MOBILE ROBOT LOCALIZATION USING IMPROVED SIR FILTERS AND PARAMETRIC MODELS OF THE ENVIRONMENT

Paulo R. A. Silva and Marcelo G. S. Bruno

Instituto Tecnológico de Aeronáutica
São José dos Campos SP 12228-900, Brazil
ph:(55-12) 3947-6906; email: {pras,bruno}@ita.br

ABSTRACT

We introduce in this paper an improved particle filter for mobile robot localization using a parametric model of the environment. The proposed filter combines a clutter suppression routine for feature extraction with an optimized importance function and measurement-driven MCMC move steps. The filter is tested with both real and synthetic data and its performance is compared to competing algorithms found in the literature.

1. INTRODUCTION

A key requirement [1] in the design of any autonomous robotic system is the ability to accurately estimate a mobile robot's position and orientation with respect to a map of its environment. Jensfelt et al [2] considered this problem representing each wall in a room by a parameterized line and utilizing some form of pre-processing of raw laser-scanner measurements to extract features that were subsequently assimilated using a simplified extended Kalman filter (EKF). In this paper, we follow Jensfelt's appproach assuming the same odometric model as in [2] and a similar feature-based observation model with a few changes for technical precision. Contrary to [2] however, we replace the extended Kalman filter with an improved sampling/importance resampling (ISIR) particle filter [3] that incorporates an optimized importance function to reduce particle degeneracy and also uses a measurement-driven Metropolis-Hastings move step [4] to reduce particle impoverishment.

Due to the existence of clutter in the room, a preliminary validation algorithm is required to separate measurements of interest that come from the room's walls from those originating from other objects such as chairs, tables or cupboards. Using the geometry of the room, we derive an algorithm for the establishment of measurement validation gates around the walls and incorporate the measurement validation process into the dynamic pose estimation problem. Once the clutter is suppressed, the least-squares approach in [2] is used to extract the features of interest.

The work of the first author was supported by CAPES, Brazil.

The paper is divided into 5 sections. Section 1 is this Introduction. In Section 2, we describe briefly the odometric and observation models and discuss the measurement validation algorithm. In Section 3, we introduce the ISIR filter. In Section 4, we evaluate the performance of the filter using both simulated and real data, and compare the proposed algorithm to other benchmarks in the literature. Finally, we summarize in Section 5 the conclusions of our work.

2. THE MODEL

Let $\mathbf{x}_k = [x_k\, y_k\, \gamma_k]^T$ be an unknown random vector that collects, at instant k, the robot's pose (x_k, y_k) and orientation angle, γ_k, with respect to a fixed (non-inertial) coordinate system, henceforth referred to as the *environment system*. Let also D_k and $\Delta\gamma_k$ represent respectively the displacement between two consecutive poses of the robot and the change in the orientation angle between instants k and $k+1$ obtained from the robot's odometer. Due to random drift affecting the robot, its true state $\mathbf{x}_{k+1}$ at instant $k+1$ is better described by the nonlinear, stochastic dynamic model, see [2],

$$\underbrace{\begin{bmatrix} x_{k+1} \\ y_{k+1} \\ \gamma_{k+1} \end{bmatrix}}_{\mathbf{x_{k+1}}} = \underbrace{\begin{bmatrix} x_k + D_k cos(\frac{\Delta\gamma_k}{2} + \gamma_k) \\ y_k + D_k sin(\frac{\Delta\gamma_k}{2} + \gamma_k) \\ \gamma_k + \Delta\gamma_k \end{bmatrix}}_{\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)} + \underbrace{\begin{bmatrix} \varepsilon_k^{(x)} \\ \varepsilon_k^{(y)} \\ \varepsilon_k^{(\gamma)} \end{bmatrix}}_{\underline{\varepsilon}_k} \quad (1)$$

where, for the purposes of this paper, $\mathbf{u}_k = (D_k, \Delta\gamma_k)$ is deterministic and known, and $\{\underline{\varepsilon}_k\}$, $k \geq 0$, is a sequence of independent, identically distributed (i.i.d.) Gaussian random vectors with zero mean and covariance matrix $\mathbf{Q}$.

