with nonmaxima suppression [13] on ORP makes feature point tracking extremely easy. This means that the data association problem can be solved simply by tracing the edge of ORP. In addition, the representation of ORP can save memory for scene description.

We can obtain the azimuth angle directly because of the proportionality between the azimuth of the feature direction and the feature location in the horizontal line.

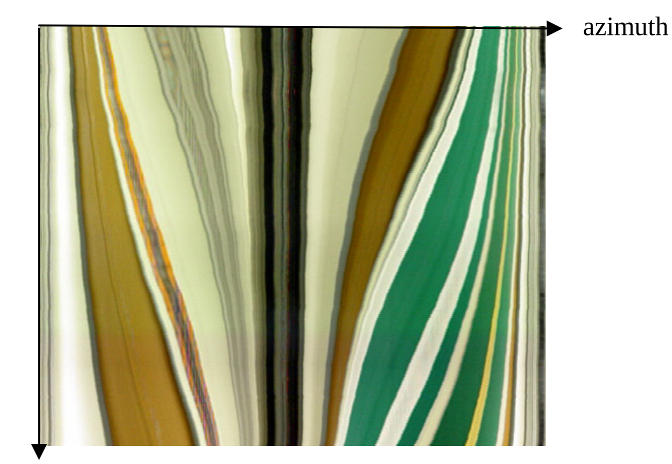


Fig. 3 Omnidirectional Route Panorama (ORP)

III. Localization in Probabilistic Approach

Approached probabilistically, the localization problem is a density estimation problem, where a robot seeks to estimate a posterior distribution over the space of its poses coordinated on the available data. Given the map and the data accumulated by the robot, the posterior probability distribution can be written as

$$p(\mathbf{s}_t \mid Z^t, U^{t-1}, M) .$$

where $\mathbf{s}_t$ is the current pose of the robot (in 2-dimensional case, it is composed of x-y-coordinates and its heading direction ϕ). The posterior is conditioned on the set of all sensor readings $Z^t = \{\mathbf{z}_i, i = 1, \ldots, t\}$ up to the current time step, control input $U^{t-1} = \{\mathbf{u}_i, i = 1, \ldots, t-1\}$ up to the previous time step and the given map M . The posterior can be derived as the following recursive form applying Bayes rule, the theorem of total probability and exploiting the Markov assumption. For convenience sake, the belief on the robot's state at time t is denoted by $b_t(\mathbf{s}_t)$ [14].

$$\begin{aligned}
& b(\mathbf{s}_t) = (\mathbf{s}_t \mid Z^t, U^{t-1}, M) \\
& \overset{(a)}{=} p(\mathbf{z}_t \mid Z^{t-1}, U^{t-1}, \mathbf{s}_t, M)\, p(\mathbf{s}_t \mid Z^{t-1}, U^{t-1}, M) / p(Z^t \mid U^{t-1}, M) \\
& \overset{(b)}{=} \eta_t\, p(\mathbf{z}_t \mid Z^{t-1}, U^{t-1}, \mathbf{s}_t, M)\, p(\mathbf{s}_t \mid Z^{t-1}, U^{t-1}, M) \\
& \overset{(c)}{=} \eta_t\, p(\mathbf{z}_t \mid \mathbf{s}_t, M)\, p(\mathbf{s}_t \mid Z^{t-1}, U^{t-1}, M) \\
& \overset{(d)}{=} \eta_t\, p(\mathbf{z}_t \mid \mathbf{s}_t, M) \int p(\mathbf{s}_t \mid \mathbf{s}_{t-1}, Z^{t-1}, U^{t-1}, M)\, p(\mathbf{s}_{t-1} \mid Z^{t-1}, U^{t-1}, M)\, d\mathbf{s}_{t-1} \\
& \overset{(e)}{=} \eta_t\, p(\mathbf{z}_t \mid \mathbf{s}_t, M) \int p(\mathbf{s}_t \mid \mathbf{s}_{t-1}, \mathbf{u}_{t-1}, M)\, p(\mathbf{s}_{t-1} \mid Z^{t-1}, U^{t-2}, M)\, d\mathbf{s}_{t-1} \\
& \overset{(e)}{=} \eta_t\, p(\mathbf{z}_t \mid \mathbf{s}_t, M) \int p(s_t \mid \mathbf{s}_{t-1}, \mathbf{u}_{t-1}, M)\, b(\mathbf{s}_{t-1})\, d\mathbf{s}_{t-1}
\end{aligned} \tag{3}$$

where η_t is a constant normalizer which ensures that the result sums up to 1.

