MOBILE ROBOT LOCALIZATION USING ODOMETRY AND KINECT SENSOR

N. Ganganath and H. Leung

Department of Electrical and Computer Engineering
Schulich School of Engineering
University of Calgary, Calgary, Alberta, Canada
Email:{ngmarasi, leungh}@ucalgary.ca

ABSTRACT

This paper presents a mobile robot localization system for an indoor environment using an inexpensive sensor system. The extended Kalman filter (EKF) and the particle filter (PF) is used for sensor fusion in pose estimation in order to minimize uncertainty in robot localization. The robot is maneuvered in a known environment with some visual landmarks. The prediction phase of the EKF and the PF are implemented using the information from the robot odometry whose error may accumulate over time. The update phase uses the Kinect measurements of the landmarks to correct the robot's pose. Experiment results show that, despite its low cost, the accuracy of the localization is comparable with most state-of-the-art odometry based methods.

Index Terms— Extended Kalman filter, particle filter, robot localization, odometry, Kinect sensor

1. INTRODUCTION

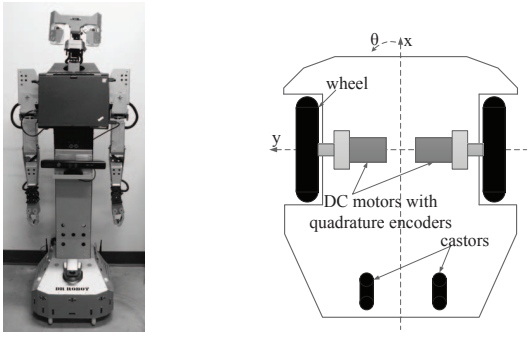
Localization is one of the fundamental problems in mobile robot navigation and motion planning. In an indoor environment with a flat floor plan, localization is identified as a problem of estimating the pose, i.e. position and orientation of a mobile robot, when the map of the environment, sensor readings, and executed actions of the robot are provided [1]. Odometry is one of the most popular solutions for wheeled mobile robots (WMRs). The encoders mounted on the wheels provide robot motion information to update the mobile robot pose. However, odometric pose estimation unboundedly accumulates errors due to different wheel diameters, wheel-slippage, wheel misalignment, and finite encoder resolution [2]. Experiment results presented in this paper, together with previous studies [3], concur that the largest factor in odometry error is due to the rotation of the robot.

The odometry errors can be eliminated using an auxiliary sensor to observe landmarks in an environment. Different sensors, such as cameras [4], sonar sensors [5], and laser range finders [5] have been used to detect landmarks and obtain the required measurements. Many solutions have been proposed for the pose estimation of mobile robots employing Kalman filtering [4], particle filtering [3, 5], and Markov localization [6].

In this paper, we propose an accurate and low cost mobile robot localization method using odometry and a Kinect sensor. The odometry model used in this paper is capable of tracking any arbitrary robot motion. The odometry and the Kinect sensor measurements are fused using the extended Kalman filter (EKF) [4] and the particle filter (PF) [3] to provide more accurate localization results. The correct detection of landmarks by applying Hough transform and depth estimation using the Kinect sensor have significantly contributed to the better performance of the robot localization. The experiments are carried out with H20 mobile robot (see Fig 1(a)) and the results are provided in order to interpret the accuracy of the proposed method.

2. ODOMETRY SYSTEM

As shown in Fig 1(b), the H20 WMR has two drive wheels and two castors. The two drive wheels are connected to DC motors with 15-bit quadrature encoders. These encoders are capable of monitoring the revolutions and steering angles of the drive wheels. The main advantage offered by these encoders is their high resolution. The odometry is implemented using the results from these encoders. The two output channels of the quadrature encoder indicate both position and direction of rotation.



(a) H20 WMR with a Kinect sensor (b) Odometry system

Fig. 1. The robot design.

