J Intell Robot Syst (2011) 62:187–203
DOI 10.1007/s10846-010-9441-8

EKF-Based Localization of a Wheeled Mobile Robot in Structured Environments

Luka Teslić · Igor Škrjanc · Gregor Klančar

Received: 19 May 2009 / Accepted: 7 June 2010 / Published online: 24 June 2010

Abstract This paper deals with the problem of mobile-robot localization in structured environments. The extended Kalman filter (EKF) is used to localize the four-wheeled mobile robot equipped with encoders for the wheels and a laser-range-finder (LRF) sensor. The LRF is used to scan the environment, which is described with line segments. A prediction step is performed by simulating the kinematic model of the robot. In the input noise covariance matrix of the EKF the standard deviation of each robot-wheel's angular speed is estimated as being proportional to the wheel's angular speed. A correction step is performed by minimizing the difference between the matched line segments from the local and global maps. If the overlapping rate between the most similar local and global line segments is below the threshold, the line segments are paired. The line parameters' covariances, which arise from the LRF's distance-measurement error, comprise the output noise covariance matrix of the EKF. The covariances are estimated with the method of classic least squares (LSQ). The performance of this method is tested within the localization experiment in an indoor structured environment. The good localization results prove the applicability of the method resulting from the classic LSQ for the purpose of an EKF-based localization of a mobile robot.

Keywords Mobile robot · Localization · Extended Kalman Filter · Covariance matrix · Line feature

1 Introduction

Localization is a fundamental problem to be solved in mobile robotics. If the robot knows its own pose in the environment, it is truly autonomous in executing

L. Teslić (✉) · I. Škrjanc · G. Klančar
Faculty of Electrical Engineering, University of Ljubljana,
Tržaška 25, 1000 Ljubljana, Slovenia
e-mail: luka.teslic@fe.uni-lj.si

given tasks. However, localizing a mobile robot using only odometry is inaccurate, since the error arising from the uncertainties of the odometric model and the measurement noise of the odometric sensor is accumulating. The robot can improve the information about its own pose by comparing the local map, obtained from the current environment scan, with the already-built global environment map. A robot can simultaneously build a global environment map and then use this map to localize itself in the environment, which is known as a SLAM (simultaneous localization and mapping) algorithm [12]. SLAM is a computationally very complex algorithm, and as shown in [3], various approaches have been developed to reduce this complexity. In the literature, many approaches and algorithms involved in solving the SLAM, localization and mapping problem have been proposed [4, 10, 11, 18, 22, 26]. The occupancy grids divide the environment into grids, where each cell of the grid is either occupied or free [25]. However, occupancy grids require a huge amount of computer memory and are therefore not appropriate when modelling a large environment [27]. Line segments, which are often applied for a representation of the environment [2, 8, 9, 28], are also chosen in this paper to model the environment because they require a smaller amount of computer memory. The drawback of the line-based environment description is that it is only suited [14, 15, 17, 19, 20, 29] to structured environments, which are mainly composed of straight-edged objects or walls. Usually, these are indoor environments. In [19] a comparison of line-extraction algorithms using a 2D laser rangefinder is reported. Based on this comparison a split-and-merge algorithm was chosen, because of its speed and correctness. In [7] the split-and-merge fuzzy line extraction algorithm is introduced. It uses a prototype-based fuzzy-clustering approach in a split-and-merge framework. This structure allows the use of the fuzzy-clustering algorithm without any previous knowledge of the number of prototypes. In [30] a robust regression model is proposed for segment extraction in the static and dynamic environments, considering the noise of the sensor data and the outliers that correspond to dynamic objects, such as the people in motion. In order to navigate a mobile robot in unknown environments a path planning algorithm must be designed. Here, the computational efficiency of the used algorithms is also of great importance. In [21] a method for obstacle avoiding trajectories using a minimum computation concept is proposed.

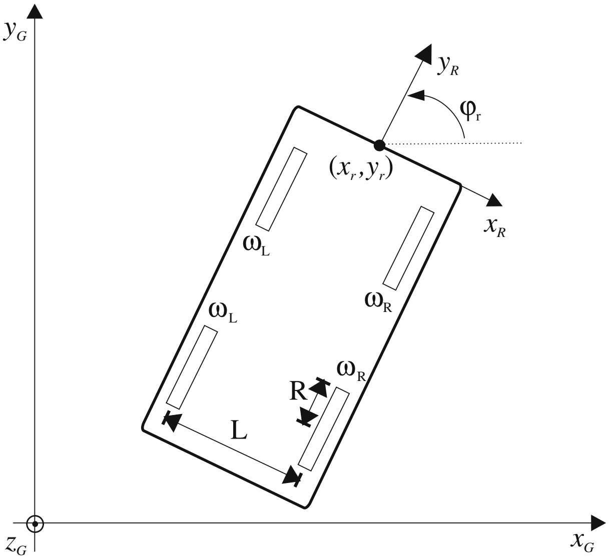
The extended Kalman filter is very often applied to solve the localization problem. Similarly to parameter estimation schemes [5, 6] the convergence properties of the EKF significantly depend on estimating the input- and output-noise covariance matrices of the process, which have to be appropriately set. In an environment represented by line segments, the line parameters' covariances comprise the output-noise covariance matrix of the EKF. A method for estimating the covariances of the line-equation parameters resulting from the classic LSQ was proposed in our previous work [24]. However, in the presented paper the focus is given to the experimental results of an EKF-based localization of a mobile robot, where the line parameters' covariances are estimated with the method resulting from classic LSQ. The prediction step of the EKF is performed by simulating the kinematic model of the robot. The standard deviation of each robot-wheel's angular speed is estimated as being proportional to the wheel's angular speed in the input-noise covariance matrix. To perform the correction step of the EKF the line segments from the local and global maps, which correspond to the same environment line segments (e.g., a wall), must be found. A matching procedure, where the most similar local and global line segments are paired if the overlapping rate between them is below the threshold, is presented. To handle the problem of visibility, only the consecutive scan points, at which the distance between the neighboring points is below some threshold, are taken for the line-segment points.

This paper is organized as follows. In Section 2 the prediction step and the correction step of the EKF are described. Section 3 presents how the lines and their covariances are extracted from the LRF's reflection points. In Section 4 are the results of localizing the robot in an indoor structured environment. The paper is concluded in Section 5.

2 Localization Algorithm

The extended-Kalman-filter approach is adopted here for the purpose of localization. The EKF consists of a prediction and a correction step. The Pioneer 3-AT mobile robot, which has four wheels (Fig. 1), is used to test the localization algorithm. When the robot is rotating about its own axis, the left-hand wheels are rotating in the opposite direction to the right-hand wheels, which is why the wheels are slipping. In the localization algorithm described in this paper a line-based environment description is adopted. As already mentioned, the algorithm can therefore be used [14, 15, 17, 19, 20, 29] indoors and also in outdoor structured environments, which are mainly composed of straight-edged objects or walls.