The derivation follows

(a) Bayes rule.

(b) Z^t does not depend on U^{t-1} and M in the absence of other data.

(c) The Markov assumption that specifies the conditional independence of future from past data given the knowledge of current location and the map.

(d) The theorem of total probability.

(e) Markov assumption.

It is simplified with the recursive form as

$$b(\mathbf{s}_t) = \eta_t\, p(\mathbf{z}_t \mid \mathbf{s}_t, M) \int p(\mathbf{s}_t \mid \mathbf{s}_{t-1}, \mathbf{u}_{t-1}, M)\, b(\mathbf{s}_{t-1})\, d\mathbf{s}_{t-1} . \tag{4}$$

This equation is general probabilistic form of recursive state update model combining motion update and measurement update.

IV. PR-SLAM IN PROBABILISTIC APPROACH

Given the sensor reading accumulated by the robot, the most popular online solutions to SLAM are direct estimations of a posterior probability distribution over the space of all possible maps and all possible robot poses. The SLAM posterior is written as

$$p(\mathbf{s}_t, M | Z^t, U^{t-1}) .$$

Where $M = \{\mathbf{m}_k ; k = 1,2,\dots,N\}$. $\mathbf{m}_k$ is a single landmark, M is a set of random vectors for all landmarks, and N is the number of the landmark.

Generally, SLAM is implemented by combining the state to be estimated as follows.

$$\mathbf{x} = (\mathbf{s}^T, \mathbf{m}_1^T, \mathbf{m}_2^T, \dots, \mathbf{m}_n^T)^T$$

The posterior is then derived from the following form combining motion update and measurement update.

$$\begin{aligned} & p(\mathbf{x}_t | Z^t, U^{t-1}) \\ & = \eta_t\, p(\mathbf{z}_t | \mathbf{x}_t) \int p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_{t-1})\, p(\mathbf{x}_{t-1} | Z^{t-1}, U^{t-2})\, d\mathbf{x}_{t-1} \end{aligned} \tag{5}$$

Almost all the SLAM algorithms adopt this direct state estimation strategy based on EKF. Although the particle filter has a powerful advantage in representing arbitrary density distribution, satisfactory particle representation in such a high dimension is almost impossible. Therefore, we divide the state by modifying Equation (5), making it possible to represent the states by particles. The posterior of vehicle and each landmark is derived as an alternating recursive form by applying Bayes' rule, the theorem of total probability, and exploiting the Markov assumption. The dynamics model of the map can be ignored because of the static world assumption.

The process sequence and the derivation are as follows.

Process: $\mathbf{u}_{t-1} \rightarrow b(\mathbf{s}_{t-1}) \rightarrow b(M_{t-1}) \rightarrow \mathbf{u}_t \rightarrow b(\mathbf{s}_t) \rightarrow b(M_t)$