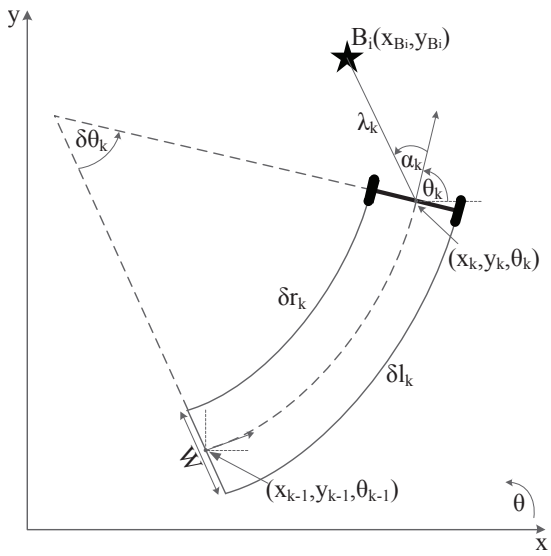


Fig. 2. The robot coordinate system and the position of i^{th} landmark.

2.1. Odometry Model

The system state vector $\mathbf{x}_k = [x_k\ y_k\ \theta_k]^T$ describes the pose of the WMR at k^{th} time step, where (x_k, y_k) is the position of the mid-axis point and θ_k is the heading angle as illustrated in Fig 2. When the wheel base (W) is given, the change in orientation in k^{th} time step $\delta\theta_k$ can be calculated as follows

$$\delta\theta_k = \frac{(\delta r_k - \delta l_k)}{W}, \tag{1}$$

where δl_k and δr_k represent the distances traveled by the left and right wheels, respectively. The distance traveled by the mid-axis point in k^{th} time step can be calculated as the average of the distances traveled by left and right wheels.

$$\delta d_k = \frac{(\delta l_k + \delta r_k)}{2}. \tag{2}$$

According to Fig 2, the displacement of the WMR in k^{th} time step can be determined by,

$$\delta x_k = \frac{\delta d_k}{\delta\theta_k}[\sin(\theta_{k-1} + \delta\theta_k) - \sin(\theta_{k-1})], \tag{3}$$

$$\delta y_k = \frac{\delta d_k}{\delta\theta_k}[\cos(\theta_{k-1}) - \cos(\theta_{k-1} + \delta\theta_k)]. \tag{4}$$

For a small value of $\delta\theta_k$, the next state of the system can be predicted as

$$\hat{\mathbf{x}}_k^- = \begin{bmatrix} \hat{x}_{k-1} + \delta d_k \cos(\hat{\theta}_{k-1} + \frac{\delta\theta_k}{2}) \\ \hat{y}_{k-1} + \delta d_k \sin(\hat{\theta}_{k-1} + \frac{\delta\hat{\theta}_k}{2}) \\ \hat{\theta}_{k-1} + \delta\theta_k \end{bmatrix}. \tag{5}$$

Here, $\hat{\mathbf{x}}_{k-1} = [x_{k-1}\ y_{k-1}\ \theta_{k-1}]^T$ is the previous state estimate of the system.

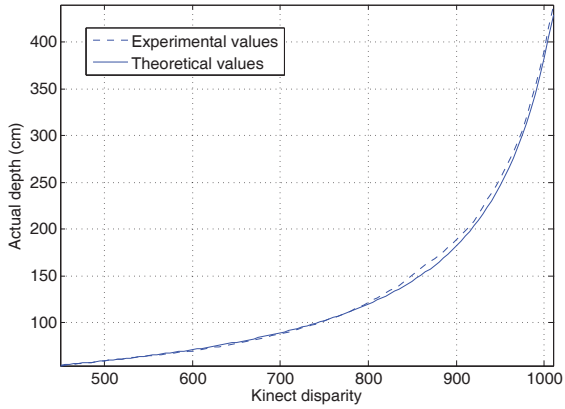


Fig. 3. Kenect depth calibration results.

3. KINECT SENSOR

Kinect is a motion sensing input device introduced by Microsoft for video gaming. It enables users to control the game with physical gestures and voice-based commands. This touch-free sensor is powered by an RGB camera and a depth sensor. Its operation range is 0.8m – 3.5m with the resolution of 1cm at a distance of 2m. In this work, a Kinect sensor is mounted on the robot so that the optical axes of both RGB and IR cameras are perpendicular to the wheel axle. The height of the Kinect is arbitrary.

