Auton Robot (2006)
DOI 10.1007/s10514-006-6474-8

Relative localization using path odometry information

Nakju Lett Doh · Howie Choset · Wan Kyun Chung

Published online: 15 June 2006

Abstract All mobile bases suffer from localization errors. Previous approaches to accommodate for localization errors either use external sensors such as lasers or sonars, or use internal sensors like encoders. An encoder's information is integrated to derive the robot's position; this is called odometry. A combination of external and internal sensors will ultimately solve the localization error problem, but this paper focuses only on processing the odometry information. We solve the localization problem by forming a new odometry error model for the synchro-drive robot then use a novel procedure to accurately estimate the error parameters of the odometry error model. This new procedure drives the robot through a known path and then uses the shape of the resulting path to estimate the model parameters. Experimental results validate that the proposed method precisely estimates the error parameters and that the derived odometry error model of the synchro-drive robot is correct.

Keywords Relative localization · Mobile robot · Odometry error model · Odometry calibration · Synchro-drive robot · Differential drive robot · Generalized voronoi graph

N. L. Doh (✉)
Intelligent Robot Research Division, Electronics and Telecommunications Research Institute, Korea

H. Choset
Robotics Institute, Carnegie Mellon University, USA

W. K. Chung
Department of Mechanical Engineering, Pohang University of Science and Technology, Korea

1. Introduction

Relative localization, which estimates the position of a robot from a starting point, has received considerable attention from the robotics community. Two means of localization have dominated the community: the use of external sensors to localize the robot (Kim et al., 2004; Borenstein and Feng, 1996b; Kim et al., 1999; Chung et al., 2001; Murata and Hirose, 1993; Adam et al., 1999; Chong et al., 1999; Jensfelt and Kristensen, 2001; Fox et al., 1999) and the filtering of odometry information to ascertain the robot's location relative to a starting location (Borenstein and Feng, 1996a; Larsen et al., 1998; Roy and Thrun, 1999; Martinelli et al., 2003; Wang, 1988; Kelly, 2004). The filtering, commonly referred to as dead-reckoning or deduced reckoning, suffers from inaccuracies caused by imprecise kinematic parameters and wheel slippage.

Already, researchers have developed methods to compensate for odometry errors. Early work began with Wang (Wang, 1988) who proposed a relative localization technique for the differential drive robot. In the 1990s, Borenstein (Borenstein and Feng, 1996a) made significant advances by analyzing odometry error sources and proposed a way of error compensation for the differential drive robot. In this approach, a robot is driven around a square allowing an estimate of the error parameters by looking at the displacement between the start and goal locations. Recently, Kelly (Kelly, 2004) proposed an odometry calibration method using linearized error equations. The works by Borenstein and Kelly used the endpoint errors for calibration.

This paper introduces a new method for odometry compensation that uses the shape of the entire path as opposed to just looking the endpoints. Similar works that use the entire path are proposed in Larsen and Martinelli's works (Larsen

et al., 1998, Martinelli et al., 2003). They utilized the augmented Kalman filter for odometry calibration.

Our new method is called the *Path Comparison (PC) method* which is based on the concept that sensor-based navigation through a *known path* bounds the absolute error while there are no bound with odometry. In this work, we use the *generalized Voronoi graph* (GVG) (Choset and Nagatani, 2001) to force the robot to follow a known path. Recall that the GVG is a set of points whose edges and nodes are equidistant to two or more objects.

A key feature of the GVG is that its sensor-based control laws force the robot to precisely follow the GVG without relying on odometry information. While recording odometry information, we force the robot to follow a path along the GVG and track the same path but in the reverse direction, not using the odometry information. Because of the GVG control laws, the actual path the robot follows back and forth is the same. However, if one were to look at the odometry information for the forward path and the reverse path, they would look different because of odometry error. Even though they look different, the two paths will have the same gross shape. By noting the coordinate transform that deforms one path into another, one can derive the parameters that describe the dead-reckoning error model for the mobile base. We can then use those parameters to correct odometry error.

In this paper, we also develop an error model for synchro-drive robots. Recall that a synchro-drive robot has three or four wheels that translate and rotate in a synchronous manner. Its motion is accomplished by two control motors: one for translation and the other for rotation. For the synchro-drive robot, it has been reported that the odometry error is attributed to a wheel angle, i.e. the heading direction of the wheel (Duckett et al., 2000). However, there is currently no established model to explain this behavior. Recently Martinelli (Martinelli, 2002) suggested an error model for the encoder signals of the synchro-drive robot. In this paper, we propose an error model at the world level where the encoder signals are decoded into a real world posture (i.e. the position and heading angle of the robot).

For the development of our error model for the synchro-drive robot, we assume that the major sources of odometry errors are disturbance forces and moments that result from the wheel's misalignment. Such forces and moments constantly drag and rotate the robot. By analyzing how these forces and moments affect the synchro-drive robot, we suggest an error model that not only explains the previous trends but also agrees well with experimental results.

This paper is organized as follows: In Section 2, an odometry error model of the synchro-drive robot is derived. Section 3 explains a way of estimating the error parameters, and Sections 4 and 5 show the experimental results of the PC-method for the synchro and the differential drive robots. Then, Section 6 provides performance comparison results between the PC-method and previous calibration techniques, and conclusion follows.

2. Odometry error model of synchro-drive robots

In this section, an odometry error model of synchro-drive robots is derived. There are three state variables (x, y, θ) in the synchro-drive robot where (x, y) is the position and θ is the heading angle of the robot. The kinematic equation of the synchro-drive robot is given by

$$\begin{pmatrix} x(k+1) \\ y(k+1) \\ \theta(k+1) \end{pmatrix} = \begin{pmatrix} x(k) + \Delta D(k) \cos(\theta(k) + \frac{\Delta\Theta(k)}{2}) \\ y(k) + \Delta D(k) \sin(\theta(k) + \frac{\Delta\Theta(k)}{2}) \\ \theta(k) + \Delta\Theta(k) \end{pmatrix}, \quad (1)$$

where $\Delta D(k)$ and $\Delta\Theta(k)$ are scalars corresponding the magnitude of the translation and rotation at time step k.

For the synchro-drive robot, possible sources of the odometry errors are as follows:

1. Offset distance between the nominal center of wheels and the center of mass induced by uneven mass distribution: A in Fig. 1.
2. Offset distance between the center of rotation and wheel: B in Fig. 1.
3. Different wheel radii: *C* in Fig. 1.
4. Wheel misalignment: *D* in Fig. 1.

We assume that the effects of the first three sources are ignorable because of the following reasons.