2.1. Observation Model

Assume that, for a rectangular room, there are N walls in the field of view of the robot's sensor at instant k, where $N = 1, \ldots, 4$. For $i = 1, \ldots, N$, the extracted features used for data assimilation at instant k are [2] the perpendicular distances, $\rho_{k,i}$, from the robot's centroid to each

detected wall, and the respective orientation angles, $\alpha_{k,i}$, $-\pi \leq \alpha_{k,i} \leq \pi$, of each detected wall line, measured with respect to the moving (inertial) coordinate system of the robot, henceforth referred to as the *robot system*. Next, let ρ_i^m, $i = 1, \ldots, N$, denote the perpendicular distances from the origin of the *environment* coordinate system to each detected wall, and let α_i^m, $i = 1, \ldots, N$, denote the corresponding orientation angles of each wall line with respect to the same fixed coordinate system. Assuming that the pairs (ρ_i^m, α_i^m) are known from the map $\mathcal{M}$ of the environment, the feature vector $\mathbf{y}_{k,i} = [\rho_{k,i}\, \alpha_{k,i}]^T$ for the ith detected wall at instant k is given then by the nonlinear model (modified from [2] to account for all possible relative positions between the robot and the walls)

$$\mathbf{y}_{k,i} = \underbrace{\begin{bmatrix} \zeta_{k,i}(\rho_i^m - \sqrt{x_k^2 + y_k^2} cos(\alpha_i^m - \beta_k)) \\ \alpha_i^m - \gamma_k + \xi_{k,i} \end{bmatrix}}_{\mathbf{h}_{k,i}(\mathbf{x}_k, \mathcal{M}_i)} + \mathbf{v}_{k,i} \quad (2)$$

where $\beta_k = \tan^{-1}(y_k/x_k)$, $\{\mathbf{v}_{k,i}\}$, $k \geq 1$, is a sequence of Gaussian random vectors with zero mean and (generally) time-varying covariance matrices $\mathbf{R}_{k,i}$, and the parameters $\zeta_{k,i}$ and $\xi_{k,i}$ are given by

$$\zeta_{k,i} = \begin{cases} 1, & || \text{ Proj}_{\rho_i^m}(x_k, y_k) \, || \leq \rho_i^m \\ -1, & || \text{ Proj}_{\rho_i^m}(x_k, y_k) \, || > \rho_i^m, \end{cases} \quad (3)$$

$$\xi_{k,i} = \begin{cases} 0, & \zeta_{k,i} = 1 \quad \text{and} \quad \alpha_i^m - \gamma_k \leq \pi \\ +\pi, & \zeta_{k,i} = -1 \quad \text{and} \quad \alpha_i^m - \gamma_k < 0 \\ -\pi, & \zeta_{k,i} = -1 \quad \text{and} \quad \alpha_i^m - \gamma_k \geq 0 \\ -2\pi, & \zeta_{k,i} = 1 \quad \text{and} \quad \alpha_i^m - \gamma_k > \pi. \end{cases} \quad (4)$$

In (3), $||$ Proj$_{\rho_i^m}(x_k, y_k)$ $||$ denotes the norm of the orthogonal projection of the point $(x_k,\, y_k)$ onto the perpendicular line connecting the origin of the environment system to the ith wall. Finally, the complete feature vector at instant k is

$$\mathbf{y}_k = \underbrace{\left[\mathbf{h}_{k,1}^T(\mathbf{x}_k, \mathcal{M}_1) \ldots \mathbf{h}_{k,N}^T(\mathbf{x}_k, \mathcal{M}_N)\right]^T}_{\mathbf{h}_k(\mathbf{x}_k, \mathcal{M})} + \mathbf{v}_k \quad (5)$$