$$\begin{aligned} b(\mathbf{s}_t) &= p(\mathbf{s}_t | Z^t, U^{t-1}) \\ &= \int p(\mathbf{s}_t | M_{t-1}, Z^t, U^{t-1})\, p(M_{t-1} | Z^t, U^{t-1})\, dM_{t-1} \\ &= \int \{ p(\mathbf{z}_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t, M_{t-1})\, p(\mathbf{s}_t | Z^{t-1}, U^{t-1}, M_{t-1}) / p(\mathbf{z}_t | Z^{t-1}, U^{t-1}, M_{t-1}) \}\, p(M_{t-1} | Z^{t-1}, U^{t-2})\, dM_{t-1} \\ &= \int \eta_t\, p(\mathbf{z}_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t, M_{t-1})\, p(\mathbf{s}_t | Z^{t-1}, U^{t-1}, M_{t-1})\, p(M_{t-1} | Z^{t-1}, U^{t-2})\, dM_{t-1} \\ &= \eta_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_{t-1})\, p(\mathbf{s}_t | Z^{t-1}, U^{t-1}, M_{t-1})\, p(M_{t-1} | Z^{t-1}, U^{t-2})\, dM_{t-1} \\ &= \eta_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_{t-1}) \{ \int p(\mathbf{s}_t | \mathbf{s}_{t-1}, Z^{t-1}, U^{t-1}, M_{t-1})\, p(\mathbf{s}_{t-1} | Z^{t-1}, U^{t-1}, M_{t-1})\, d\mathbf{s}_{t-1} \}\, p(M_{t-1} | Z^{t-1}, U^{t-2})\, dM_{t-1} \\ &= \eta_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_{t-1}) \{ \int p(\mathbf{s}_t | \mathbf{s}_{t-1}, U_{t-1}, M_{t-1})\, p(\mathbf{s}_{t-1} | Z^{t-1}, U^{t-2})\, d\mathbf{s}_{t-1} \}\, p(M_{t-1} | Z^{t-1}, U^{t-2})\, dM_{t-1} \\ &= \eta_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_{t-1}) \{ \int p(\mathbf{s}_t | \mathbf{s}_{t-1}, \mathbf{u}_{t-1}, M_{t-1})\, b(\mathbf{s}_{t-1})\, d\mathbf{s}_{t-1} \}\, b(M_{t-1})\, dM_{t-1} \end{aligned} \tag{6}$$

$$\begin{aligned} b(M_t) &= p(M_t | Z^t, U^{t-1}) \\ &= \int p(M_t | \mathbf{s}_t, Z^t, U^{t-1})\, p(\mathbf{s}_t | Z^t, U^{t-1})\, d\mathbf{s}_t \\ &= \int \{ p(\mathbf{z}_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t, M_t)\, p(M_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t) / p(Z_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t) \}\, p(\mathbf{s}_t | Z^t, U^{t-1})\, d\mathbf{s}_t \\ &= \int \eta'_t\, p(\mathbf{z}_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t, M_t)\, p(M_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t)\, p(\mathbf{s}_t | Z^t, U^{t-1})\, d\mathbf{s}_t \\ &= \eta'_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_t)\, p(M_t | Z^{t-1}, U^{t-1}, \mathbf{s}_t)\, p(\mathbf{s}_t | Z^t, U^{t-1})\, d\mathbf{s}_t \\ &= \eta'_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_t) \{ \int p(M_t | M_{t-1}, Z^{t-1}, U^{t-1}, \mathbf{s}_t)\, p(M_{t-1} | Z^{t-1}, U^{t-1}, \mathbf{s}_t)\, dM_{t-1} \}\, p(\mathbf{s}_t | Z^t, U^{t-1})\, d\mathbf{s}_t \\ &= \eta'_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_t) \{ \int p(M_t | M_{t-1})\, p(M_{t-1} | Z^{t-1}, U^{t-2})\, dM_{t-1} \}\, p(\mathbf{s}_t | Z^t, U^{t-1})\, d\mathbf{s}_t \\ &= \eta'_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_t) \{ \int p(M_t | M_{t-1})\, b(M_{t-1})\, dM_{t-1} \}\, b(\mathbf{s}_t)\, d\mathbf{s}_t \end{aligned} \tag{7}$$

where $(M_t | M_{t-1})$ is static.

These complex equations are simplified with the alternating recursive form as follows.

$$b(\mathbf{s}_t) = p(\mathbf{s}_t | Z^t, U^{t-1}) = \eta_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_{t-1}) \{ \int p(\mathbf{s}_t | \mathbf{s}_{t-1}, \mathbf{u}_{t-1}, M_{t-1})\, b(\mathbf{s}_{t-1})\, d\mathbf{s}_{t-1} \}\, b(M_{t-1})\, dM_{t-1} \tag{8}$$

$$b(M_t) = p(M_t | Z^t, U^{t-1}) = \eta'_t \int p(\mathbf{z}_t | \mathbf{s}_t, M_t) \{ \int p(M_t | M_{t-1})\, b(M_{t-1})\, dM_{t-1} \}\, b(\mathbf{s}_t)\, d\mathbf{s}_t \tag{9}$$

The vehicle pose is updated from the previous belief on the map, and the map is updated from the current belief on the robot pose from Equations (8) and (9).

