ISA Transactions 49 (2010) 145–153

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Using a LRF sensor in the Kalman-filtering-based localization of a mobile robot

Luka Teslić*, Igor Škrjanc, Gregor Klančar

Faculty of Electrical Engineering, University of Ljubljana, LMSV&LAMS, Tržaška 25, 1000 Ljubljana, Slovenia

ARTICLE INFO

Article history:
Received 22 May 2009
Received in revised form
23 September 2009
Accepted 28 September 2009
Available online 13 October 2009

Keywords:
Mobile robot
Localization
SLAM
Kalman Filter
Covariance matrix
Least-squares method

ABSTRACT

This paper deals with the problem of estimating the output-noise covariance matrix that is involved in the localization of a mobile robot. The extended Kalman filter (EKF) is used to localize the mobile robot with a laser range finder (LRF) sensor in an environment described with line segments. The covariances of the observed environment lines, which compose the output-noise covariance matrix in the correction step of the EKF, are the result of the noise arising from a range-sensor's (e.g., a LRF) distance and angle measurements. A method for estimating the covariances of the line parameters based on classic least squares (LSQ) is proposed. This method is compared with the method resulting from the orthogonal LSQ in terms of computational complexity. The results of a comparison show that the use of classic LSQ instead of orthogonal LSQ reduce the number of computations in a localization algorithm which is a part of a SLAM (simultaneous localization and mapping) algorithm. Statistical accuracy of both methods is also compared by simulating the LRF's measurements and the comparison proves the efficiency of the proposed approach.

1. Introduction

The problem of robotic mapping and localization has been widely studied. Here, a robot must know its own pose (the localization problem) in order to build a map, and the robot also needs to know the environment map (the mapping problem) to localize itself to its current pose. The robot can localize itself using odometric measurements and by comparing the local map, obtained from the current view of the robot, with an already-built global environment map. The problems of mapping and localization can be handled separately if the robot's pose is given to the robot by a human or by using GPS (global positioning system) and INS (Inertial Navigation System) sensors (outdoor environments) when map building. The map of the environment can then be used to solve the localization problem. To avoid assuming that the robot's pose is known a SLAM (simultaneous localization and mapping) algorithm must be built, where the problems of localization and mapping are merged. A SLAM algorithm is known as a computationally very complex [1] operation. To solve the SLAM problem a joint state composed of a robot's pose and the locations of observed stationary landmarks (e.g., line segments) must be estimated. The observation-update step requires that all the landmarks and the joint covariance matrix be updated every time an observation is made [2]. This means that the extent of the computation expands quadratically with the number of landmarks in a map [2]. Many approaches [1] have been developed to reduce this complexity and the associated computation. In [3] a comprehensive survey of the SLAM problem is presented, and in the literature, many approaches and algorithms involved in solving the SLAM, localization and mapping problem have been proposed [3–12]. In [13] a comparison of line-extraction algorithms using a 2D laser range finder is reported. Based on this comparison a split-and-merge algorithm was chosen in this paper, because of its speed and good correctness.

The Kalman-filtering technique is very often used to solve the localization or SLAM problem. The convergence properties of the Kalman filter and therefore the SLAM algorithm significantly depend on setting the process' input- and output-noise covariance matrices. These matrices have to be appropriately set. In our previous work [14] a Kalman-Filtering-based localization algorithm for a mobile robot with a LRF is presented and tested on a simulator built in Matlab, where the input-noise covariance matrix of the EKF is derived from the known noise variances of the angular velocity measurements of both robot wheels. However, this paper focuses on a derivation of the output-noise covariance matrix of the EKF and a method for estimating the line parameters' covariances is proposed. In an environment described with line segments, covariances of the line-equation parameters compose the output-noise covariance matrix of the EKF. Line segments are often applied for environment representation and in papers [15–21] different methods are used to estimate the covariances of the normal line-equation parameters. In this work a proposed method for estimating these covariances is derived from classic least squares (LSQ) and is computationally more efficient than the method resulting from orthogonal LSQ. If the line parameters and their covariance matrix are calculated from 8 to 200 points, the method resulting from classic LSQ has in the noise case with nonzero or zero LRF's

* Corresponding author. Tel.: +386 1 4768702; fax: +386 1 4264631.
E-mail address: luka.teslic@fe.uni-lj.si (L. Teslić).