where $\mathbf{v}_k = \left[\mathbf{v}_{k,1}^T \ldots \mathbf{v}_{k,N}^T\right]^T$. For simplicity, we assume in this paper that the feature error sequences $\{\mathbf{v}_{k,i}\}$ and $\{\mathbf{v}_{k,j}\}$, $k \geq 1$, are mutually independent and identically distributed for $i \neq j$, and also independent of $\{\underline{\varepsilon}_k\}$ and of $\mathbf{x}_0$, see equation (1).

2.2. Validation Gates and Feature Extraction

Let $\widehat{\mathbf{x}}_{k|k-1}$ be an estimate of robot's state at instant k based on the features observed from instant one up to instant $k-1$, and let $(x_{s,i},\, y_{s,i})$ and $(x_{e,i},\, y_{e,i})$ denote respectively the start and end points, in the environment system, that specify the ith wall in the known parametric model $\mathcal{M}$ of the room. The measurement validation gate for the ith detected wall is defined by the vector, see [2], $G = [\hat{\rho}_{k,i}\ \hat{\alpha}_{k,i}\ \delta\ \varphi_{s,k}^i\ \varphi_{e,k}^i]^T$ where $(\hat{\rho}_{k,i}, \hat{\alpha}_{k,i}) = \mathbf{h}_{k,i}(\widehat{\mathbf{x}}_{k|k-1},\, \mathcal{M}_i)$ are respectively the predicted perpendicular distance and orientation angle corresponding to the ith line, δ is the width of the validation gate, and $(\varphi_{s,k}^i, \varphi_{e,k}^i)$ are the (counterclockwise positive-oriented) angles between the x^R-axis in the moving robot system and the lines that connect the origin of that coordinate system respectively to the start and end points of the ith line. From the geometry of the problem, it can be shown that

$$\begin{bmatrix} \varphi_{s,k}^i \\ \varphi_{e,k}^i \end{bmatrix} = \begin{bmatrix} \theta_{s,k}^i + \hat{\alpha}_{k,i} - \frac{\pi}{2} \\ -\theta_{e,k}^i + \hat{\alpha}_{k,i} + \frac{\pi}{2} \end{bmatrix} \quad (6)$$

where $\theta_{s,k}^i = \sin^{-1}(\hat{\rho}_{k,i}/d_{s,k}^i)$, $\theta_{e,k}^i = \sin^{-1}(\hat{\rho}_{k,i}/d_{e,k}^i)$, and $d_{n,k}^i = \sqrt{(x_{n,i} - \hat{x}_{k|k-1})^2 + (y_{n,i} - \hat{y}_{k|k-1})^2}$, $n = \{e, s\}$. Assume now that, at instant k, the laser range finder generates 180 data points, $P_{k,l}$, $l = 1, \ldots, 180$, distributed from the $-y^R$ axis to the $+y^R$ axis, with equal spacing of one degree between them. The ith line is declared visible if $\varphi_{s,k}^i \in [-\pi/2, \pi/2]$ or $\varphi_{e,k}^i \in [-\pi/2, \pi/2]$. For each detected line i, following a conversion of the data set from robot to environment coordinates using the predicted pose $\widehat{\mathbf{x}}_{k|k-1}$, we project all data points $P_{k,l}$ for which $\max(0, \varphi_{s,k}^i + \pi/2) < (l\pi)/180 < \min(\pi, \varphi_{e,k}^i + \pi/2)$ onto the perpendicular line connecting the origin of the environment system to the ith line and declare $P_{k,l}$ a valid measurement within the corresponding validation gate of that line if and only if

$$\text{abs}(\rho_i^m - || \, Proj_{\rho_i^m}(P_{k,l}) \, ||) \leq \delta$$

where abs(.) denotes the absolute value of a real number. Finally, once a set of validated measurements is obtained for each detected line i, we use the least-squares algorithm in [2] to estimate the corresponding line features $\rho_{k,i}$ and $\alpha_{k,i}$ and the respective error covariance matrix $\mathbf{R}_{k,i}$. We omit the details in this paper due to lack of space.