Fig. 1 Robot's pose according to the global coordinates



2.1 Prediction Step of the EKF

The robot's pose is predicted by simulating the discrete kinematic model of the robot

$$
\begin{aligned}
\mathbf{x}_p(k+1) &= \mathbf{f}(\mathbf{x}_p(k), \mathbf{u}(k)) : \\
x_r(k+1) &= x_r(k) + T\frac{R}{2}(\omega_R(k) + \omega_L(k))\cos(\varphi_r(k)), \\
y_r(k+1) &= y_r(k) + T\frac{R}{2}(\omega_R(k) + \omega_L(k))\sin(\varphi_r(k)), \\
\varphi_r(k+1) &= \varphi_r(k) + T\frac{R}{L}(\omega_R(k) - \omega_L(k)),
\end{aligned} \tag{1}
$$

where the state $\mathbf{x}_p(k) = [x_r(k),\ y_r(k),\ \varphi_r(k)]^T$ denotes the robot's pose with respect to the global coordinates (Fig. 1), T is the sampling time, R denotes the radius of both robot wheels, and L denotes the distance between the wheels. $\mathbf{u}(k) = [\omega_R(k),\ \omega_L(k)]^T$ is the input vector, where $\omega_L(k)$ and $\omega_R(k)$ are measurements of the rotational speed of the left- and right-hand wheels with the encoders at the time kT, respectively. Let $\omega_{Rc}(k)$ and $\omega_{Lc}(k)$ be the rotational speeds of the left- and right-hand wheels, which yields a correct estimation of the robot's pose $\mathbf{x}_p(k+1)$ (Eq. 1). The error for the rotational speeds of the corresponding wheels $\omega_{Rn}(k)$ and $\omega_{Ln}(k)$ can then be defined as

$$
\begin{aligned}
\omega_L(k) &= \omega_{Lc}(k) + \omega_{Ln}(k), \qquad \omega_R(k) = \omega_{Rc}(k) + \omega_{Rn}(k), \\
\mathbf{n}(k) &= [\omega_{Rn}(k),\ \omega_{Ln}(k)]^T.
\end{aligned} \tag{2}
$$

The error vector $\mathbf{n}(k)$ above captures the uncertainties of the odometry model and is assumed to be zero mean and Gaussian noise. The covariance matrix of this vector is the input-noise covariance matrix $\mathbf{Q}(k)$ for the EKF, which is defined in what follows.

The prediction step of the EKF

$$
\begin{aligned}
\tilde{\mathbf{x}}_p(k) &= \mathbf{f}(\hat{\mathbf{x}}_p(k-1), \mathbf{u}(k-1)), \\
\tilde{\mathbf{P}}(k) &= \mathbf{A}(k)\hat{\mathbf{P}}(k-1)\mathbf{A}^T(k) + \mathbf{W}(k)\mathbf{Q}(k-1)\mathbf{W}^T(k),
\end{aligned} \tag{3}
$$

$$
\mathbf{A}_{ij}(k) = \frac{\partial \mathbf{f}_i}{\partial \hat{\mathbf{x}}_{p\ j}(k-1)}\ |_{(\hat{\mathbf{x}}_p(k-1), \mathbf{u}(k-1))}, \qquad \mathbf{W}_{ij}(k) = \frac{\partial \mathbf{f}_i}{\partial \mathbf{n}_j(k-1)}\ |_{(\hat{\mathbf{x}}_p(k-1), \mathbf{u}(k-1))}\ . \tag{4}
$$

$\hat{\mathbf{x}}_p(k-1)$ above denotes the state estimate at time instant $k-1$ based on all the measurements collected up to that time, whereas $\hat{\mathbf{P}}(k-1)$ is the covariance matrix of the corresponding estimation error. $\mathbf{Q}(k-1)$ is the input-noise covariance matrix. $\tilde{\mathbf{x}}_p(k)$ denotes the state prediction and $\tilde{\mathbf{P}}(k)$ denotes the covariance matrix of the state-prediction error.

When driving the robot with some tangential and rotational speed, the error relating to the robot's pose, estimated only by the kinematic (Eq. 1) model (odometry), is accumulating. This is due to the error arising from the inaccurate odometry model 1 and the measurement error of the wheels' encoders. Both errors are treated as the noises of the rotational speeds of the left- and right-hand wheels $\omega_{Rn}(k)$ and $\omega_{Ln}(k)$ (Eq. 2). The standard deviation of the noise of the right-hand wheel's rotational speed $std(\omega_{Rn}(k))$ is modelled as being proportional to the rotational speed of the right-hand wheel $\omega_R(k)$. This results in the variance $var(\omega_{Rn}(k)) = \delta\omega_R^2(k)$, where δ is a constant. Analogously, the variance of the noise of the left-hand wheel's rotational speed is modelled as $var(\omega_{Ln}(k)) = \delta\omega_L^2(k)$. The input-noise covariance matrix $Q(k)$ for the EKF is then defined as

$$\mathbf{Q}(k) = \begin{bmatrix} \delta\omega_R^2(k) & 0 \\ 0 & \delta\omega_L^2(k) \end{bmatrix}. \tag{5}$$

The parameter δ in the covariance matrix above was estimated experimentally, as shown in Section 4.

2.2 Correction Step of the EKF

The robot's pose is corrected by minimizing the difference between the line parameters of the local environment map and the line parameters of the global map, transformed into the robot's coordinates. The global environment map composed of line segments is a-priori known to the robot. After that, the robot builds a local environment map based on the current LRF scan. The local environment map is also composed of line segments.

The global environment map is composed of a set of line segments, described with the edge points and line parameters α_j and p_j $(j = 1, ..., n_G)$ of the line equation in normal form that is based on the global coordinates: $x_G \cos\alpha_j + y_G \sin\alpha_j = p_j$. The line segments of the current environment scan are merged in a local map, and are described with the edge points and the parameters ψ_i and r_i (Fig. 2) of the line equation in normal form, according to the robot's coordinates,

$$x_R \cos\psi_i + y_R \sin\psi_i = r_i. \tag{6}$$

The line segments of the global map, which correspond to the same environment line segments (e.g., a wall) as the line segments of the local map, must be found, as shown in the next section. The matching line parameters ψ_i and r_i from the current local map are collected in the vector $\mathbf{z}(k)$ (Eq. 7), which is used as the input for the correction step of the EKF to update the vehicle's state

$$\mathbf{z}(k) = [r_1, \psi_1, ..., r_N, \psi_N]^T. \tag{7}$$

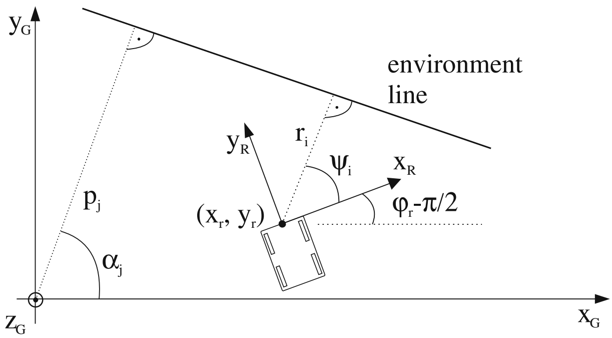


Fig. 2 The line parameters (p_j, α_j) according to the global coordinates, and the line parameters (r_i, ψ_i) according to the robot coordinates