To implement this, we need to specify the motion model $p(\mathbf{s}_t | \mathbf{s}_{t-1}, \mathbf{u}_{t-1}, M_{t-1})$ that characterizes the effect of the robot's actions on its pose, and the measurement model for the vehicle $p(\mathbf{z}_t | \mathbf{s}_t, M_{t-1})$ and for the landmarks $p(\mathbf{z}_t | \mathbf{s}_t, M_t)$ that characterizes the perception between vehicle and landmarks. These models are assumed to be time-invariant.

V. MOTION MODEL

The state is defined for the vehicle and each landmark.

$$\mathbf{s} = (x_v, y_v, \phi)^T$$

$$\mathbf{m}_k = (x_{m_k}, y_{m_k})^T \; ; \; k = 1,2,\ldots,n$$

The motion update for differential-drive vehicle follows the dynamics model as

$$\mathbf{s}_t = f(\mathbf{s}_{t-1}, \mathbf{u}_{t-1}, \mathbf{w}_{t-1}) \tag{10}$$

where $\mathbf{u}$ is robot control and $\mathbf{w}$ is additive motion noise assumed as Gaussian.

$$\mathbf{u} = (u_1, u_2)^T \, , \; \mathbf{w} = (w_1, w_2)^T \; ; \; p(\mathbf{w}) \sim N(0, Q)$$

where Q is process noise covariance.

In the vector of robot control and additive motion noise, the first entry is for translation, and the second is for rotation.

Applying the "static world assumption" there is no motion model for each landmark. And the motion model for each entry of vehicle state is

$$x_{v,t} = x_{v,t-1} + \cos(\phi_{t-1} + (u_{2,t} + w_{2,t})/2) \cdot (u_{1,t} + w_{1,t}) \tag{11}$$

$$y_{v,t} = y_{v,t-1} + \sin(\phi_{t-1} + (u_{2,t} + w_{2,t})/2) \cdot (u_{1,t} + w_{1,t}) \tag{12}$$

$$\phi_t = \phi_{t-1} + (u_{2,t} + w_{2,t}) \tag{13}$$

Fig. 4 shows the posterior uncertainty generated from the motion model. Blue line illustrates a commanded robot path, and the shaded cloud illustrates the posterior distribution of the robot's pose.

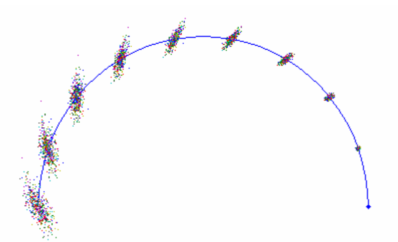


Fig. 4 The posterior distribution with the probabilistic motion model

VI. MEASUREMENT MODEL

The measurement vector is defined as

$$\mathbf{z} = (\theta_1, \theta_2, \ldots, \theta_N)^T$$

which means all observable bearings on the landmarks.

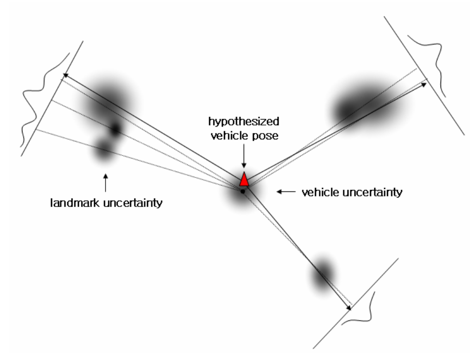


Fig. 5 The measurement model for the vehicle in PR-SLAM

As shown in Fig. 5, the measurement model for the vehicle is defined as follows.

$$p(\mathbf{z}_t \mid \mathbf{s}_t, M_{t-1}) = \prod_{j=1}^{j=n} p(z_{j,t} \mid \mathbf{s}_t, m_{j,t-1}) \tag{14}$$

The measurement procedure for a single vehicle sample (hypothesis) is as follows.

1. Obtain angle measurements for each landmark.
2. Project the rays to the direction of the measurement angles from the hypothesized vehicle state after the motion drift.
3. Find the marginal densities for each direction of measurements.
4. Set the measurement value as the product of measurements for each landmark.
5. Repeat Steps 2 to 4 for all the hypothesized vehicle poses.