3.1. Kinect Calibration

The Kinect depth image is acquired using the Light Coding technology [7]. The coded light is captured by the IR camera in order to produce the Kinect disparity matrix. The relationship between the Kinect disparity and actual depth value is given by,

$$z = \frac{b \times f}{\frac{1}{8}(\mu - d_{kinect})}, \tag{6}$$

where z is the actual depth, b is the distance between the IR camera and laser-based IR projector lenses, f is the focal length of the IR camera in pixels, μ is an offset value for a given Kinect device, and d_{kinect} is the Kinect disparity which provides 2048 levels of sensitivity in VGA resolution with 11-bit depth. The factor $1/8$ is used due to the fact that d_{kinect} is in $1/8$ pixel units. Unlike ordinary stereo pairs, the actual depth does not become infinity at the zero Kinect disparity.

Fig 3 illustrates the Kinect depth calibration results. The theoretical curve represents the results obtained using (6). The experimental values were obtained by manually measuring the physical distance to the target object from the Kinect sensor. The target object was moved away from the Kinect sensor by 10cm each time and corresponding Kinect disparity values were obtained. However this calibration results may slightly differ from one sensor to another. Therefore each sensor should calibrated separately before using it for depth estimation.

3.2. Landmark Detection with Kinect Sensor

In the measurement update phase, the WMR uses landmarks around it to estimate its pose. In this work, different colored circles are used as landmarks. The images and their depth values are acquired by the Kinect sensor. Hough transform filters are used to detect the landmarks (circles) in RGB image frames. The landmarks are distinguished from each other using HSI color model.

The azimuth angle α_k with respect to the WMR x-axis and the distance λ_k to the i^{th} landmark $B_i(x_{B_i}, y_{B_i})$ at a time instant k can be calculated using the Kinect measurements and the focal length of the RGB camera. The value of the measurement function can be determined as

$$\mathbf{h}(\hat{\mathbf{x}}_k^-) = \begin{bmatrix} \tan^{-1}\left(\frac{y_{B_i}-\hat{y}_k^-}{x_{B_i}-\hat{x}_k^-}\right) - \hat{\theta}_k^- \\ \sqrt{(x_{B_i}-\hat{x}_k^-)^2 + (y_{B_i}-\hat{y}_k^-)^2} \end{bmatrix}. \quad (7)$$

The difference between Kinect observation and measurement function is used to update the odometry predication in both the EKF and the PF.

4. FILTER REALIZATION

In this work, the EKF and the PF are used for the pose estimation of the WMR. The initial state estimate is taken as $\mathbf{x}_k = \mathbf{0}$, i.e. the initial vehicle pose defines the base coordinate frame. The prediction steps of the filters are executed using the odometry information. The measurement update step takes place only when a landmark is detected. Whenever a landmark is not detected, the predicted state is taken as the state estimate for the next iteration of the filter, i.e $\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^-$.

4.1. Eextended Kalman Filter

The EKF is applicable to non-linear systems where the associated uncertainties are assumed to be Gaussian. Here, the system, measurement and input noises are assumed to be zero mean and uncorrelated. Hence the noise covariance matrices, $\mathbf{Q}_k$, $\mathbf{R}_k$, and $\mathbf{U}_k$ become diagonal. The system and measurement noises are also assumed to be time invariant, which leads $\mathbf{Q}_k$ and $\mathbf{R}_k$ to be time invariant as well [4].

$$\mathbf{Q}_k = \begin{bmatrix} \sigma_{x_k}^2 & 0 & 0 \\ 0 & \sigma_{y_k}^2 & 0 \\ 0 & 0 & \sigma_{\theta_k}^2 \end{bmatrix}, \quad (8)$$

$$\mathbf{R}_k = \begin{bmatrix} \sigma_{\alpha_k}^2 & 0 \\ 0 & \sigma_{\lambda_k}^2 \end{bmatrix}. \quad (9)$$

The system position noise variances of (x, y) coordinates are denoted by $\sigma_{x_k}^2$ and $\sigma_{y_k}^2$. Here, $\sigma_{\theta_k}^2$ is the orientation noise variance, while $\sigma_{\alpha_k}^2$ and $\sigma_{\lambda_k}^2$ are the measurement noise variances.