The parameters p_j and α_j of the matched line segment from the global map (according to the global coordinates) are transformed into the parameters $\hat{r}_i$ and $\hat{\psi}_i$ (according to the coordinates of the robot) by

$$C_j = p_j - \tilde{x}_r(k)\cos\alpha_j - \tilde{y}_r(k) sin\alpha_j,$$
$$\begin{bmatrix} \hat{r}_i \\ \hat{\psi}_i \end{bmatrix} = \mu_i(\tilde{\mathbf{x}}_p(k), p_j, \alpha_j) = \begin{bmatrix} |C_j| \\ \alpha_j - (\tilde{\varphi}_r - \frac{\pi}{2}) + (-0.5 \cdot sign(C_j) + 0.5)\pi \end{bmatrix}, \quad (8)$$

where $\tilde{\mathbf{x}}_p(k)$ (Eq. 3) denotes the prediction of the robot's pose and the operator $|.|$ denotes the absolute value. The observation model can then be defined by the vector

$$\mu(\mathbf{x}_p(k)) = \left[\mu_1(\mathbf{x}_p(k), p_1, \alpha_1)^T, ..., \mu_N(\mathbf{x}_p(k), p_N, \alpha_N)^T\right]^T. \quad (9)$$

The correction step of the EKF

$$\hat{\mathbf{x}}_p(k) = \tilde{\mathbf{x}}_p(k) + \mathbf{K}(k)(\mathbf{z}(k) - \mu(\tilde{\mathbf{x}}_p(k))), \quad (10)$$

$$\mathbf{K}(k) = \tilde{\mathbf{P}}(k)\mathbf{H}^T(k)(\mathbf{H}(k)\tilde{\mathbf{P}}(k)\mathbf{H}^T(k) + \mathbf{R}(k))^{-1},$$
$$\hat{\mathbf{P}}(k) = \tilde{\mathbf{P}}(k) - \mathbf{K}(k)\mathbf{H}(k)\tilde{\mathbf{P}}(k), \quad \mathbf{H}_{ij}(k) = \frac{\partial \mu_i}{\partial \tilde{\mathbf{x}}_{p\ j}(k)} |_{\tilde{\mathbf{x}}_{p\ j}(k)}. \quad (11)$$

$\hat{\mathbf{x}}_p(k)$ above denotes the state correction and $\hat{\mathbf{P}}(k)$ denotes the covariance matrix of the state-correction error.

When applying the EKF, the noise arising from the LRF's distance and angle measurements affects the line parameters $\mathbf{z}(k) = [r_1, \psi_1, ..., r_N, \psi_N]^T$ of the local map. The covariance matrix of the vector $\mathbf{z}(k)$ is the output-noise covariance matrix $\mathbf{R}(k)$ (Eq. 11) of the EKF and has, as in [14], a block-diagonal structure, where the $i - th$ block

$$\mathbf{R}_i(k) = \begin{bmatrix} var(r_i) & cov(r_i,\ \psi_i) \\ cov(\psi_i\ , r_i) & var(\psi_i) \end{bmatrix}, \quad (12)$$

represents the covariance matrix of the line parameters $(r_i,\ \psi_i)$. In the following the formulas to define this covariance matrix are presented.

2.3 Matching the Line Segments from the Local and Global Maps

To perform the correction step of the EKF, the line segments of the global map and the line segments of the local map, which correspond to the same environment line segments (e.g., a wall), must be found. Some matching approaches can be found in [1, 14, 20], while here the matched line segments are found as follows. Each line segment of the local map is compared to all of the line segments of the global map transformed into the robot's coordinates according to the prediction of the robot's pose $\tilde{\mathbf{x}}_p(k)$ (Eq. 3). The line segment L of the local map has the parameters $(r,\ \psi)$ and the line segments G_j $(j = 1, ..., n_G)$ of the global map, which are transformed into the robot's coordinates, have the parameters $(\hat{r}_j,\ \hat{\psi}_j)$. The global line segment

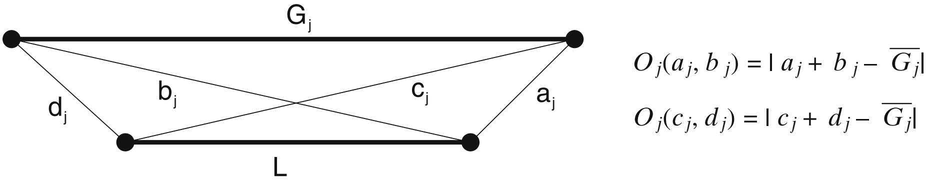

G_j
b_j
c_j
d_j
a_j
L
$O_j(a_j, b_j) = | a_j + b_j - \overline{G_j}|$
$O_j(c_j, d_j) = | c_j + d_j - \overline{G_j}|$

Fig. 3 Defining the overlapping rate $O_j(a_j, b_j)$ and $O_j(c_j, d_j)$ between the local line segment L and the global line segment transformed into the robot's coordinates G_j, where $\overline{G_j}$ is the length of a line segment G_j

G_S $(j = S)$, which is the most similar to the local line segment L, satisfies the following criteria

$$((r - \hat{r}_S)^2 < (r - \hat{r}_j)^2) \; and \; ((\psi - \hat{\psi}_S)^2 < (\psi - \hat{\psi}_j)^2) \; and$$
$$(O_S(a_S, b_S) < O_j(a_j, b_j)) \; and \; (O_S(c_S, d_S) < O_j(c_j, d_j)),$$
$$j = 1, ..., S - 1, S + 1, ..., n_G, \tag{13}$$

where *and* is the Boolean operator. $O_j(a_j, b_j)$ and $O_j(c_j, d_j)$ together represent the overlapping rate between the local line segment L and the transformed, global line segments G_j. a_j, b_j, c_j and d_j (Fig. 3) are Euclidean distances between the edge points of the line segments L and G_j and the overlapping function $O_j(.,.)$ is defined as

$$O_j(a_j, b_j) = |a_j + b_j - \overline{G_j}|, \quad j = 1, ..., n_G, \tag{14}$$

where $\overline{G_j}$ denotes the length of a line segment G_j. If the overlapping rate between the local line segment L and the most similar global line segment G_S is below the threshold values

$$((r - \hat{r}_S)^2 < T_r) \; and \; ((\psi - \hat{\psi}_S)^2 < T_\psi) \; and$$
$$(O_S(a_S, b_S) < T) \; and \; (O_S(c_S, d_S) < T), \tag{15}$$

the line segments are paired. T_r,T_ψ and T above are the thresholds.