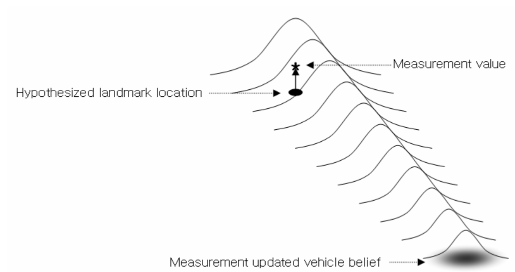


Fig. 6 The measurement model for the landmark in PR-SLAM

The measurement model for each landmark is

$$p(z_{j,t} \mid \mathbf{s}_t, m_{j,t}) \, .$$

The measurement procedure for a single landmark sample (hypothesis) is as follows.

1. Generate measurement density for each landmark from the measurement updated vehicle belief. For example, this can be the superposition of measurement rays from the vehicle belief (Fig. 6).
2. Set the measurement value as the measurement density at the hypothesized landmark location.
3. Repeat Step 2 for all the hypothesized landmark positions.
4. Continue Step 3 for all landmark entries.

VII. Experiments

We tested the PR-SLAM algorithm in indoor environments with a differential-drive mobile platform of Pioneer II (Fig. 7). It carries a laptop computer with a Pentium IV processor clocked at 1.6 GHz and an omnidirectional color camera capturing 30 frames/sec with the resolution of 640×480 pixels. The experiment was conducted in real time.

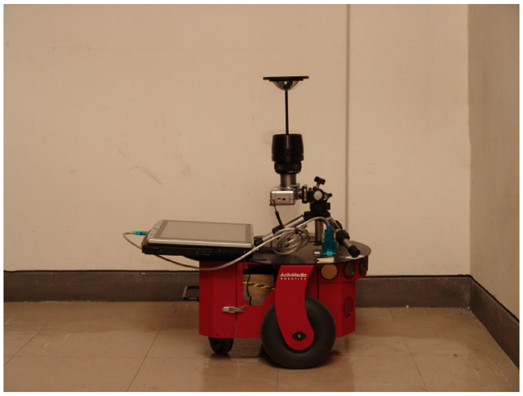


Fig. 7 System configuration

Before conducting the experiments, we tested PR-SLAM with a simulation in two different situations. First, we assumed very inaccurate odometer which reports only moving distance with 10% error (it means the standard deviation of added Gaussian noise is 10% of the real value) and doesn't report rotation angle. Second, we conducted a simulation without odometry. Also, the measurement noise was added in both of the cases. The std. of the Gaussian noise was 0.3° in angle. It corresponds to 1 pixel in the given image resolution. The sample number used for the representation of the density was 1000 for vehicle and each landmark. In both of the cases, we could get successful results.

Fig. 8 shows the results of simulation conducted with an odometer of poor accuracy. The real path of the vehicle is denoted by black circles and the current position is denoted by a small red circle. The uncertainties of the states are represented by the cloud of dots. The red circle with a direction indicator is the mean pose of the vehicle state estimate, and the green circles are the means of the landmark state estimates. Initially, the samples both for the vehicle and landmarks are distributed uniformly in the state space. With the PR-SLAM algorithm, the robot and landmarks find the correct state according to the visual measurements. After the first iteration, the hypothesis of landmark position aggregates along the line of sight according to the measured direction of the landmark. As the robot moves on, all the sample hypotheses were examined and updated iteratively. The vehicle state was examined with the landmark beliefs of previous step and the landmarks were examined with the updated vehicle posterior density. The landmarks previously positioned with its small uncertainty play an important role in localization of the vehicle. With the corrected vehicle pose, the landmarks were continuously stitched together along the landmarks previously determined in position. With some observation on the experiments, we have come to know that the uncertainty of the landmark is the intersection of the previous uncertainty and the current observation.

Fig. 9 is the results of simulation conducted without odometry after step 10. For the first 10 step, we let the vehicle move according to odometer reading for the purpose of initialization. Even with the bearing measurement only, the results are satisfactory.