3. IMPROVED SIR FILTER

The highly nonlinear nature of the odometric and observation models in (1) and (2) suggests the use of particle filtering [3], also known in robotics as Monte Carlo localization [5], to approximate the optimal minimum mean square error (MMSE) estimate of the unknown state $\mathbf{x}_k$ given the observed features $\mathbf{y}_{1:k}$. Briefly, a particle filter recursively draws M samples $\left\{\mathbf{x}_k^{(j)}\right\}$ according to a known probability density function $q(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}^{(j)}, \mathbf{y}_{1:k})$ referred to as the importance function, and weighs those samples appropriately such that their weighted average converges (in some statistical sense) to the desired MMSE estimate of the state as the

number of samples (or particles) goes to infinity.

Importance Function Approximation In order to minimize the variance of the particle weights conditioned on the observations and the simulated sample trajectories, it is desirable to sample from the optimal importance function, see [3], $q(\mathbf{x}_k \mid \mathbf{x}_{0:k-1}^{(j)}, \mathbf{y}_{1:k}) = p(\mathbf{x}_k \mid \mathbf{x}_{k-1}^{(j)}, \mathbf{y}_k)$. Since there is however no closed-form analytical expression for the ideal importance function, we proceed then as suggested in [3] and linearize the observation equation (5) around $\mathbf{f}(\mathbf{x}_{k-1}^{(j)}, \mathbf{u}_{k-1})$ to approximate the desired importance function by a multivariate normal distribution $\mathcal{N}(\mathbf{x}_k - \mathbf{m}_k^{(j)}, \mathbf{\Sigma}_k^{(j)})$ such that

$$\mathbf{\Sigma}_k^{(j)} = \left[\mathbf{Q}^{-1} + (\mathbf{H}_k^{(j)})^T \mathbf{R}_k^{-1} (\mathbf{H}_k^{(j)})\right]^{-1} \quad (7)$$

$$\begin{aligned}\mathbf{m}_k^{(j)} &= (\Sigma_k^{(j)}) \left\{\mathbf{Q}^{-1}\mathbf{f}(\mathbf{x}_{k-1}^{(j)}, \mathbf{u}_{k-1})\right. \\ &+ (\mathbf{H}_k^{(j)})^T \mathbf{R}_k^{-1} \left[\mathbf{y}_k - \mathbf{h}_k(\mathbf{f}(\mathbf{x}_{k-1}^{(j)}, \mathbf{u}_{k-1}), \mathcal{M})\right. \\ &+ \left.\left. \mathbf{H}_k^{(j)} \mathbf{f}(\mathbf{x}_{k-1}^{(j)}, \mathbf{u}_{k-1})\right]\right\} . \quad (8)\end{aligned}$$

In (8), $\mathbf{R}_k = \text{diag}(\mathbf{R}_{k,1}, \ldots, \mathbf{R}_{k,N})$ and $\mathbf{H}_k^{(j)} = \left[(\mathbf{H}_{k,1}^{(j)})^T \ldots (\mathbf{H}_{k,N}^{(j)})^T\right]^T$ where $\mathbf{H}_{k,i}^{(j)} = \frac{\partial \mathbf{h}_{k,i}(\mathbf{x}_k, \mathcal{M}_i)}{\partial \mathbf{x}_k}$ evaluated at $\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}^{(j)}, \mathbf{u}_{k-1})$. After sampling the jth particle $\tilde{\mathbf{x}}_k^{(j)} \sim \mathcal{N}(\mathbf{x}_k - \mathbf{m}_k^{(j)}, \mathbf{\Sigma}_k^{(j)})$ at the kth time step, we update the corresponding importance weight $\tilde{w}_k^{(j)}$ using the recursion [3]