3 Identification of the Line Parameters

The line segments are extracted from the reflection points of the laser rangefinder. The LRF in each environment scan (Fig. 4) gives the set of distances $\mathbf{d}_s = [d_{s0^\circ}, ..., d_{s180^\circ}]$ to the obstacles (e.g., a wall) at the angles $\theta_s = [0^\circ, ..., 180^\circ]$. Figure 4 outlines the problem of visibility. The distances between the consecutive reflection points (D_A in Fig. 4), for which the laser beams are close to being perpendicular to the observed environment line segment, are very small (few centimeters). In contrast, the distances between the consecutive reflection points (D_B in Fig. 4) are very large (a few meters and more), if the corresponding laser beams are close to being parallel to the observed environment line segment. Since the reflection points lying on the same environment line are far away from their neighboring points, there is a large part of unobserved environment between them. There could, for example, be the beginning of a wide corridor (Fig. 4). Therefore, only the consecutive points for which

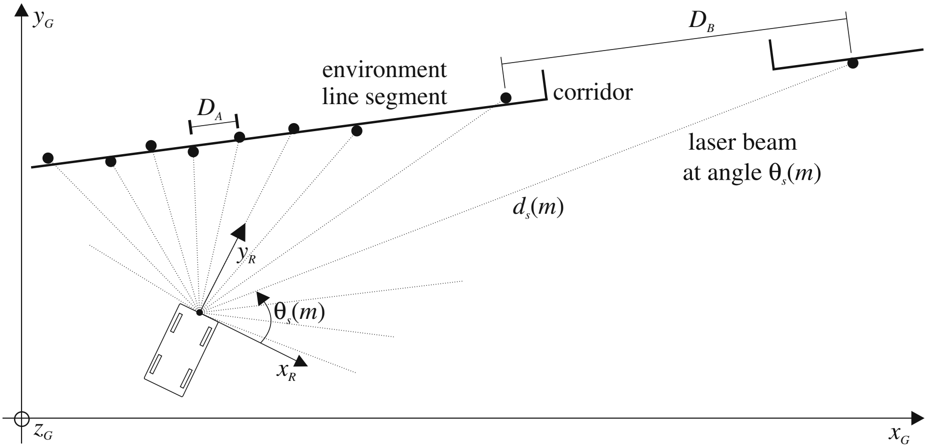


Fig. 4 Reflection points between the laser-beam lines and the environment line segment (wall). The distances between the consecutive reflection points (D_B) can be very large, if the corresponding laser beams are close to being parallel to the observed environment line segment (the problem of visibility)

the distance between the neighboring points is small enough are taken for the line-segment points, as shown in what follows. All the consecutive points

$$\mathbf{x}_{scan}(m) = \mathbf{d}_s(m)\cos\theta_s(m), \quad \mathbf{y}_{scan}(m) = \mathbf{d}_s(m)\sin\theta_s(m), \quad m = 1, ..., Np, \qquad (16)$$

for which the reflections have occurred (e.g., $\mathbf{d}_s(m) \leq R_{LRF}$) are clustered; the other points ($\mathbf{d}_s(m) > R_{LRF}$) are ignored, where R_{LRF} denotes the range of a LRF. Each cluster is then split into more clusters if the distance between two consecutive points is greater than the threshold T_S. This threshold is, in a particular environment, set with regard to the expected smallest distance required to distinguish between two different consecutive line segments. The problem of visibility described above is also taken into account here, since each of the consecutive points that lie on the same environment line with the distance to its neighboring point being greater than the threshold T_S forms a single cluster. Clusters with only one reflection point are eliminated in the following step. If there are fewer than N_{min} points in a cluster, the cluster is ignored, where N_{min} is the minimum number of reflection points required to define a line segment. Due to the LRF's noise, more than 2 points must be taken into consideration to reliably model the environment line segment with the calculated line segment. Each cluster is then split with the split-and-merge algorithm [19] in the consecutive sets of reflection points, where each set of points ($\mathbf{x}$, $\mathbf{y}$) corresponds to some environment line segment. If a certain set of line-segment points ($\mathbf{x}$, $\mathbf{y}$) is composed of fewer than N_{min} points, the set is again discarded. The split-and-merge algorithm is, as shown in [19], fast and has good correctness. In addition to the line parameters it also gives the edge points of the line segments. The line parameters are very often computed using the Hough transform [13, 16, 20, 23]. However, this algorithm is computationally more expensive and the result of the Hough transform does not include the edge points of the line segments, which is important information for localization and map building.

The classic LSQ method is used for the line fitting. If the set of points $(\mathbf{x},\ \mathbf{y})$ belongs to a vertical line segment, the line parameters cannot be computed in the least-square sense directly. The reason is that the result of the least-squares method is the parameters of an explicit line equation. In this form the vertical line causes the estimated slope-parameter to go to infinity. To obtain the best fit of the classic-LSQ-estimated line parameters to the given set of data $(\mathbf{x},\ \mathbf{y})$, the slope of the line is estimated first. If the absolute value of the slope estimated from the edge points of the set $(\mathbf{x}, \mathbf{y})$ is greater than 1, all the points are rotated by an angle of $-\frac{\pi}{2}$ to have a well-conditioned LSQ estimation problem. This is done by exchanging the vector $\mathbf{x}$ with the vector $\mathbf{y}$, and the vector $\mathbf{y}$ with the vector $-\mathbf{x}$. The set of line-segment points $(\mathbf{x}, \mathbf{y})$ is then reduced to the parameters r and ψ of the line equation in normal form according to the robot's coordinates (Eq. 6) as follows

$$\hat{\boldsymbol{\theta}} = [\hat{k}_l, \hat{c}_l]^T = (\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{y}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{x}(1) \ \dots \ \mathbf{x}(n) \\ 1 \quad \dots \quad 1 \end{bmatrix}^T, \quad \mathbf{y} = [\mathbf{y}(1), \dots, \mathbf{y}(n)]^T, \quad (17)$$

$$r(\hat{k}_l, \hat{c}_l) = \frac{\hat{c}_l}{\sqrt{\hat{k}_l^2 + 1}} sign(\hat{c}_l), \quad \psi(\hat{k}_l) = arctan2\left(\frac{sign(\hat{c}_l)}{\sqrt{\hat{k}_l^2 + 1}}, \frac{-\hat{k}_l}{\sqrt{\hat{k}_l^2 + 1}} sign(\hat{c}_l)\right), \quad (18)$$

where n denotes the number of line-segment points. $\hat{k}_l$ and $\hat{c}_l$ are parameters of the explicit line equation $y_R = k_l \cdot x_R + c_l$, which are estimated with the classic LSQ method. Both parameters are converted into the parameters $r(\hat{k}_l, \hat{c}_l)$ and $\psi(\hat{k}_l)$ (Eq. 18) of the line equation in normal form, where the function *arctan*2 is a four-quadrant arctan function. If the line-segment points $(\mathbf{x},\ \mathbf{y})$ are rotated by $-\frac{\pi}{2}$, the angle $\frac{\pi}{2}$ must be added to the calculated angle ψ (Eq. 18) to get the right line parameter. The line parameter r (Fig. 2) is invariant to the rotation of the line-segment points and therefore remains unchanged.