Fig. 10 is the simulation results tested in the situation of under-constrained condition. In chapter I, we mentioned the minimum geometrical constraints for the localization. Only two bearing measurements inform that the candidate location is on the circle passing through the vehicle and the two landmarks. In the experiment, we can observe the uncertainty expands on the circle.

Fig. 11 shows the experimental results for the proposed SLAM from the real indoor environment. We did not use the odometry information in the experiment. The real map of the corridor was overlaid to check the estimation accuracy. The location of the doors and signboards were accurately estimated.

The processing time is linearly proportional to the sample number and landmark number.
The average processing time for a single iteration is

$$\text{The time consumed for a single iteration} \\ = 0.043 * no.of\ landmark + 0.051\,(\text{sec}) \quad (15)$$

More efficient representation for the observation density will enhance the computational speed of the algorithm by far.

VIII. Conclusion

We proposed a novel SLAM algorithm named as PR-SLAM. By partitioning the state and alternating the turns for state update, we overcame the complexity problem that arises in adopting particle filter in SLAM.

Taking the advantageous characteristic of particle filter such as the robustness to clutter in dynamic and noisy environment, PR-SLAM reduced the computational load to scale linearly with the number of landmarks in the map.

The practicality of the proposed algorithm was demonstrated by conducting the computer simulation in various situations. In addition, experiments in real indoor environment were conducted.

Work is currently underway on map management. We seek to find the confidence criteria based on the consistency of the feature from which landmarks are added and deleted. In addition, we plan to apply this algorithm to a large scale environment which contains hundreds of landmarks.

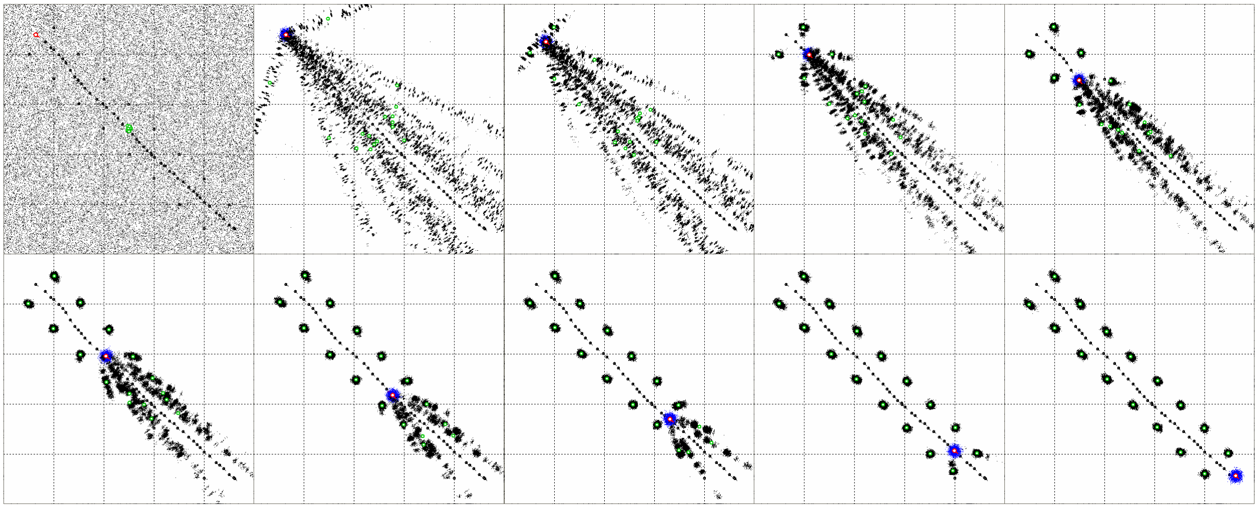


Fig. 8 The simulation results of PR-SLAM conducted with an odometer of poor accuracy.
The images are captured in frame 0, 1, 2, 5, 10, 15, 23, 31, 39, 47