1. Offset distance between the nominal center of wheels and the center of mass induced by uneven mass distribution: The uneven mass distribution changes the position of the center of mass. This change will induce different forces if a base plate of the wheels is elastic (Fig. 2(a)). In general, however, the base plate can be assumed as rigid (i.e. infinite stiffness) and each wheel will exert the same repulsive force (Fig. 2(b)).
2. Offset distance between the center of rotation and wheel: The offset distance will give rise to a translational error in

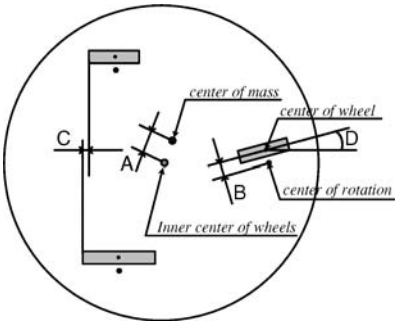


Fig. 1 Possible sources of the odometry error of the synchro-drive robot.

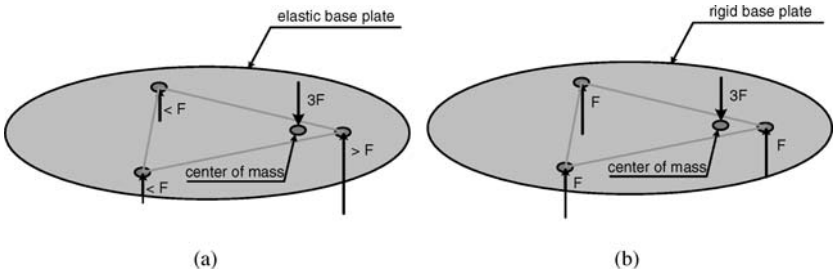


Fig. 2 The effect of the uneven mass distribution: (a) it will induce different forces if a base plate of the wheels is elastic. (b) In general, however, the base plate can be assumed as rigid (i.e. infinite stiffness) and each wheel will exert the same repulsive force.

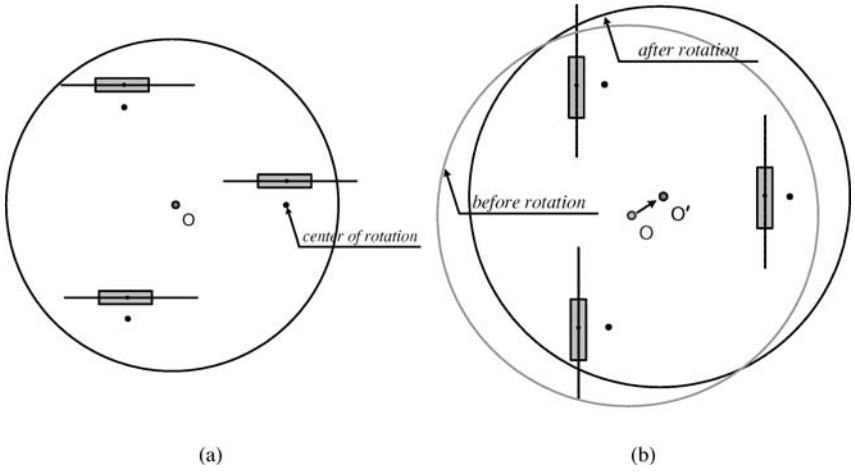


Fig. 3 The effect of the offset distance between the center of rotation and wheel: the offset gives rise to a translational error in case of a pure rotation. (a) is before rotation and (b) is after rotation. Note that the origin O is shifted to O'.

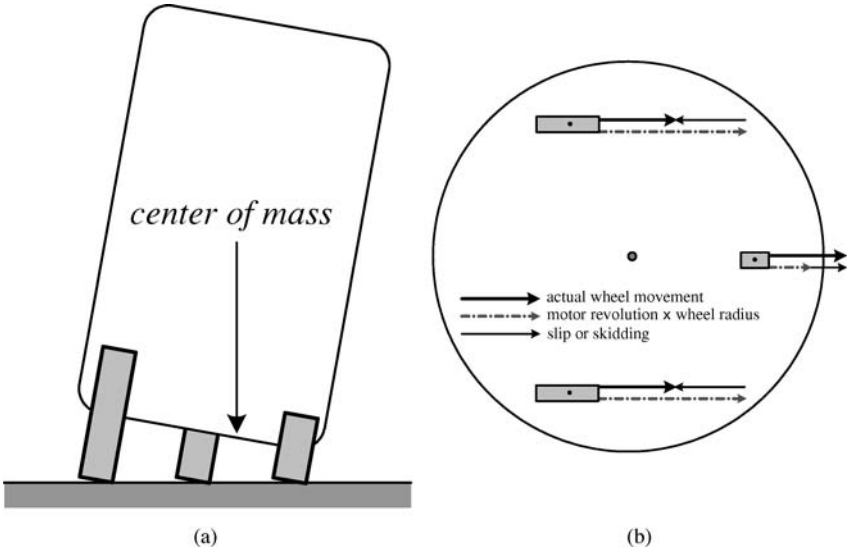


Fig. 4 The effect of the different wheel radii: (a) The different wheel radii will make the robot inclined and thus move the center of mass. (b) The actual distance that the wheel moved is a sum of motor revolution times wheel radius and slip or skidding.

case of a pure rotation (Fig. 3). However, this error is not magnified as the robot moves (i.e. it is not accumulative) and can be ignored.

3. Different wheel radii: The different wheel radii will make the robot inclined and thus move the center of mass (Fig. 4(a)). But, as explained above, the effect of the movement can be ignored. A more important factor is slip and skidding. For example, the actual distance that the wheel moved is a sum of motor revolution times wheel radius and slip or skidding (Fig. 4(b)). The slip or skidding happen in the translational direction and does not severely affect the heading angle of the robot. Thus, these are not accumulative and can be ignored.

In contrast, the wheel misalignment induces additional forces and moments, which forcefully drag and rotate the robot. These additional forces and moments introduce errors into the heading angle θ (i.e. accumulative). If these forces and moments were not present, then θ would be equal to a wheel angle θ_w, i.e. the heading angle of a wheel.

Table 1 Nomenclature.

Notation	Meaning
(x_t, y_t)	Transformed coordinate by θ_w
$(\vec{i}_t, \vec{j}_t)$	Unit vectors on (x_t, y_t)
(α, β, γ)	Misaligned angles of wheel A, B, C
F	Force on each wheel
L	Length between two wheels
O	Nominal center of wheels

The following subsections explain how the additional forces and moments, respectively, induce the rotational errors.