In the literature the normal-line-equation parameters r and ψ (Eq. 6) are very often computed using the orthogonal least-squares method [14]

$$\begin{bmatrix} r^* \\ \psi^* \end{bmatrix} = \begin{bmatrix} \bar{x}cos(\psi^*) + \bar{y}sin(\psi^*) \\ \frac{1}{2}arctan(\frac{-2S_{xy}}{S_{y^2} - S_{x^2}}) \end{bmatrix} \triangleq \begin{bmatrix} f_1(\mathbf{x}(1), \mathbf{y}(1), \dots, \mathbf{x}(n), \mathbf{y}(n)) \\ f_2(\mathbf{x}(1), \mathbf{y}(1), \dots, \mathbf{x}(n), \mathbf{y}(n)) \end{bmatrix}. \quad (19)$$

$$\bar{x} = \frac{\sum_{j=1}^n \mathbf{x}(j)}{n}, \quad \bar{y} = \frac{\sum_{j=1}^n \mathbf{y}(j)}{n}, \quad S_{x^2} = \sum_{j=1}^n (\mathbf{x}(j) - \bar{x})^2,$$
$$S_{y^2} = \sum_{j=1}^n (\mathbf{y}(j) - \bar{y})^2, \quad S_{xy} = \sum_{j=1}^n (\mathbf{x}(j) - \bar{x})(\mathbf{y}(j) - \bar{y}). \quad (20)$$

3.1 Estimation of Line Parameters' Covariances

The variances of the line parameters r_i and ψ_i (Eq. 18) and the covariances between them, which comprise the covariance matrix $\mathbf{R}_i$ (Eq. 12) of vector $[r_i,\ \psi_i]$ must also be computed in order to perform the correction step of the EKF. The variances and covariances must be taken into account, since the noise of the LRF's readings $\mathbf{d}_s(m)$ and $\theta_s(m)$ affects both of the extracted line parameters. A method for estimating the line parameters' covariances resulting from the classic LSQ, which is proposed in our previous work [24], is briefly outlined here. According to the least-squares theory the covariance matrix of the line parameters' vector $[\hat{k}_l,\ \hat{k}_l]$ (17) is calculated first by

$$\mathbf{Ce} = var(\mathbf{y}(j))(\mathbf{U}^T\mathbf{U})^{-1} = \begin{bmatrix} var(\hat{k}_l) & cov(\hat{k}_l, \hat{c}_l) \\ cov(\hat{c}_l, \hat{k}_l) & var(\hat{c}_l) \end{bmatrix}, \tag{21}$$

$$var(\mathbf{y}(j)) = \frac{\sum_{j=1}^{n}(\mathbf{y}(j) - \hat{\mathbf{y}}(j))^2}{n-1}, \qquad \hat{\mathbf{y}}(j) = \hat{k}_l \cdot \mathbf{x}(j) + \hat{c}_l, \tag{22}$$

where $var(\mathbf{y}(j))$ is the vertical-error variance of the line-segment points $(\mathbf{x}(j),\ \mathbf{y}(j))$ $(j = 1,\ ..., n)$ according to the estimated line with the parameters $\hat{k}_l$ and $\hat{c}_l$. Knowing the variances and covariances between the parameters $\hat{k}_l$ and $\hat{c}_l$ the variances and covariances between the parameters r and ψ are calculated as follows

$$\begin{aligned} var(\psi) &= K_{\psi k}^2 var(\hat{k}_l), \\ var(r) &= K_{rk}^2 var(\hat{k}_l) + K_{rc}^2 var(\hat{c}_l) + 2K_{rk}K_{rc} \cdot cov(\hat{k}_l, \hat{c}_l), \\ cov(r, \psi) &= K_{rk}K_{\psi k} var(\hat{k}_l) + K_{rc}K_{\psi k} \cdot cov(\hat{k}_l, \hat{c}_l), \\ cov(\psi, r) &= cov(r, \psi), \end{aligned} \tag{23}$$

$$K_{rk} = \frac{-\hat{c}_l\hat{k}_l}{\sqrt{\hat{k}_l^2 + 1}(\hat{k}_l^2 + 1)} sign(\hat{c}_l), \qquad K_{rc} = \frac{sign(\hat{c}_l)}{\sqrt{\hat{k}_l^2 + 1}}, \qquad K_{\psi k} = \frac{1}{\hat{k}_l^2 + 1}. \tag{24}$$

If the line-segment points $(\mathbf{x},\ \mathbf{y})$ are rotated by $-\frac{\pi}{2}$, the line-parameter r (Fig. 2) remains unchanged. Consequently, the variance of parameter r (Eq. 18) also remains unchanged and is equal to the already-calculated $var(r)$ in Eq. 23. The variance of the true angle $var(\psi + \frac{\pi}{2})$ and the covariance $covar(r,\ \psi + \frac{\pi}{2})$ are also equal to the already calculated variance $var(\psi)$ and covariance $covar(r,\ \psi)$ (Eq. 23), respectively.

If the normal-line-equation parameters r and ψ (Eq. 6) are computed by the orthogonal LSQ method (Eqs. 19 and 20), then the covariance matrix $\mathbf{R}_i$ (Eq. 12) of the vector $[r_i,\ \psi_i]$ can be, according to reference [14], calculated as follows

$$\mathbf{C}^* = \sum_{j=1}^{n}(A_jB_j)C_{m_j}(A_jB_j)', \quad \mathbf{A}_j = \begin{bmatrix} \dfrac{\partial f_1}{\partial \mathbf{x}(j)} & \dfrac{\partial f_1}{\partial \mathbf{y}(j)} \\ \dfrac{\partial f_2}{\partial \mathbf{x}(j)} & \dfrac{\partial f_2}{\partial \mathbf{y}(j)} \end{bmatrix}, \tag{25}$$

$$\mathbf{B}_j = \begin{bmatrix} \cos\theta_s(j) & -\mathbf{d}_s(j)\sin\theta_s(j) \\ \sin\theta_s(j) & \mathbf{d}_s(j)\cos\theta_s(j) \end{bmatrix}, \mathbf{C}_{m_j} = \begin{bmatrix} \sigma_{d_j}^2 & \sigma_{d_j\theta_j} \\ \sigma_{\theta_j d_j} & \sigma_{\theta_j}^2 \end{bmatrix}. \tag{26}$$

$\mathbf{C}_{m_j}$ $(j = 1, ..., n)$ above is the covariance matrix of a line-segment point in the polar coordinates $[\mathbf{d}_s(j), \theta_s(j)]$, which is a raw LRF-sensor measurement. $\mathbf{A}_j$ is the Jacobian matrix of $\mathbf{f} = [f_1,\ f_2]$ (Eq. 19) according to the cartesian coordinates, of which the partial derivatives are shown in [14].

In Section 4 the experimental results of the localization algorithm, where the line parameters' covariances are estimated with the method resulting from the classic LSQ (Eq. 23), are presented.

3.2 Estimation of the Computational Complexity

Localization is a real-time process. Therefore, the computational efficiency of the algorithms involved in solving the localization problem is of great importance. In the localization algorithm described in this paper the vectors of the two line parameters $[r_i,\ \psi_i]^T$ (Fig. 2) and their covariance matrices must be calculated in each environment scan made with a LRF sensor for all the observed line segments. The method for estimating the covariances of the observed line parameters resulting from the classic LSQ method [24] and the method resulting from the orthogonal LSQ [14] were briefly outlined in the previous subsection. In our previous work [24] the computational complexities of both methods were compared with each other. It is demonstrated that the use of the classic LSQ instead of the orthogonal LSQ reduces the number of computations in the process of estimating the two normal line-equation parameters and their covariance matrix.