$$\tilde{w}_k^{(j)} = C_k w_{k-1}^{(j)} \frac{p(\mathbf{y}_k \mid \tilde{\mathbf{x}}_k^{(j)}, \mathcal{M})\, p(\tilde{\mathbf{x}}_k^{(j)} \mid \mathbf{x}_{k-1}^{(j)})}{\mathcal{N}(\tilde{\mathbf{x}}_k^{(j)} - \mathbf{m}_k^{(j)},\, \mathbf{\Sigma}_k^{(j)})} \quad (9)$$

where C_k is computed such that $\sum_j \tilde{w}_k^{(j)} = 1$.

Resampling and Move Steps In order to achieve a further reduction in particle degeneracy, we resample a new particle set $\left\{\overline{\mathbf{x}}_k^{(j)}\right\}$ from the original set $\left\{\tilde{\mathbf{x}}_k^{(j)}\right\}$ with replacement according to the weights $\tilde{w}_k^{(j)}$, see [3]. All particle weights are then reset to $\overline{w}_k^{(j)} = 1/M$, $j = 1, \ldots, M$. Finally, in order to restore diversity in the particle population after the resampling step, we follow [4] and propose an alternative *measurement-driven* Metropolis-Hastings (MH) move step that moves the weighted particle set $\left\{\overline{\mathbf{x}}_k^{(j)}, 1/M\right\}$ to a new weighted sample set $\left\{\mathbf{x}_k^{(j)}, 1/M\right\}$ using the locally optimized importance function $\mathcal{N}(\mathbf{x}_k - \mathbf{m}_k^{(j)}, \mathbf{\Sigma}_k^{(j)})$ as proposal density. Specifically, let

$$w'(\mathbf{x}) = \frac{p(\mathbf{y}_k \mid \mathbf{x}, \mathcal{M})\, p(\mathbf{x} \mid \overline{\mathbf{x}}_{k-1}^{(j)})}{\mathcal{N}(\mathbf{x} - \overline{\mathbf{m}}_k^{(j)},\, \overline{\mathbf{\Sigma}}_k^{(j)})}$$

where $\overline{\mathbf{\Sigma}}_k^{(j)}$ and $\overline{\mathbf{m}}_k^{(j)}$ are obtained from equations (7) and (8) respectively, evaluated at the resampled particle $\overline{\mathbf{x}}_{k-1}^{(j)}$. The proposed move step for $j = 1, \ldots, M$ is as follows:

- Sample $\widehat{\mathbf{x}}_k^{(j)} \sim \mathcal{N}(\mathbf{x}_k - \overline{\mathbf{m}}_k^{(j)}, \overline{\mathbf{\Sigma}}_k^{(j)})$.
- Sample $u \sim \mathcal{U}([0,1])$ and make the decision

$$\begin{aligned}&\text{If } u \leq \min\left\{1, \tfrac{w'(\widehat{\mathbf{x}}_k^{(j)})}{w'(\overline{\mathbf{x}}_k^{(j)})}\right\} \quad \mathbf{x}_k^{(j)} = \widehat{\mathbf{x}}_k^{(j)} \ \text{(accept move)} \\ &\text{else} \qquad\qquad\qquad\qquad\quad \mathbf{x}_k^{(j)} = \overline{\mathbf{x}}_k^{(j)} \ \text{(reject move)}.\end{aligned}$$

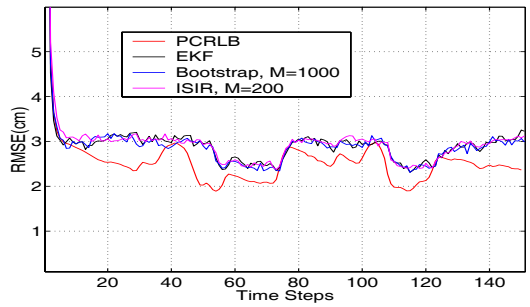
3.1. Dynamic Measurement Validation and Feature Extraction