2.1. Rotational errors by the additional forces

Let us define some notation in Fig. 5 and Table 1. When a robot translates, forces are exerted to the floor via robot's wheels. Let $\vec{F}_{\text{real}}$ be the real force that the robot experiences and let F_{est} be the estimated force the robot should experience if there were no wheel misalignment.[1] The difference can be derived from geometric relationships in Fig. 5 as

$$\begin{aligned}\vec{\Delta F} &= \vec{F}_{\text{real}} - \vec{F}_{\text{est}} = \{F(\cos\alpha + \cos\beta + \cos\gamma)\ \vec{i}_t \\ &\quad + F(\sin\alpha + \sin\beta + \sin\gamma)\vec{j}_t\} - 3F\vec{i}_t\ . \end{aligned} \tag{2}$$

Note that if $\alpha = \beta = \gamma = 0$, then we have no wheel misalignment and $\vec{\Delta F} = 0$. In this work, we assume that α, β, γ are all close to zero. Thus (2) becomes

$$\begin{aligned}\vec{\Delta F} &\simeq -\frac{F}{2}(\alpha^2 + \beta^2 + \gamma^2)\ \vec{i}_t + F(\alpha + \beta + \gamma)\ \vec{j}_t \\ &\equiv \Delta F_x\ \vec{i}_t + \Delta F_y\ \vec{j}_t\ . \end{aligned} \tag{3}$$

The ΔF_x and ΔF_y drag the robot by an amount of Δd_x and Δd_y as seen in Fig. 6. The Δd_x and Δd_y can be written as

$$\begin{aligned}\Delta d_x &= \frac{T^2 \Delta F_x}{2M}, \\ \Delta d_y &= \frac{T^2 \Delta F_y}{2M}, \end{aligned} \tag{4}$$

where M is the mass of the robot and T is the infinitesimal sampling time.

[1] From here on, the time notation 'k' will be omitted in unnecessary cases for simplicity.

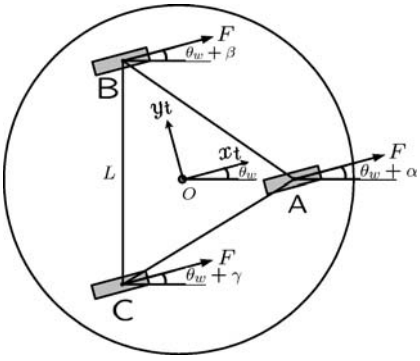


Fig. 5 Notations for error analysis of the synchro-drive robot.

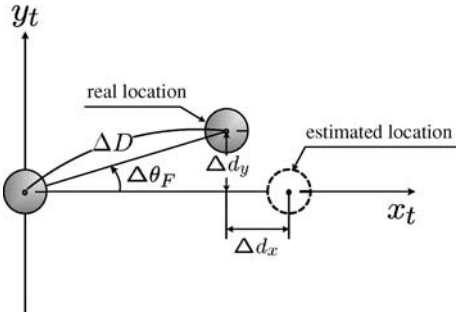


Fig. 6 The effect of disturbance forces on the rotational errors. This figure is exaggerated. The disk at the origin represents the starting location of the robot and the dotted circle represents where the robot should have moved to after being commanded to move in the x_t direction by an amount of ΔD. However, because of the wheel misalignment, the robot moves in the $\Delta\theta_F$ direction and is displaced from the estimated location by an amount of Δd_x, Δd_y.

The scalar ΔD is

$$\Delta D = TV, \tag{5}$$

where V is the velocity of the robot.

Let us define $\Delta\theta_F$ as a rotational error induced by ΔF during T. From Fig. 6 and (3-5), the $\Delta\theta_F$ can be calculated as

$$\begin{aligned}\Delta\theta_F &= \sin^{-1}\frac{\Delta d_y}{\Delta D} \simeq \frac{\Delta d_y}{\Delta D} = \frac{TF(\alpha + \beta + \gamma)}{2VM} \\ &\equiv E_F \frac{TF}{V}, \end{aligned} \tag{6}$$

where E_F is an error parameter of the error model whose unit is kg^{-1}.

2.2. Rotational errors by the additional moments

Now we will look at how the additional moments from the wheel misalignment cause the rotational errors. The nominal center of wheels, O, is assumed to be the center of rotation. The equation of the disturbance moments on O (de-

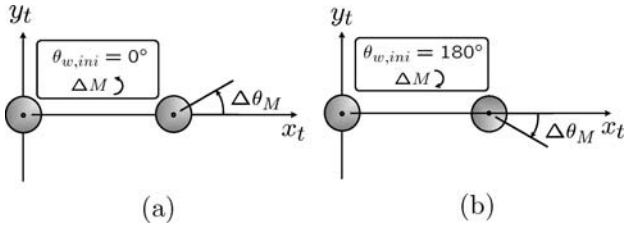


Fig. 7 The effects of the additional moments to the rotational errors for the cases when the initial wheel angles, $\theta_{w,ini}$ are (a) 0.0° and (b) 180.0°.

note it ΔM) can be derived from geometric relationships in Fig. 5 as

$$\begin{aligned}\Delta M &= \frac{\sqrt{3}LF}{3}\left\{\sin(\theta_w+\alpha)-\frac{1}{2}\sin(\theta_w+\beta)\right.\\ &\quad -\frac{\sqrt{3}}{2}\cos(\theta_w+\beta)-\frac{1}{2}\sin(\theta_w+\gamma)\\ &\quad \left.+\frac{\sqrt{3}}{2}\cos(\theta_w+\gamma)\right\}\\ &\equiv FC_1\sin(\theta_w+C_2),\end{aligned} \tag{7}$$

where C_1, C_2 are constants.

The scalar $\Delta\theta_M$, which is defined as a rotational error induced by ΔM during T, can be written as

$$\begin{aligned}\Delta\theta_M &= \frac{T^2FC_1\sin(\theta_w+C_2)}{2I}\\ &= T^2FE_{M1}\sin(\theta_w+E_{M2}),\end{aligned} \tag{8}$$

where I is the robot's inertia. Here E_{M1} and E_{M2} are error parameters whose units are $\mathrm{kg}^{-1}\cdot\mathrm{m}^{-1}$ and rad. The scalar $\Delta\theta_M$ is shown in Fig. 7 when $\alpha = 0.01rad$, $\beta = \gamma = 0$ and initial wheel angles $\theta_{w,ini} = 0.0°$ or 180.0°. This figure shows that $\Delta\theta_M$ depends on θ_w.

2.3. Odometry error model

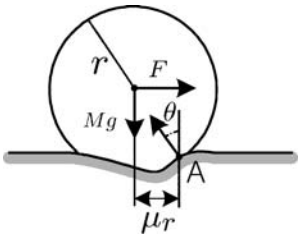
The major source of the odometry errors is the rotational errors because these are accumulative while the translational errors are not. Thus an error model for the rotational errors will be developed.