The computational complexity of both methods is analyzed by counting up all the elementary mathematical operations involved in the calculation of the line parameters r and ψ (Eq. 6) and their covariance matrix $\mathbf{R}_i$ (Eq. 12). The computational costs of the method resulting from the orthogonal LSQ in the noise cases with the nonzero and zero LRF's angular variance $\sigma^2_{\theta_j}$ (Eq. 26) are $C_{ols1}(n) = 69n + 36$ and $C_{ols2}(n) = 58n + 36$, respectively. The computational complexity depends on the number of line-segment points n. The computational costs of the method resulting from the classic LSQ are $C_{ls}(n) = 12n + 52$. The computational complexities of both methods are compared relative to each other by

$$C_{r1}(n) = \frac{C_{ols1}(n)}{C_{ls}(n)} = \frac{69n + 36}{12n + 52}, \qquad C_{r2}(n) = \frac{C_{ols2}(n)}{C_{ls}(n)} = \frac{58n + 36}{12n + 52}, \tag{27}$$

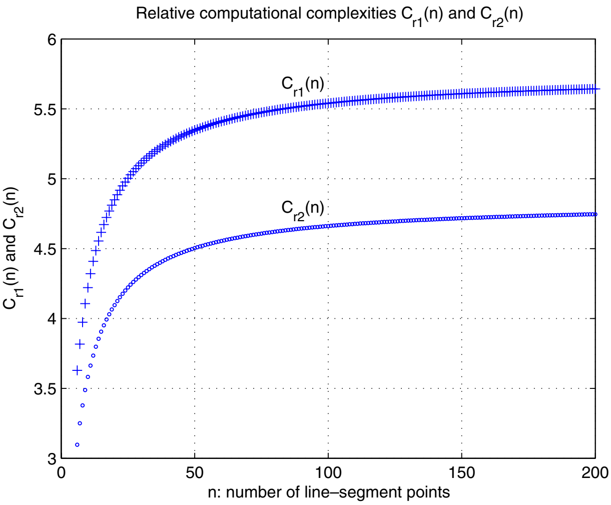


Fig. 5 The relative computational complexities $C_{r1}(n) = \frac{C_{ols1}(n)}{C_{ls}(n)}$ and $C_{r2}(n) = \frac{C_{ols2}(n)}{C_{ls}(n)}$ as a function of the number of line-segment points $n \geq 6$

where the relative computational complexities $C_{r1}(n)$ and $C_{r2}(n)$ refer to the noise cases with the nonzero and zero LRF's angular variance $\sigma^2_{\theta_j}$ (Eq. 26), respectively. Figure 5 shows the relative computational complexities $C_{r1}(n)$ and $C_{r2}(n)$ as a function of the number of line-segment points $n \geq 6$. If the line parameters and their covariance matrix are calculated from 8 to 200 points, the method resulting from the classic LSQ has, in the noise case with the nonzero or zero LRF's angular variance, about 4 to 5.6 or 3.4 to 4.7 times fewer operations than the method resulting from the orthogonal LSQ, respectively.

4 Experimental Results

In order to perform the correction step of the EKF the covariances of the line parameters must be estimated in each environment scan made with a LRF sensor

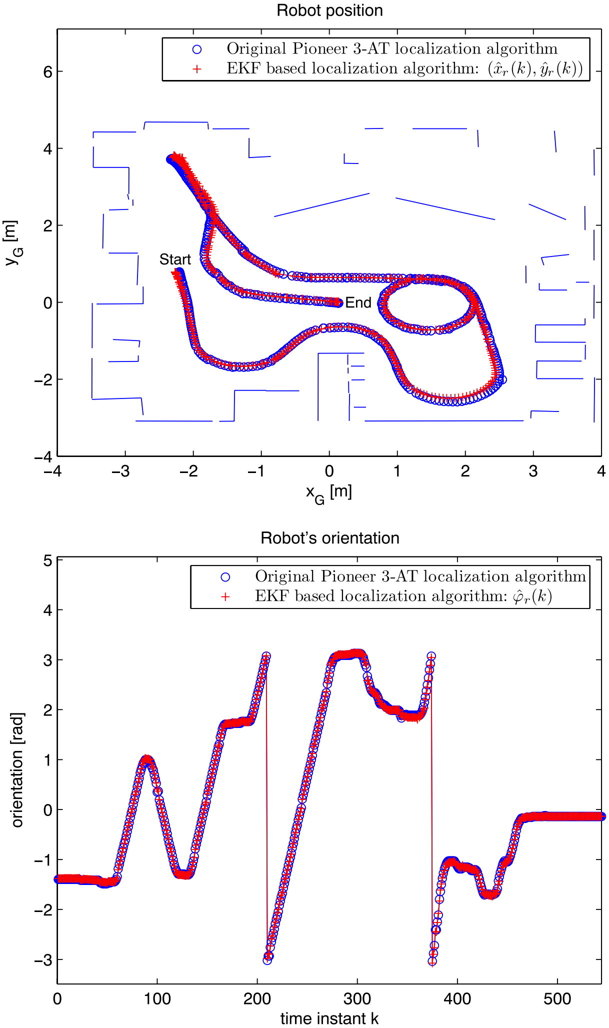


Fig. 6 **a** Robot's position $(\hat{x}_r(k),\ \hat{y}_r(k))$ estimated with the EKF compared to the position estimated with the original Pioneer 3-AT localization algorithm. The line segments represent the global map of a room. **b** Robot's orientation $\hat{\varphi}_r(k)$ estimated with the EKF compared to the orientation estimated with the original Pioneer 3-AT localization algorithm

for all the observed line segments. In this section the method (Eqs. 18 and 23) for calculating the normal-line-equation parameters $[r_i,\ \psi_i]$ (Eq. 6) and their covariance matrices $\mathbf{R}_i$ (Eq. 12), which results from the classic LSQ, is tested within the EKF-based localization of a real mobile robot. A Pioneer 3-AT mobile robot, which is equipped with encoders for the wheels, a Sick LMS200 laser rangefinder and a laptop computer, was used for the experiments. The localization algorithm is implemented in the C++ environment.

The parameters needed to perform the prediction step of the EKF are defined as follows. The sampling time in the prediction step of the EKF (Eqs. 1 and 3) is $T = 100$ ms. The parameters R and L (Fig. 1) from the kinematic model 1 were experimentally identified as the values $R = 10.9$ cm and $L = 58.7$ cm. The parameter