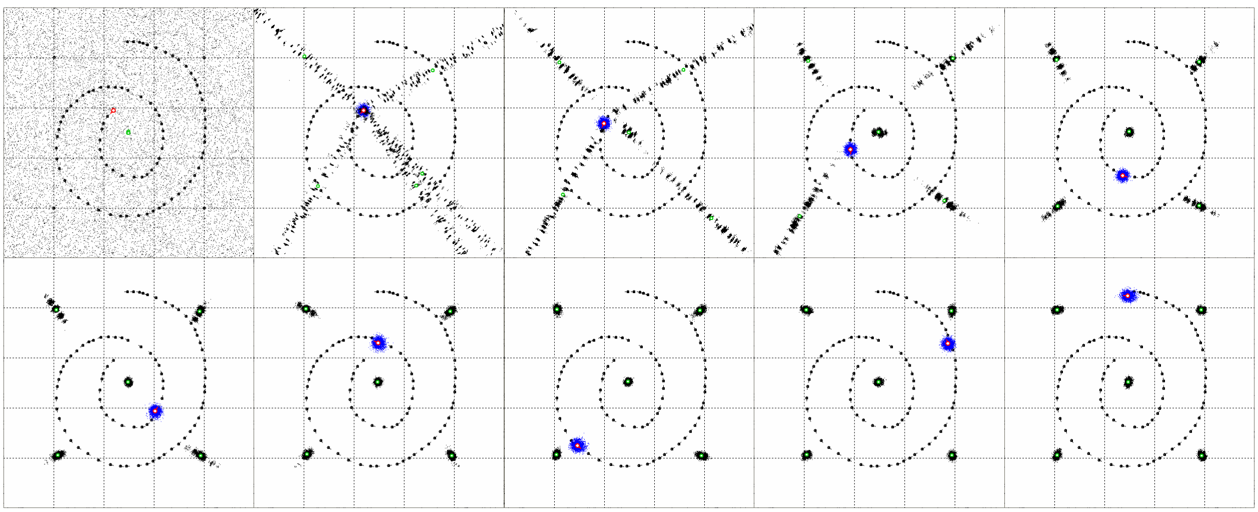


Fig. 9 The simulation results of PR-SLAM without odometry (after frame 10). The images are captured in frame 0, 1, 3, 6, 10, 15, 35, 55, 70, 84

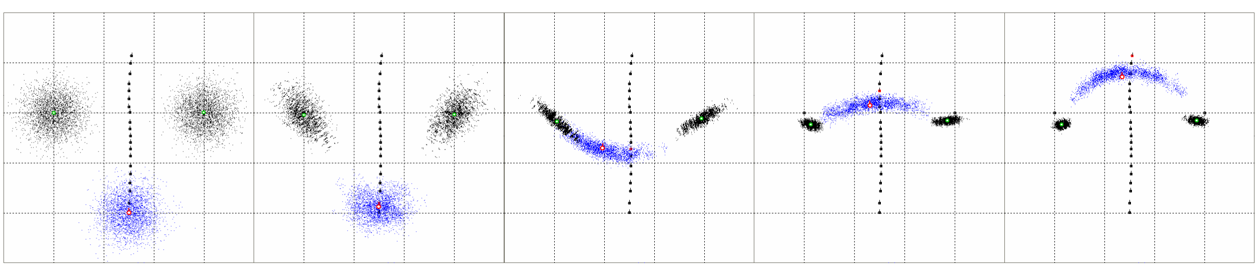
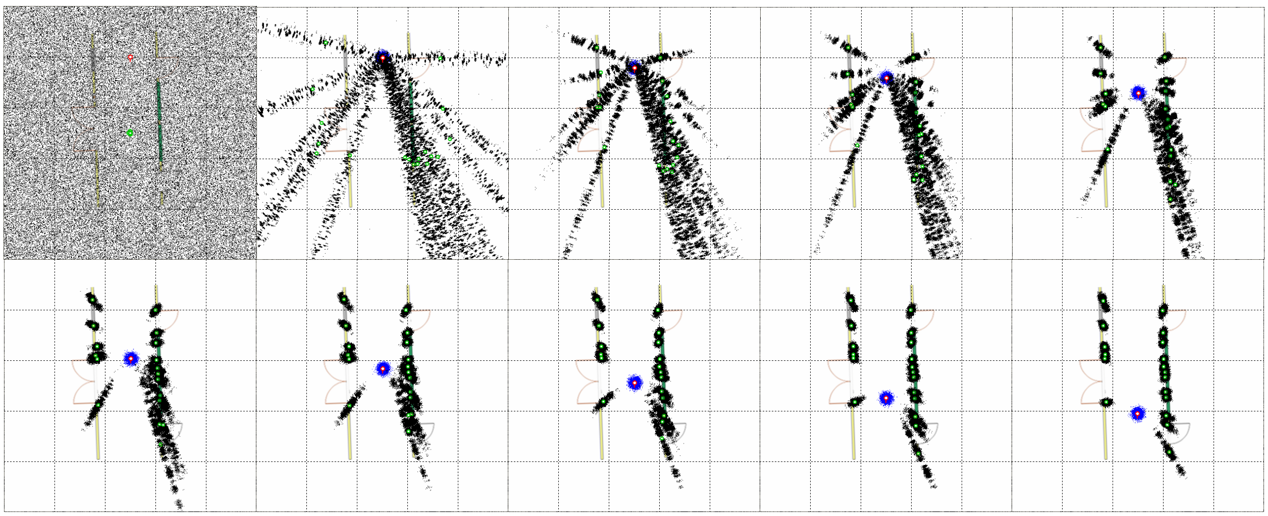