The rotational errors during T, denoted as $\Delta\theta_e$, can be derived as

$$\begin{aligned}\Delta\theta_e &= \Delta\theta_F+\Delta\theta_M\\ &= E_F\frac{TF}{V}+T^2FE_{M1}\sin(\theta_w+E_{M2}).\end{aligned} \tag{9}$$

When a wheel rolls, there aries deflection between the wheel and the surface as shown in Fig. 8. In this figure, M, g, r, θ, and μ_r are mass, gravity, wheel radius, attack angle of the normal force and a distance induced by the deflection. By the moment equilibrium at point A (Beer and Johnston, 1994), the force is given as

Fig. 8 Force induced by the deflection between the wheel and the surface.



$$\begin{aligned}F &> \frac{Mg\mu_r}{r\cos\theta} && \text{when accelerating.}\\ F &= \frac{Mg\mu_r}{r\cos\theta} && \text{when } V \text{ is constant.}\\ F &< \frac{Mg\mu_r}{r\cos\theta} && \text{when decelerating.}\end{aligned} \tag{10}$$

Unfortunately, the force is hardly measurable. But note that if V is constant, F is also constant. In this case, (9) can be simplified as (11).

$$\Delta\theta_e = E_1+E_2\sin(\theta_w+E_3), \tag{11}$$

where E_1, E_2, and E_3 are error parameters of the simplified error model. Here all the units of $E_{1,2,3}$ are in radians. Throughout this paper, the simplified model is used.

The kinematic equation of odometry whose rotational errors are compensated for by (11) is

$$\begin{pmatrix}x(k+1)\\ y(k+1)\\ \theta(k+1)\end{pmatrix} = \begin{pmatrix}x(k)+\Delta D(k)\cos\left(\theta(k)+\frac{\Delta\Theta(k)}{2}\right)\\ y(k)+\Delta D(k)\sin\left(\theta(k)+\frac{\Delta\Theta(k)}{2}\right)\\ \theta(k)+\Delta\Theta(k)+\Delta\theta_e\end{pmatrix}. \tag{12}$$

Two examples of the effects of $\Delta\theta_F$ and $\Delta\theta_M$ on $\Delta\theta_e$ when $\alpha = 0.01rad$ and $\beta = \gamma = 0$ are shown in Fig. 9 for different initial wheel angles. The dashed lines are the real paths of two different runs, one for $\theta_{w,ini} = 0°$ and the other for $\theta_{w,ini} = 180°$, after being commanded to move in the x_t direction. The odometry path in this figure is represented by segment $\overline{OA}$.

For experimental validation, a synchro-drive robot was commanded to track 5.1 meter along x_t. The y_t position variations for various wheel angles are shown in Fig. 10. The simulation model displays the same qualitative behavior as the real robot.

Note that the proposed error model is valid when V and F are constants. It is not realistic in a populated environment where the robot frequently changes its velocity. However, this

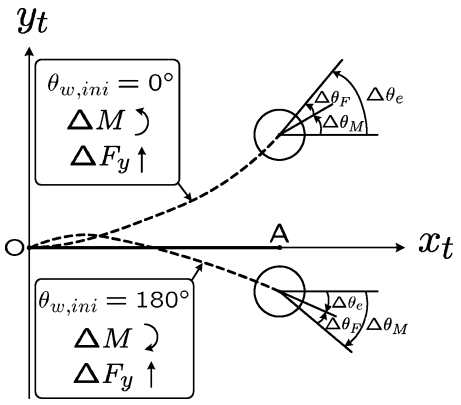


Fig. 9 Effects of $\Delta\theta_F$ and $\Delta\theta_M$ on the rotational errors ($\Delta\theta_e$) for different initial wheel angles. The dashed lines are the real paths of two different runs for $\theta_{w,ini} = 0°$ and 180° after being commanded to move in the x_t direction. The odometry path in this figure is represented by segment $\overline{OA}$.

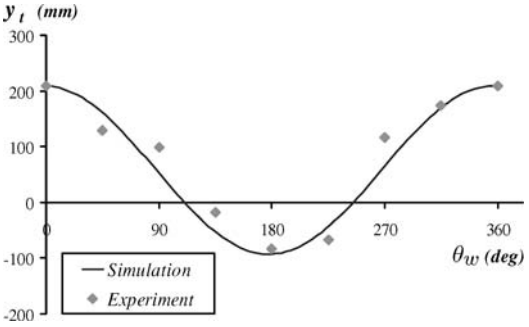


Fig. 10 The y_t position variations for various wheel angles after $5.1m$ line movement along x_t.

model is accurate enough in a navigation of a long distance because the robot moves most of its navigation in a constant velocity. The experimental results in Section 4 will verify the usefulness of the proposed model.

3. Error parameters estimation using the PC-method

This section describes our estimation technique for error parameters using path odometry information called the *Path Comparison (PC) method*. The PC-method is based on the idea that a sensor-based navigation through the GVG bounds the absolute error while there are no bounds with odometry. Recall that the GVG (Choset and Nagatani, 2001) is a set of points whose edges and nodes are equidistant to two or more objects. Thus if a robot navigates the GVG twice, once forward and then backward, two different odometry paths are generated but along the same path on the GVG in the real world. By noting the coordinate transform that deforms one path into another, one can derive the error parameters and then use those parameters to correct odometry.

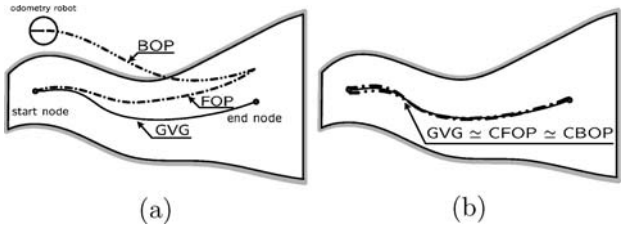


Fig. 11 Schematic explanation of the PC-method. (a) is a plot before correction and (b) is a plot after correction.

The estimation procedure of the PC-method is given below with a schematic explanation in Fig. 11.

1. Navigate a GVG path from a start node to an end node. The start and end nodes can be chosen arbitrarily as long as the length between the two nodes is long "enough" for parameter estimation. This first set of data is denoted as a Forward Odometry Path (FOP).
2. Turn the robot 180° and navigate from the end node to the start node. We will denote this odometry data as a Backward Odometry Path (BOP).
3. Take an initial guess of the error parameters which produce a Corrected FOP (CFOP) and a Corrected BOP (CBOP) using the given error model.
4. Starting with this initial guess, perform optimizations to find the error parameters that minimize the position error between the CFOP and the CBOP.

Two remarks have to be made. First, note that our paths are represented by a discrete set of points. In other words, we do not have a continuous function but rather a sequence of closely spaced points which "approximate" the path. To perform our optimization, we have to make an identification between points in the FOP and the BOP. Naturally, the start point of the FOP and the end point of the BOP are identified, and the end point of the FOP and the start point of the BOP are the same. A problem occurs when the FOP and the BOP have a different number of points. In such a situation, we have to explicitly pair up points in the FOP and BOP, as indicated in Fig. 12.