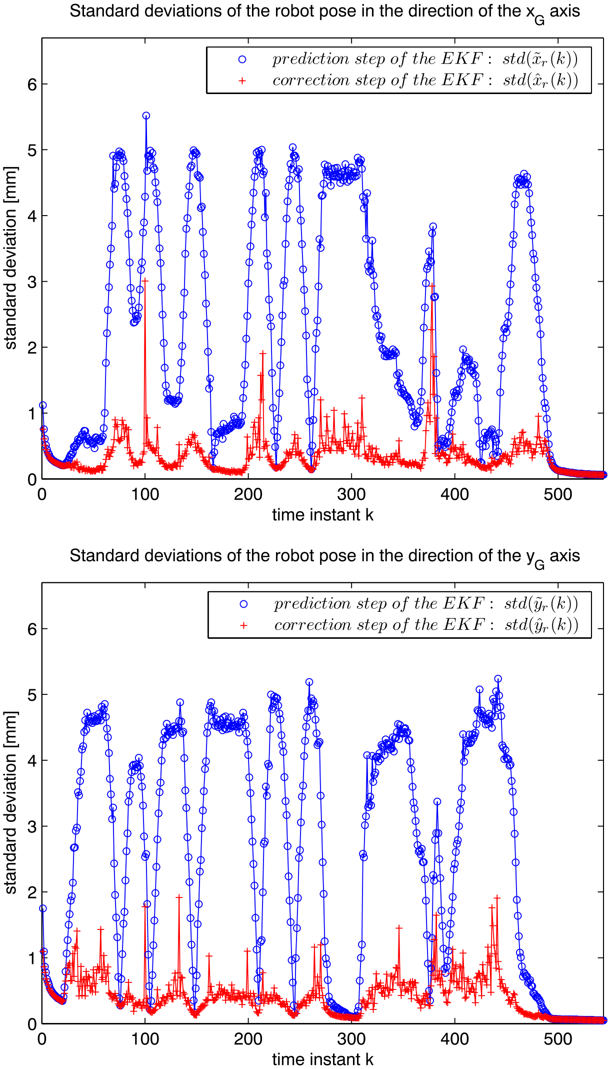
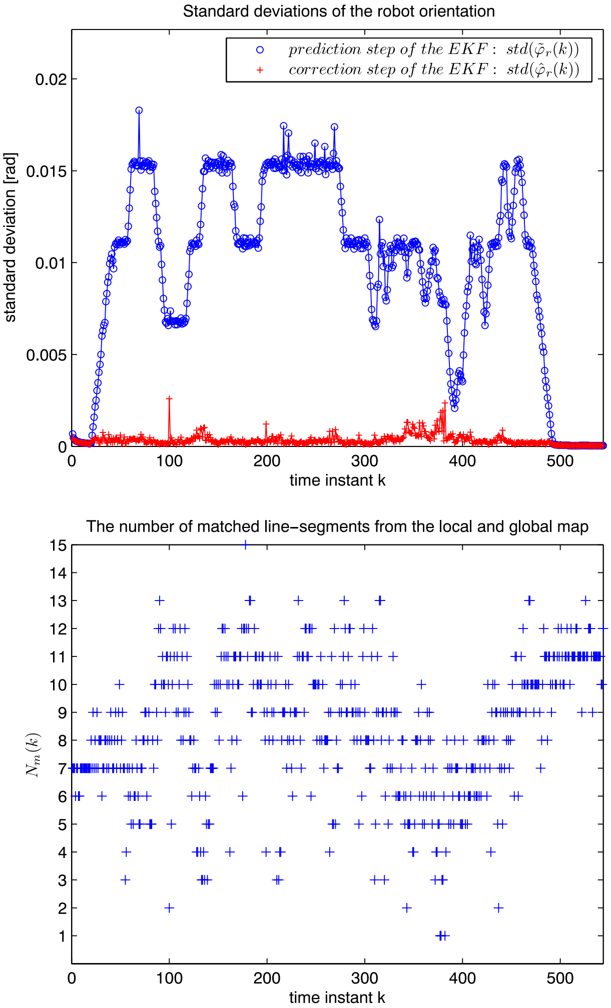


Fig. 7 **a** Standard deviation of the predicted and corrected robot's pose in the direction of the x_G axis $std(\tilde{x}_r(k))$ and $std(\hat{x}_r(k))$, respectively. **b** Standard deviation of the predicted and corrected robot's pose in the direction of the y_G axis $std(\tilde{y}_r(k))$ and $std(\hat{y}_r(k))$, respectively

Fig. 8 **a** Standard deviation of the predicted and corrected robot's orientation $std(\tilde{\varphi}_r(k))$ and $std(\hat{\varphi}_r(k))$, respectively. **b** The number of matched line segments from the local and global maps $N_m(k)$ during the experiment



δ in the input-noise covariance matrix $Q(k)$ (Eq. 5) of the EKF was also estimated experimentally as follows. The differences between the true robot position and the position estimated by the kinematic model 1 when driving the robot straight forwards several times (from the minimum to the maximum tangential speed of the robot) were observed. The differences between the true robot orientation and the orientation estimated by the kinematic model when rotating the robot around its own axis several times (from the minimum to the maximum angular speed of the robot) were also observed. According to the error in the position and the orientation from the experiments, the parameter δ was calculated and set to the value 0.01. The parameters needed to perform the correction step of the EKF are defined as follows. The thresholds (Eq. 15) for finding the corresponding line segments from the local and global maps according to the overlapping rate were heuristically set to the values $T_r = 20$ cm, $T_\psi = \frac{\pi}{6}$ and $T = 40$ cm. The range of the used LRF sensor is $R_{LRF} = 80$ m. The threshold for splitting the clusters of the LRF's reflection points is heuristically set to the value $T_S = 15$ cm. A minimum number of reflection points required to define a line segment is set to the value $N_{min} = 5$.

The localization algorithm is tested in an indoor structured environment (room), which is mainly composed of walls and other straight-edged objects. The global environment map, which is represented with line segments, is shown in Fig. 6a. During the experiment the robot was driven with a tangential speed of approximately $0.64 m/s$. When the robot was turning its angular speed was approximately $48°/s$, and when rotating around its own axis the angular speed was approximately $87°/s$. Figures 6a, b show robot's pose $\hat{\mathbf{x}}_p(k) = [\hat{x}_r(k),\ \hat{y}_r(k),\ \hat{\varphi}_r(k)]^T$ (Eq. 10), estimated with the EKF. This pose is compared to the pose estimated with the original Pioneer 3-AT localization algorithm, and both poses are very close to each other. Figures 7a, b and 8a show, for each time instant k, the standard deviations of the robot's pose prediction error $std(\tilde{x}_r(k))$, $std(\tilde{x}_r(k))$ and $std(\tilde{\varphi}_r(k))$, which are obtained from the corresponding covariance matrix $\tilde{\mathbf{P}}(k)$ (Eq. 3). Figures 7a, b and 8a show also the standard deviations of the robot's pose correction error $std(\hat{x}_r(k))$, $std(\hat{x}_r(k))$ and $std(\hat{\varphi}_r(k))$, which are obtained from the corresponding covariance matrix $\hat{\mathbf{P}}(k)$ (Eq. 11). Figure 8b shows that the number of matched line segments from the local and the global maps $N_m(k)$ was quite large during the experiment, which is important to achieve good convergence properties of the EKF. On average, a value of around $N_m(k) = 8$ was achieved. The minimum value $N_m(k) = 1$ was only achieved three times. The maximum value was $N_m(k) = 15$.