Given a properly weighted set of samples $\left\{\mathbf{x}_{k-1}^{(j)}, 1/M\right\}$ that represents the posterior probability density function $p(\mathbf{x}_{k-1} \mid \mathbf{y}_{1:k-1})$ at instant $k-1$, we first draw auxiliary particles $(\mathbf{x}_k^*)^{(j)}$ from $p(\mathbf{x}_k \mid \mathbf{x}_{k-1}^{(j)})$ and compute the predicted state estimate $\widehat{\mathbf{x}}_{k|k-1} = (1/M)\sum_j (\mathbf{x}_k^*)^{(j)}$. Using the estimate $\widehat{\mathbf{x}}_{k|k-1}$, we proceed then as in Section 2.2 to compute the features $\mathbf{y}_k$. The auxiliary particles $\left\{(\mathbf{x}_k^*)^{(j)}\right\}$ are then discarded and replaced with the new set $\left\{\tilde{\mathbf{x}}_k^{(j)}\right\}$ sampled from the measurement-driven importance function $\mathcal{N}(\mathbf{x}_k - \mathbf{m}_k^{(j)}, \mathbf{\Sigma}_k^{(j)})$.

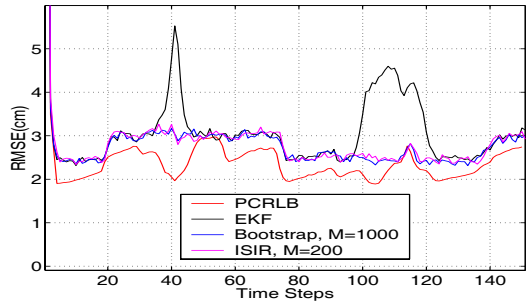
4. EXPERIMENTAL RESULTS

We simulated first the state and observation models in equations (1) and (2) using real odometric data $\{\mathbf{u}_k\}$ recorded by an ISR Magellan Pro robot. For simplicity, we assumed time-invariant, empirically-estimated covariance matrices $\mathbf{Q}$ and $\mathbf{R}_{k,i} = \mathbf{R}, \forall i$, given by $\mathbf{R} = \text{diag}(50^2, (\pi/5)^2)$, $Q(1,1) = 16.49$, $Q(2,2) = 5.24$, $Q(3,3) = 0.0000989$, $Q(1,2) = Q(2,1) = -4.18$, $Q(1,3) = Q(3,1) = 0.0229$, and $Q(2,3) = Q(3,2) = -0.0014$. The initial pose (x_0, y_0) is assumed Gaussian with standard deviation equal to 20 cm in both dimensions. Figures 1(a) and (b) show the root mean-square (RMS) errors for the position estimates, respectively in the x and y coordinates, for (i) a conventional 1000-particle bootstrap filter [3] , (ii) the proposed ISIR filter with $M = 200$ particles, and (iii) the extended Kalman filter (EKF). The RMS curves were estimated from 1000 Monte Carlo runs and are superimposed to the ideal posterior Cramer-Rao lower bound (PCRLB) evaluated using the algorithm in [6]. Figures 1(a) and (b) show that the 200-particle ISIR filter approaches the PCRLB and has roughly the same performance as a bootstrap filter operating with five times as many particles. The EKF also approaches the PCRLB most of the time, but its performance in the the estimation of the y-coordinate deteriorates in certain regions where, according to the recorded odometry, the robot is making sharp turns.