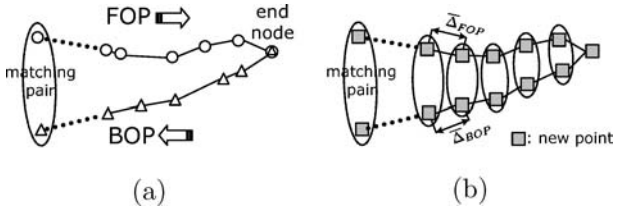


Fig. 12 Matching pair extraction procedure. (a) shows sequential points of the FOP and BOP where the dotted ellipse is the matching pair. In (b), new points are sampled for every Δ_{FOP} or Δ_{BOP} and the matching pairs are selected in order starting from the end node.

For that purpose, we define averages of the stepwise translations of the FOP and the BOP as

$$\overline{\Delta}_{\mathrm{FOP}} = \frac{L_{\mathrm{FOP}}}{N},$$
$$\overline{\Delta}_{\mathrm{BOP}} = \frac{L_{\mathrm{BOP}}}{N}, \tag{13}$$

where L_{FOP} and L_{BOP} are the total lengths of the FOP and the BOP, and N is the number of points of the FOP or the BOP. Then new points are sampled for every $\overline{\Delta}_{\mathrm{FOP}}$ or $\overline{\Delta}_{\mathrm{BOP}}$ via the interpolation, and the matching pairs are selected in order from the end node (Fig. 12(b)).

Second, the GVG does not allow us to follow "the same path" strictly because various errors arise from the sensor, the control, and the matching pair selection. However, in a long GVG path, it is reasonable to assume that the GVG tracks the real path.

We believe that the PC-method can estimate the error parameters of all the mobile bases. But, in this paper, we apply the PC-method to the synchro and the differential drive robots whose odometry models are given.

4. Application I: Synchro-drive robot

The PC-method is applied to a synchro-drive robot (Nomad 200 in Fig. 13) to validate the synchro-drive error model in Section 2 and to verify the estimation performance of the PC-method in Section 3.

We ran the robot through a GVG path with a length of 94.4 m (Fig. 14(a)) to get the FOP and the BOP. Then the error parameters, which minimize position errors between the CFOP and the CBOP, were calculated by the steepest

Fig. 13 Picture of the Nomad 200 robot with synchro-drive mobile base.

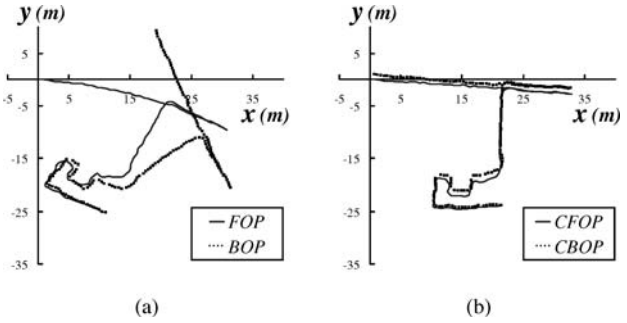


Fig. 14 Forward and backward odometry paths for the error parameter estimation of the synchro-drive robot. (a) is the FOP and BOP, and (b) is the CFOP and CBOP.

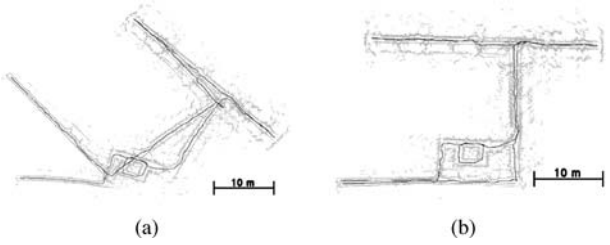


Fig. 15 The constructed maps by the synchro-drive robot. (a) is an odometry map and (b) is a PC-method corrected map.

descent method. The resulting parameters are

$$[E_1 \quad E_2 \quad E_3] = [1.29e^{-4} \quad 2.30e^{-4} \quad 1.571]. \tag{14}$$

Then (12) was used to compensate for the odometry errors. Fig. 14(b) displays the CFOP and CBOP after error correction.

To verify the accuracy of the error parameters, the robot was set to yield two sets of odometry data. One is the raw odometry data and the other is the corrected odometry data given by the corrected kinematic equation (12) and the error parameters (14). Then the robot navigated 177.2 m, returning to the starting point. The speed of the robot was intentionally kept low, 0.13 m/s, to minimize non-systematic errors (Borenstein and Feng, 1996a). Two maps from the raw odometry and the corrected odometry are shown in Fig. 15.

This experiment was repeated a total of three times. Endpoint errors, which are simply the final points estimated by the odometry because the real end point is the same as the real start point, were analyzed in Fig. 16. The average error of the raw odometry was 19.52 m, and that of the corrected odometry was 0.28 m, which is a 70.6 fold enhancement in this experiment.

Note that the error parameters should be estimated with a long GVG path for better accuracy. Naturally, if we run the robot on a shorter path at a faster speed, the error parameters become inaccurate. In our experiments, we ran the robot on a 49 m GVG path (half the length of the previous one) at twice the speed to get another set of error parameters. These

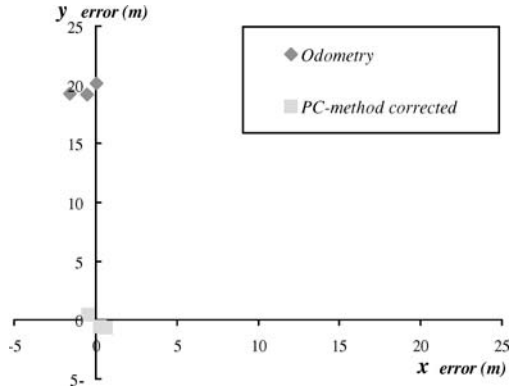


Fig. 16 Endpoint errors after 177.2 m of navigation with the synchro-drive robot.

error parameters were used to compensate for the odometry, and the average of the final errors after 177.2 m of navigation increased to 4.23 m. These errors were induced by inaccurate error parameters and the non-systematic errors (Borenstein and Feng, 1996a) caused by the fast speed.

5. Application II: Differential drive robot

This section shows that the PC-method precisely estimates the error parameters of the differential drive robot. According to Borenstein (Borenstein and Feng, 1996a) the odometry error of the differential drive robot is induced by the difference of wheel radius and inaccurate wheel base length. The different wheel radii are modeled by the ratio between the right and left wheel as $E_d = D_R/D_L$ where D_R, D_L are the diameters of the right and left wheel, respectively. The inaccurate wheel base length is defined by a fraction of actual and nominal wheel base as $E_b = b_{\text{actual}}/b_{\text{nominal}}$. This section uses the same error model for the differential drive robot.

The PC-method was applied to a differential drive robot (Nomad Scout in Fig. 17) to get the error parameters. The robot was run along a GVG path whose length was 138 m

Fig. 17 Picture of the Nomad Scout robot with a differential drive mobile base.