5 Conclusion

In this paper an EKF-based localization of a four-wheeled mobile robot equipped with encoders for the wheels and a LRF sensor is presented. The prediction step of the EKF is performed by simulating the kinematic model of the robot. The standard deviation of each robot-wheel's angular speed is, in the input-noise covariance matrix, estimated as being proportional to the wheel's angular speed. In order to perform the correction step of the EKF the covariances for all the observed environment lines must be estimated in each environment scan made with a LRF sensor. A method for estimating the line parameters' covariances resulting from the classic LSQ was tested within the localization experiment in an indoor structured environment. The good localization results prove the applicability of this method for the purposes of the EKF-based localization of a mobile robot. The line segments from the local and global maps are paired if the overlapping rate between the most similar line segments is below the threshold. The number of matched line segments is quite large, which is important to achieve good convergence properties of the EKF. The robot's pose, estimated by the EKF, was compared to the pose estimated by the original Pioneer-3AT localization algorithm, and both poses are very close to each other. The paper was focused on solving the problem of mobile-robot localization and in the experimental results a good convergence of the EKF was achieved. In future work the localization algorithm will be extended into the SLAM algorithm.

References

1. Anousaki, G.C., Kyriakopoulos, K.J.: Simultaneous localization and map building of skid-steered robots. IEEE Robot. Autom. Mag. **14**(1), 79–89 (2007)
2. Arras, K.O., Siegwart, R.Y.: Feature extraction and scene interpretation for map-based navigation and map building. In: Proceedings of SPIE, Mobile Robotics XII, vol. 3210, pp. 42–53 (1997)
3. Bailey, T., Durrant-Whyte, H.: Simultaneous localization and mapping (SLAM): part II. IEEE Robot. Autom. Mag. **13**(3), 108–117 (2006)
4. Baltzakis, H., Trahanias, P.: Hybrid mobile robot localization using switching state-space models. In: IEEE International Conference on Robotics and Automation, 2002. Proceedings. ICRA '02, pp. 366–373 (2002)
5. Blažič, S., Škrjanc, I., Gerkšič, S., Dolanc, G., Strmčnik, S., Hadjiski, M.B., Stathaki, A.: Online fuzzy identification for an intelligent controller based on a simple platform. Eng. Appl. Artif. Intell. **22**(4-5), 628–638 (2009)
6. Blažič, S., Škrjanc, I., Matko, D.: Globally stable direct fuzzy model reference adaptive control. Fuzzy Sets Syst. **139**(1), 3–33 (2003)
7. Borges, G.A., Aldon, M.-J.: A split-and-merge segmentation algorithm for line extraction in 2-D range images. In: Proceedings of the International Conference on Pattern Recognition, vol. 1, pp. 1441 (2000)
8. Borges, G.A., Aldon, M.-J.: Line extraction in 2D range images for mobile robotics. J. Intell. Robot. Syst. **40**(3), 267–297 (2004)
9. Choi, Y.-H., Lee, T.-K., Oh, S.-Y.: A line feature based SLAM with low grade range sensors using geometric constraints and active exploration for mobile robot. Auton. Robots **24**(1), 13–27 (2008)
10. Crowley, J.L., Wallner, F., Schiele, B.: Position estimation using principal components of range data. In: 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings, pp. 3121–3128 (1998)
11. Diosi, A., Kleeman, L.: Laser scan matching in polar coordinates with application to SLAM. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005. (IROS 2005), pp. 3317–3322 (2005)
12. Durrant-Whyte, H., Bailey, T.: Simultaneous localization and mapping: part I. IEEE Robot. Autom. Mag. **13**(2), 99–110 (2006)
13. Forsberg, J., Larsson, U., Ahman, P., Wernersson, A.: The Hough transform inside the feedback loop of a mobile robot. In: IEEE International Conference on Robotics and Automation, 1993. Proceedings, vol. 1, pp. 791–798 (1993)
14. Garulli, A., Giannitrapani, A., Rossi, A., Vicino, A.: Mobile robot SLAM for line-based environment representation. In: 44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05, pp. 2041–2046 (2005)
15. Garulli, A., Giannitrapani, A., Rossi, A., Vicino, A.: Simultaneous localization and map building using linear features. In: Proceedings of the 2nd European Conference on Mobile Robots, pp. 44–49 (2005)
16. Giesler, B., Graf, R., Dillmann, R., Weiman, C.F.R.: Fast mapping using the log-Hough transformation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 1998. Proceedings, vol. 3, pp. 1702–1707 (1998)
17. Jensfelt, P., Christensen, H.I.: Pose tracking using laser scanning and minimalistic environmental models. IEEE Trans. Robot. Autom., **17**(2), 138–147 (2001)
18. Latecki, L.J., Lakaemper, R., Sun, X, Wolter, D.: Building polygonal maps from laser range data. In: ECAI Int. Cognitive Robotics Workshop, Valencia, Spain, August 2004 (2004)
19. Nguyen, V., Martinelli, A., Tomatis, N., Siegwart, R.: A comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005. (IROS 2005), pp. 1929–1934 (2005)
20. Pfister, S.T., Roumeliotis, S.I., Burdick, J.W.: Weighted line fitting algorithms for mobile robot map building and efficient data representation. In: IEEE International Conference on Robotics and Automation, 2003. Proceedings. ICRA '03, vol. 1, pp. 1304–1311 (2003)
21. Pozna, C., Troester, F., Precup, R.-E., Tar, J.K., Preitl, S.: On the design of an obstacle avoiding trajectory: method and simulation. Math. Comput. Simul. **79**(7), 2211–2226 (2009)
22. Rofer, T.: Using histogram correlation to create consistent laser scan maps. In: IEEE/RSJ International Conference on Intelligent Robots and System, 2002, vol. 1, pp. 625–630 (2002)
23. Schiele, B., Crowley, J.L.: A comparison of position estimation techniques using occupancy grids. In: IEEE Conference on Robotics and Autonomous Systems, 1994, (ICRA 94) (1994)

24. Teslić, L., Škrjanc, I., Klančar, G.: Using a LRF sensor in the Kalman-filtering-based localization of a mobile robot. ISA Trans. **49**(1), 145–153 (2010)
25. Thrun, S.: Robotic mapping: a survey. In: Lakemeyer, G., Nebel, B. (eds.) Exploring Artificial Intelligence in the New Millennium. Morgan Kaufmann, San Francisco (2002)
26. Tomatis, N., Nourbakhsh, I., Siegwart, R.: Hybrid simultaneous localization and map building: a natural integration of topological and metric. Robot. Auton. Syst. **44**(1), 3–14 (2003)
27. Veeck, M., Veeck, W.: Learning polyline maps from range scan data acquired with mobile robots. In: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings, vol. 2, pp. 1065–1070 (2004)
28. Yan, Z., Shubo, T., Lei, L., Wei, W.: Mobile robot indoor map building and pose tracking using laser scanning. In: International Conference on Intelligent Mechatronics and Automation, 2004. Proceedings, pp. 656–661 (2004)
29. Yaqub, T., Tordon, M.J., Katupitiya, J.: Line segment based scan matching for concurrent mapping and localization of a mobile robot. In: 9th International Conference on Control, Automation, Robotics and Vision, 2006. (ICARCV '06), pp. 1–6 (2006)
30. Zhang, X., Rad, A.B., Wong, Y.-K.: A robust regression model for simultaneous localization and mapping in autonomous mobile robot. J. Intell. Robot. Syst. **53**(2), 183–202 (2008)