The variance of the noise generated by each encoders can be determined as the sum of the variance of each independent unit because the encoder measurements are statistically independent and they accumulate errors over time. Therefore it is possible to assume that the variance of each unit of travel is proportional to the total distance traveled. The covariance of the input vector $\mathbf{u}_k = [\delta l_k \ \delta r_k]^T$ can be given by,

$$\mathbf{U}_k = \begin{bmatrix} \sigma_L^2 & 0 \\ 0 & \sigma_R^2 \end{bmatrix}, \quad (10)$$

where $\sigma_L^2 = k_L^2 \sum_{i=0}^{k} |\delta l_i|$ and $\sigma_R^2 = k_R^2 \sum_{i=0}^{k} |\delta r_i|$. Here, k_L^2 and k_R^2 are positive constants [2].

4.2. Particle Filter

As a solution to the Gaussian density assumption inherent in EKF localization, Monte Carlo localization was introduced (also known as PF localization) for mobile robots [5]. It can globally localize the robot due to its capability of representing multi-modal distributions. It requires less amount of memory and returns more accurate results compared to grid-based Markov localization [5]. The uncertainty in state estimation is represented by a set of samples that are randomly drawn from the probability density function, which are also known as particles. If a landmark is detected, particle weights are re-evaluated based on the Kinect measurements. In addition to the prediction and the update steps common to both the EKF and the PF, the PF performs a resampling step. It avoids the depletion of the sample population after few iterations. In this work, we have used linear time resampling [8] to eliminate the near-zero-weight particles.

5. RESULTS

The experiments were carried out in an indoor environment of 7m×2m with 4 landmarks. The starting point and 5 way points were marked on the floor with exact measurements. The WMR was maneuvered through the way points and measurements were taken.

Fig 4 illustrates the statistical results obtain from 20 individual experiments. The ellipses around the mean values represent the standard deviations of the results. The mean values of the estimated results are close to the way points while their standard deviations are very small. It confirms that the proposed methods are capable of providing more accurate localization results. It should be noted that some pose errors, especially the errors in y-direction were due to the deviations in maneuvering the WMR onto the way points. Therefore the actual errors are expected to be slightly smaller. Since all methods under test are affected by such errors, the results remain qualitatively unchanged.

The root mean squared error (RMSE) of localization at each way point is shown in Fig 5. According to the given results, the RMSE error of odometry measurement keeps increasing considerably with the distance traveled. This leads to an erroneous pose estimation of the WMR. However the RMSE has been reduced substantially using the EKF and the

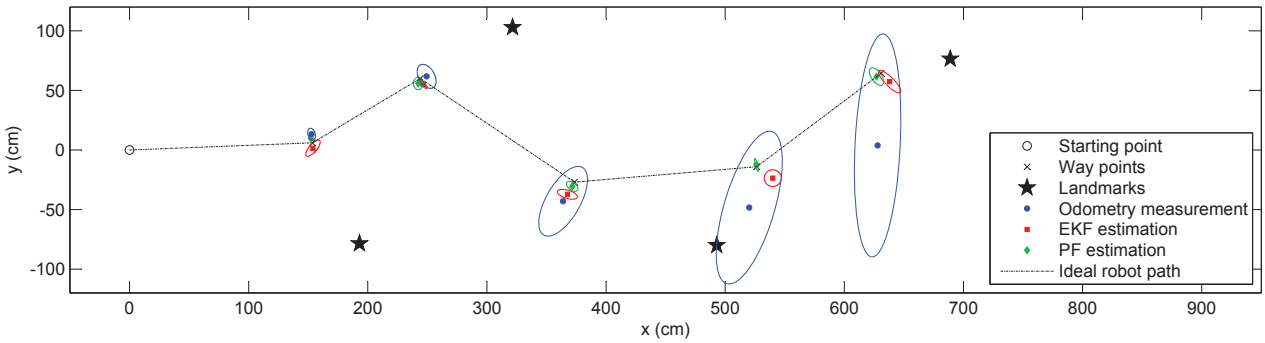


Fig. 4. Standard deviations of the localization results.