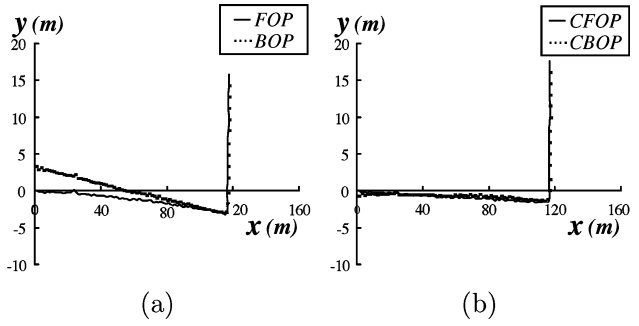


Fig. 18 Forward and backward odometry paths for the error parameter estimation of the differential drive robot. (a) is the FOP and BOP, and (b) is the CFOP and CBOP.

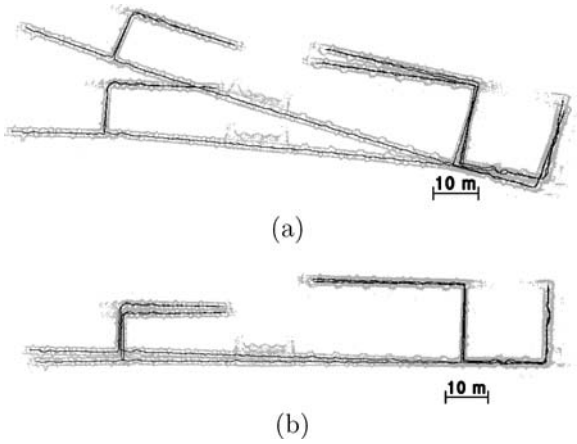


Fig. 19 Maps constructed by the differential drive robot. (a) is an odometry map, and (b) is a PC-method corrected map.

and the steepest descent method was used to find the error parameters as $E_d = 1.000135$, $E_b = 1.003095$. The FOP and BOP are shown in Fig. 18(a), and the CFOP and CBOP are shown in Fig. 18(b).

To verify the accuracy of the error parameters, the robot was set to yield two sets of odometry data. One is the

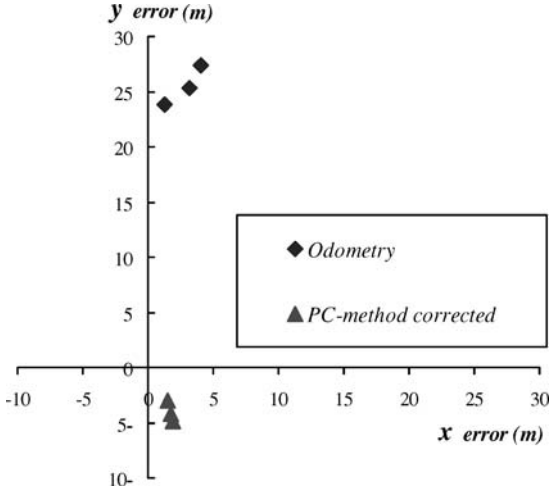


Fig. 20 Endpoint errors after 719.1 m of navigation with the differential drive robot.

Fig. 21 A differential drive robot that was used for performance comparison.

raw odometry data and the other is the corrected odometry data. Then the robot, moving at 0.25 m/s, navigated 719.1 m, returning to the starting point. Two maps from the raw odometry and the corrected odometry are shown in Fig. 19.

This experiment is repeated a total of three times. The endpoint errors were analyzed in Fig. 20. The average error of the raw odometry was 25.7 m, and that of the corrected odometry was 4.4 m.

6. Experimental comparison

In this section, we compare the calibration performance of the PC-method with two other approaches. Those are the UMBmark (Borenstein and Feng, 1996a) and Kelly's technique (Kelly, 2004).

The UMBmark is a well known tool which uses the endpoints after a 4 m × 4 m square navigation is repeated 10 times. It is only good for the differential drive robot and in a square path. In contrast, Kelly's technique is applicable for all mobile bases and in any arbitrary path.

For fair comparison, we estimated the error parameters from the same path length and at the same floor condition. The path length was fixed to 160m as required by the UMBmark (4 m × 4 m by 10 times). Among various mobile bases, the differential drive was selected because the UMBmark is only applicable to that type of robot. A differential drive robot (Fig. 21), which was built in our lab, was used for experiments. We intentionally did not use the Nomad Scout in Section 5 to show that the PC-method can be applied for any mobile base.

First, the error parameters were estimated by the UMBmark whose calibration results are shown in Fig. 22. Then, Kelly's technique was applied for the error parameters esti-

Table 2 The error parameters of the differential drive robot from the UMBmark, Kelly's technique and the PC-method.

	The UMBmark	Kelly's technique	The PC-method
E_d	0.998659	0.999169	0.998993
E_b	1.005730	1.000844	1.000300

Table 3 Means and standard deviations of the endpoint errors of various methods.

Method	Mean (m)	Standard deviation (m)
Odometry	5.49	2.05
UMBmark	2.29	1.56
Kelly's tech	1.42	0.75
PC-method	0.82	0.57

Table 4 Results of student's t-test for the mean of the PC-method against that of odometry, UMBmark, and Kelly's technique.

Case	t-value	Probability
Odometry vs PC	6.92	$> 99.9\%$
UMBmark vs PC	2.77	98.8%
Kelly's tech. vs PC	1.99	94.0%

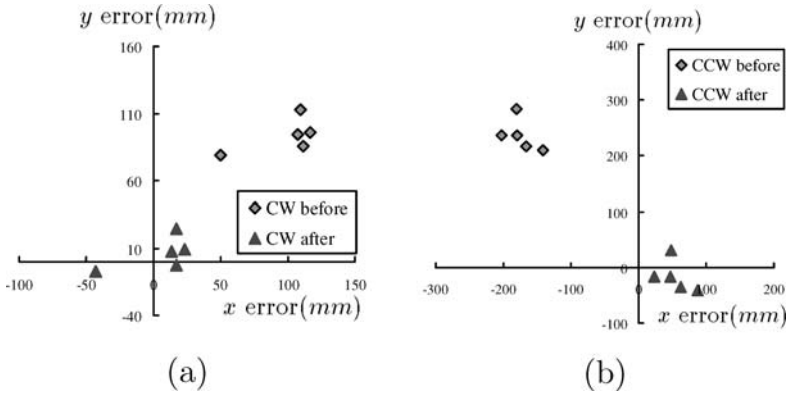


Fig. 22 The endpoint errors after a 4 m × 4 m square navigation. (a) shows the results before and after calibration for CW direction, and (b) plots those for CCW direction.