Fig. 10 The simulation results of PR-SLAM in under-constrained condition (there are only two landmarks).
The images are captured in frame 0, 2, 8, 21, 26.

(a) Input images



(b) Estimated map & vehicle position

Fig. 11 Estimation results for the vehicle and landmarks in an indoor environment.
The real map of the corridor is overlaid to check the accuracy of the estimation. The images are captured in frame 0, 1, 3, 5, 8, 11, 13, 16, 19, 22.

REFERENCES

[1] Carlo Tomasi, Takeo Kanade, "Shape and Motion from Image Streams under orthography: A Factorization Method", International Journal of Computer Vision, vol. 9, no. 2, pp 137-154, 1992.

[2] Andrew J. Davison, "Real-Time Simultaneous Localization and Mapping with a Single Camera", The 9th International Conference on Computer Vision, 2003.

[3] Andrew J. Davison, David Murray, "Simultaneous Localization and Map-Building Using Active Vision", IEEE Transaction on Pattern Analysis and Machine Intelligence, vol.24, no. 7, July 2002.

[4] M. W. M. Gamini Dissanayake, Steven Clark, Hugh F. Durrent-Whyte, M. Csorba, "A Solution to the Simultaneous Localization and Map Building (SLAM) Problem", IEEE Transactions on Robotics and Automation, vol.17, no.3, June 2001.

[5] Jose E. Guivant, Eduardo Mario Nebot, "Optimization of the Simultaneous Localization and Map-Building Algorithm for Real-Time Implementation", IEEE Transactions on Robotics and Automation, vol.17, no.3, June 2001.

[6] John J. Leonard, Hugh F. Durrant Whyte, "Mobile Robot Localization by Tracking Geometric Beacons", IEEE Transactions on Robotics and Automation, vol.7, no.3, June 1991.

[7] P. Newman, J. Leonard, J. D. Tardos, J. Neira, "Exploration and Return: Experimental Validation of Real-Time Concurrent Mapping and Localization", Proceedings of 2002 IEEE International Conference on Robotics & Automation.

[8] Larry Matthies, "Dynamic Stereo Vision", PhD Thesis, Carnegie Mellon University, October 1989.

[9] Peng Chang, Martial Hebert, "Robust Tracking and Structure from Motion with Sample Based Uncertainty Representation", Proceedings of 2002 IEEE International Conference on Robotics & Automation.

[10] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, "FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem", AAAI2002.

[11] Yasushi Yagi, Kousuke Imai, Masahiko Yachida, "Iconic Memory-based Omnidirectional Route Panorama Navigation", Proceedings of 2003 IEEE International Conference on Robotics & Automation

[12] Simon Baker and Shree K. Nayar, "A Theory of Single-Viewpoint Catadioptric Image Formation", International Journal of Computer Vision, vol. 35, no. 2, 175-196, 1999

[13] John Canny, "A Computational Approach to Edge Detection", IEEE Transaction on Pattern Analysis & Machine Intelligence, vol. 8, pp 679-698, 1986

[14] Sebastian Thrun, 'Probabilistic Algorithms in Robotics", AI Magazine, vol. 21, no. 4, pp 93-109, 2000