Next, we tested the 200-particle ISIR filter and the EKF using *real measurements* recorded in a cluttered room using a laser scanner mounted on the Magellan Pro robot. The sampling period between two consecutive recorded laser



(a)



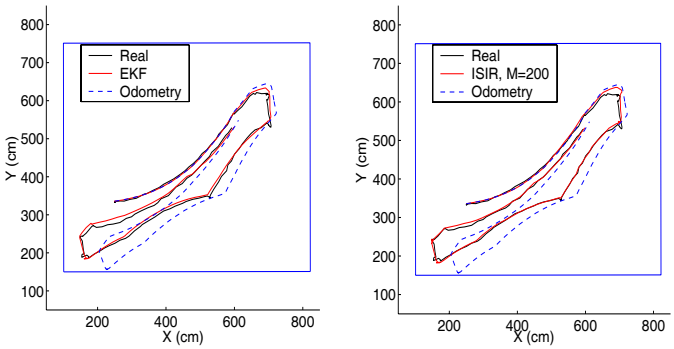
(b)

Fig. 1. RMS error position estimates (zoomed view): (a) x-coordinate, (b) y-coordinate.

scans is $1s$; the average processing times per scan on a AMD Athlon 1.67 GHz PC running MatLab 6.5 are respectively $0.625s$ for the ISIR filter and $0.393s$ for the EKF. Figures 2(a) and (b) show the estimated robot trajectory, respectively for the ISIR and extended Kalman filters, over 150 consecutive time steps. The filtered trajectories are superimposed to the trajectory estimated by the deterministic odometric model alone (without any data assimilation) and compared to our best estimate of the ground truth. We see from Figure 2 that, as expected, the trajectory predicted by the odometric model deviates from the true robot's path as time increases, highlighting the need for data assimilation. Both the ISIR and extended Kalman filters were capable however of tracking the robot's position fairly accurately for this particular set of data, even though the chosen trajectory includes abrupt turns close to corners where the robot's sensor has a very narrow field of view. Figure 2 also suggests that the ISIR filter outperforms the EKF slightly following the robot's first and second turns (respectively top right and bottom left of the curve). More extensive experiments are needed however to determine whether that improvement is statistically significant.

5. CONCLUSIONS

We introduced in this paper an improved sampling/importance resampling (ISIR) particle filter for mobile robot localization using a parametric model of the environment. Synthetic data experiments showed that a 200-particle ISIR filter approaches the ideal PCRLB, outperforming the extended Kalman filter (EKF) and matching the performance of a 1000-particle bootstrap filter. Experiments with real data showed on the other hand that both the 200-particle ISIR filter and the EKF were capable of tracking the robot's pose fairly accurately despite heavy clutter in the room and a suboptimal specification of the odometric and measurement model parameters.



(a) (b)

Fig. 2. Estimated robot trajectory using real data: (a) EKF, (b) ISIR filter.

6. REFERENCES

[1] S. Thurn, W. Bugard, and D. Fox, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Mach. Learn.*, vol.31, pp. 29-53, 1998.

[2] P. Jensfelt and H. I. Christensen, "Pose tracking using laser scanning and minimalistic environment models," *IEEE Trans. Robot. and Autom.*, v.17, n.2, pp. 138-147, April 2001.

[3] A. Doucet, S. J. Godsill, and C. Andrieu,"On sequential Monte Carlo sampling methods for Bayesian filtering," *Stat. Comput.*, vol.10, pp. 197-208, 2000.

[4] W. R. Gilks and C. Berzuini,"Following a moving target ‗ Monte Carlo inference for dynamic Bayesian models," *J. R. Statist. Soc. B*, vol.63, pp. 127-146, 2001.

[5] S. Thrun, D. Fox, and W. Burgard, "Monte Carlo localization with mixture proposal distributions," *Proc. of the AAAI Nat. Conf. on Artif. Intelligence*, 2000.

[6] P. Tichavský, C. H. Muravchik, and A. Nehorai,"Posterior Cramér-Rao Bounds for Discrete-Time Nonlinear Filtering," *IEEE Trans. on Signal Processing*, vol.46, n.5, pp. 1386-1395, May 1998.