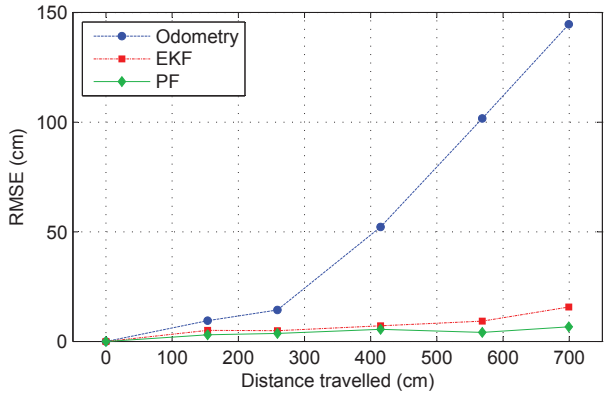


Fig. 5. Root mean squared localization error.

PF with the Kinect sensor. This proves that the proposed estimation methods are robust and stable, while the results obtained using the PF are slightly more stable compared to the the EKF.

The Euclidean distance error at each way point is given in Table 1. These values were obtained by calculating the Euclidean distance between the way points and the average of the measureed/estimated robot positions. The errors in estimation results are significantly lower compared to the errors in odometry measurements. This suggests that the proposed methods using the Kinect sensor can perform accurate mobile robot localization. According to the given results, the PF pose estimation is more accurate compared to the EKF pose estimation with the Kinect sensor.

6. CONCLUSION

The odometric pose estimation accumulates errors with time. More precise mobile robot localization can be achieved by reducing the uncertainty in the odometric pose estimation using a Kinect sensor to observe landmarks in an environment. The EKF and the PF can be used for the sensor fusion in pose estimation. The proposed measurement model for the Kinect sensor together with odometry model is capable of providing an accurate system model for a WMR. A robust and accurate mobile robot localization method using an inexpensive sensor system was proposed, implemented, and tested on H20 mobile robot.

Table 1. Euclidean distance error at each way point (cm).

Way point	1	2	3	4	5
Odometry	7.3634	5.6828	18.6643	34.9313	60.7425
EKF	4.5655	5.1575	11.7987	16.5151	10.0205
PF	3.9186	4.4614	3.9726	1.4753	4.8919

7. REFERENCES

[1] J.-S. Gutmann and D. Fox, "An Experimental Comparison of Localization Methods Continued," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2002, vol. 1, pp. 454 – 459.

[2] K.S. Chong and L. Kleeman, "Accurate Odometry and Error Modelling for a Mobile Robot," in *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 1997, vol. 4, pp. 2783–2788.

[3] Ioannis Rekleitis, *Cooperative Localization and Multi-robot Exploration*, Ph.D. thesis, School of Computer Science, McGill University, Montreal, Canada, 2003.

[4] E. Kiriy and M. Buehler, "Three-State Extended Kalman Filter for Mobile Robot Localization," Tech. Rep., McGill University, Montreal, Canada, 2002.

[5] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte Carlo Localization for Mobile Robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, May 1999, vol. 2, pp. 1322 –1328.

[6] D. Fox, W. Burgard, and S. Thrun, "Markov Localization for Mobile Robots in Dynamic Environments," *Journal of Artificial Intelligence Research*, vol. 11, no. 3, pp. 391–427, 1999.

[7] PrimeSense Ltd., Palo Alto, CA 94301, USA, *The PrimeSensor Reference Design*, 1.08 edition, 2010.

[8] J. Carpenter, P. Clifford, and P. Fearnhead, "Improved Particle Filter for Nonlinear Problems," *Proc. IEEE Radar, Sonar and Navigation*, vol. 146, no. 1, pp. 2–7, Feb 1999.