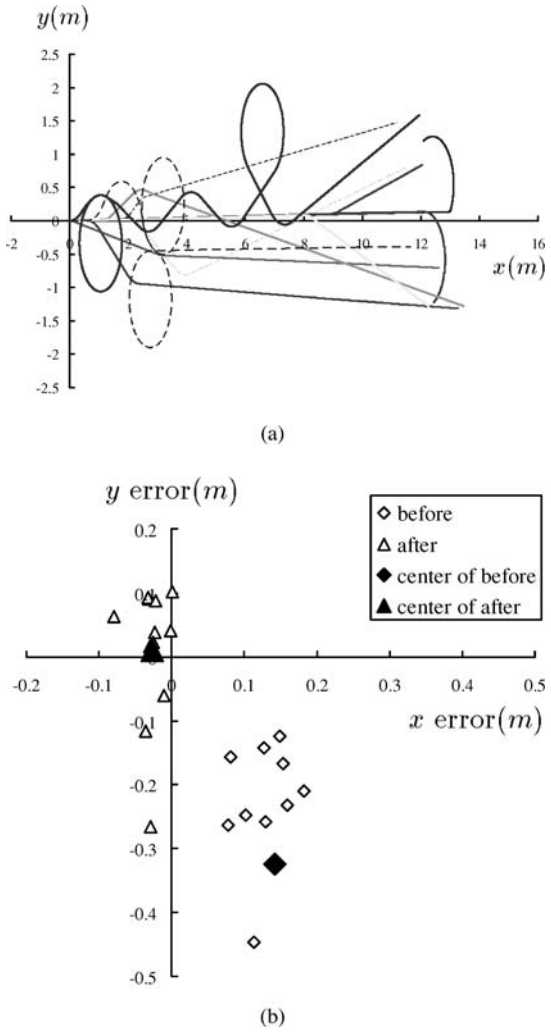


Fig. 23 Experimental data of Kelly's technique. (a) shows 12 arbitrary paths whose total length is 160 m, and (b) plots the endpoint errors before and after calibration.

mation using 12 endpoints after arbitrary path navigations. Fig. 23(a) shows the paths which are measured to be 160 m, and Fig. 23(b) plots the endpoint errors. Finally, the PC-method was applied for a 160 m path as shown in Fig. 24.

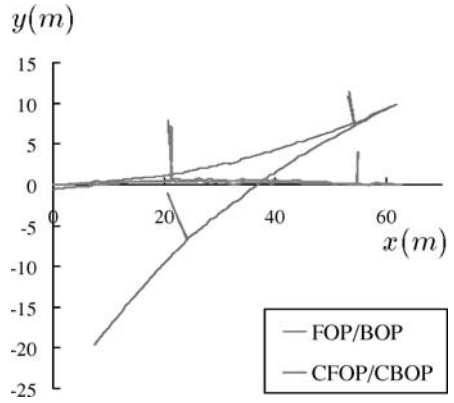
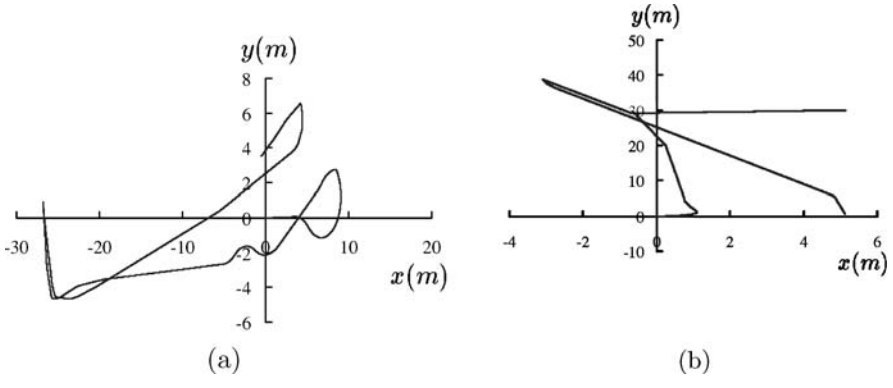


Fig. 24 Forward and backward odometry paths where the total length is 160 m. The thin line is the FOP and BOP, and the thick line is the CFOP and CBOP.

Estimated error parameters from the various methods are listed in Table 2. To validate the accuracy of these parameters, the robot was set to yield four estimation values from the raw odometry, the UMBmark, Kelly's technique, and the PC-method. Then the robot navigated 10 paths shown in Fig. 25. Each path had an arbitrary shape, and the length of each path was about 100 m (giving a total test length of about 1 km). The endpoint errors, which are the differences between the real endpoint and the estimated values, are shown in Fig. 26(a). The average magnitudes of the errors are plotted in Fig. 26(b) which validate that the error parameters from the PC-method are more accurate than those of the UMBmark or Kelly's technique by 2.8 and 1.7 times, respectively.

The Student's t-tests were conducted for the means and standard deviations of the endpoint errors given in Table 3 to check whether the enhancement of the performance was done by chance or not. The mean of the PC-method is tested against that of odometry, UMBmark, and Kelly's technique. The results are given in Table 4 which shows that the mean of the PC-method is statistically different from other means with a probability of at least 94.0%.

Fig. 25 Two examples of 10 test paths. Each path length is around 100 m, and the shape is arbitrary.



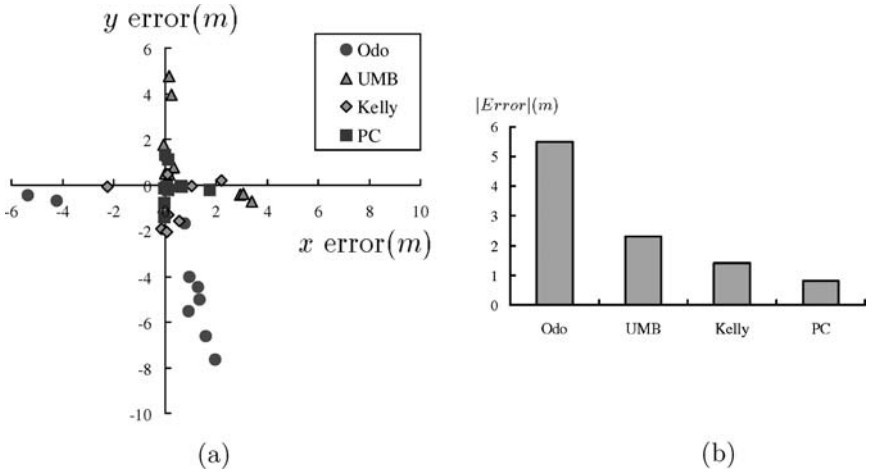


Fig. 26 The endpoint errors after navigating 10 arbitrary paths. (a) shows the endpoint errors, and (b) plots the average magnitudes of the errors.

7. Conclusion

This paper described a new method for accurate relative localization using path odometry information. This method is called the Path Comparison (PC) method. The PC-method can be applied to both synchro and differential drive robots, and it estimates the error parameters accurately by using lengthy path data.

Moreover, an odometry error model of synchro-drive robots was developed by analyzing the sources of odometry error and by deriving the disturbance forces and moments equations. The developed model assumes a constant velocity and force which is not realistic in the strict sense. But the usefulness of this model was experimentally shown for the navigation of a long distance.

Experiments were conducted for the synchro and the differential drive robots using the PC-method. The proposed method produced $3 \sim 5$ m errors for a 720 m travel path, and the accuracy of the PC-method is 2.8 times better than the UMBmark, and 1.7 times better than Kelly's technique.

Practically, a localization cannot be successfully performed using only odometry even though it is precisely calibrated. However, if a calibrated odometry is combined with other localization algorithm, the robustness of the localization algorithm will be increased.

Acknowledgments This research was supported by the National Research Laboratory Program (M1-0302-00-0040-03-J00-00-024-00) of the Ministry of Science & Technology, a grant (02-PJ3-PG6-EV04-0003) of Ministry of Health and Welfare, and a grant (05MI1410) of Ministry of Information and Communication, Republic of Korea.

References

Adam, A., Rivlin, E., and Rostein. H. 1999. Fusion of fixation and odometry for vehicle navigation. *IEEE Trans. on Systems, Man, and Cybernetics*, 29(6):593–603.

Beer, F.P. and Johnston, E.R. 1994. Vector mechanics for engineers. McGraw Hill, pp. 344–345.

Borenstein, J. and Feng. L. 1996a. Measurement and correction of systematic odometry errors in mobile robots. *IEEE Trans. on Robotics and Automation*, 12(6):869–880.

Borenstein, J. and Feng, L. 1996b. Gyrodometry: A new method for combining data from gyros and odometry in mobile robots. *International Conference on Robotics and Automation*, 423–428.

Chong, K.S. and Kleeman, L. 1999. Mobile robot map building for an advanced sonar arrray and accurate odometry. *International Journal of Robotics Research*, 18(1):20–36.

Choset, H. and Nagatani, K. 2001. Topological SLAM: Toward exact localization without explicit localization. *IEEE Trans. on Robotics and Automation*, 17(2):125–137.

Chung, H., Ojeda, L., and Borenstein. J. 2001. Accurate mobile robot dead-recknoing with a precision-calibrated fiber-optic gyroscope. *IEEE Trans. on Robotics and Automation*, 17(1):80–84.

Duckett, D., Marsland, S., and Shapiro, J. 2000. Learning globally consistent maps by relaxation. *International Conference on Robotics and Automation*, 3841–3846.

Fox, D., Burgard, W., and Thrun, S. 1999. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence*, 11:391–427.

Jensfelt, P. and Kristensen, S. 2001. Active global localization for a mobile robot using myltiple hypothesis tracking. *IEEE Trans. on Robotics and Automation*, 17(5):748–760.

Kelly, A. 2004. Linearized error propagation in odometry. *International Journal of Robotics Research*, 23:179–218.

Kim, M.C., Chung, W.K., and Youm. Y. 1999. Posture estimation of car-like mobile robot using disturbance conditions. *Advanced Robotics*, 13(2):189–202.

Kim, S.B., Choi, K., Lee, S., Choi, J., Hwang, T., Jang, B., and Lee, J. 2004. A bimodal approach for land vehicle localization. *ETRI Journal*, 26(5):497–500.

Larsen, T.D., Bak, M., Andersen, N.A., and Ravn, O. 1998. Location estimation for autonomously guided vehicle using an augmented kalman filter to autocalibrate the odometry. *FUSION 98, Spie conference Las Vegas*.

Martinelli, A. 2002. The odometry error of a mobile robot with a synchronous drive system. *IEEE Trans. on Robotics and Automation*, 18(3):399–405.

Martinelli, A., Tomatis, N., Tapus, A., and Siegwart, R. 2003. Simultaneous localization and odometry calibration for mobile robot. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 1499–1504.

Murata, S. and Hirose, T. 1993. Onboard locating system using real-time image processing for a self-navigating vehicle. *IEEE Trans. on Industrial Electronics*, 40(1):145–154.

Roy, N. and Thrun, S. 1999. Online self-calibration for mobile robots. *International Conference on Robotics and Automation*, 2292–2297.

Wang, C.M. 1988. Location estimation and uncertainty analysis for mobile robots. *International Conference on Robotics and Automation*, 1230–1235.

Nakju Lett Doh received his BS, his MS, and his Ph.D. degree in Mechanical Engineering from Pohang University of Science and Technology (POSTECH), KOREA, in 1998, 2000, and 2005, respectively. Since then, he is a senior researcher in Intellgient Robot Reserarch Division, Electronics and Telecommunications Research Institute (ETRI), KOREA. He received the glod prize in Intelligent Robot Contest hosted by Northern KyoungSang Province at 2000 and the gold prize in Humantech Thesis Competition hosted by Samsung Electronics at 2005. In 2003, he got the best student paper award in IEEE International Conference on Robotics and Automation held in Taiwan. His research interests are the localization and navigation of mobile robots and ubiquitous robotic space for intelligent robot navigation.

Howie Choset is an Associate Professor of Robotics at Carnegie Mellon University where he conducts research in motion planning and design of serpentine mechanisms, coverage path planning for de-mining and painting, mobile robot sensor based exploration of unknown spaces, and education with robotics. In 1997, the National Science Foundation awarded Choset its Career Award to develop motion planning strategies for arbitrarily shaped objects. In 1999, the Office of Naval Research started supporting Choset through its Young Investigator Program to develop strategies to search for land and sea mines. Recently, the MIT Technology Review elected Choset as one of its top 100 innovators in the world under 35. Choset directs the Undergraduate Robotics Minor at Carnegie Mellon and teaches an overview course on Robotics which uses series of custom developed Lego Labs to complement the course work. Professor Choset's students have won best paper awards at the RIA in 1999 and ICRA in 2003. Finally, Choset is a member of an urban search and rescue response team using robots with the Center for Robot Assisted Search and Rescue. Now, he is active in extending the mechanism design and path planning work to medical mechatronics.

Wan Kyun Chung received his BS degree in Mechanical Design from Seoul National University in 1981, his MS degree in Mechanical Engineering from KAIST in 1983, and his Ph.D. in Production Engineering from KAIST in 1987. He is Professor in the school of Mechanical Engineering, POSTECH (he joined the faculty in 1987). In 1988, he was a visiting professor at the Robotics Institute of Carnegie-Mellon University. In 1995 he was a visiting scholar at the university of California, Berkeley. His research interests include the localization and navigation for mobile robots, underwater robots and development of robust controller for precision motion control. He is a director of National Research Laboratory for Intelligent Mobile Robot Navigation. He is serving as an Associate Editor for IEEE Tr. on Robotics, international editorial board for Advanced Robotics.