0019-0578/$ – see front matter © 2009 ISA. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.isatra.2009.09.009

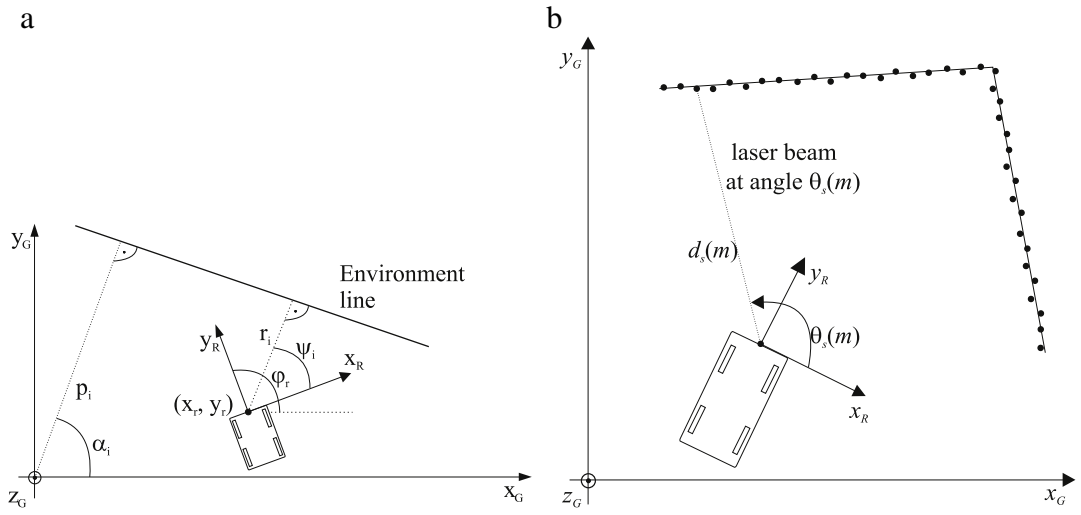

Fig. 1. (a) The line parameters (p_i, α_i) according to the global coordinates, and the line parameters (r_i, ψ_i) according to the robot's coordinates. (b) Reflection point between the laser–beam line and the environment line segment (wall).

angular variance about 4 to 5.6 or 3.4 to 4.7 times fewer operations than the method resulting from orthogonal LSQ, respectively. In the future, the localization algorithm shown in this paper will be extended into the SLAM algorithm, where the same method as proposed in this paper will be used for estimating the line parameters and their covariances. Therefore, in this SLAM algorithm the computational costs of calculating the line parameters and their covariances will be the same as in the localization algorithm shown in this paper. The statistical accuracy of each method is also analyzed and both methods are compared. The accuracy of line parameters' covariances estimated with the method resulting from classic LSQ depends on the number of line-segment points from which these covariances are estimated. The accuracy of the method resulting from orthogonal LSQ depends on the accuracy of the variance of the LRF's distance-measurement error and variance of the laser–beam angle error which must be a priori given from a LRF's noise model. The standard deviations and the covariance of the line parameters estimated with each method are close to the statistically estimated reference standard deviations and covariance in both noise cases (zero and nonzero LRF's angular variance).

This paper is organized as follows. In Section 2 the prediction step and the correction step of the EKF are described first. Then the methods for estimating the line parameters resulting from classic and orthogonal LSQ are presented. Further, the proposed method to estimate the line parameters' covariances resulting from classic LSQ and a method resulting from orthogonal LSQ are presented. Then the computational complexities of both methods are analyzed and compared to each other. In Section 3 the statistical validation of the accuracy of both methods using the simulated measurements of the LRF sensor is performed and the accuracy of both methods is compared. The paper is concluded in Section 4.

2. Estimation of the line parameters and their covariances for the EKF

The extended Kalman filter (EKF) approach, which consists of a prediction and a correction step, is adopted here for the purpose of localization. In the localization algorithm presented in this paper the parameters of the environment lines and their covariances are needed in order to perform the correction step of the EKF. If the localization algorithm shown in this paper is extended into the SLAM algorithm, these line parameters and their covariances must also be computed. The SLAM is according to [2] formulated as follows. SLAM is a process by which a mobile robot can build a map of an environment and at the same time use this map to deduce its location. In SLAM, both the trajectory of the mobile robot and the location of all environment landmarks are estimated online without the need for any a priori knowledge of a robot's pose.

2.1. Prediction and correction step of the EKF

The robot's pose is predicted by simulating the odometry model $\mathbf{f}(.)$

$$\mathbf{x}_p(k+1) = \mathbf{f}(\mathbf{x}_p(k), \mathbf{u}(k)) + \mathbf{n}(k), \tag{1}$$

where the state $\mathbf{x}_p(k) = [x_r(k), \ y_r(k), \ \varphi_r(k)]^{\mathrm{T}}$ denotes the robot pose with respect to the global coordinates (Fig. 1a) and $\mathbf{u}(k)$ denotes the input vector to the odometry model (e.g., velocities of the robot's left- and right-hand wheel). Vector $\mathbf{n}(k)$ denotes the noise capturing the uncertainties of the odometric model and is modeled [16,19] as zero mean and Gaussian noise with covariance matrix $\mathbf{Q}(k)$ (2). In [22,23,19,18,24] some odometry models and statistical models for the odometry error are presented. The prediction step of the EKF is performed as follows

$$\begin{aligned} &\tilde{\mathbf{x}}_p(k+1) = \mathbf{f}(\hat{\mathbf{x}}_p(k), \mathbf{u}(k)), \\ &\tilde{\mathbf{P}}(k+1) = \mathbf{A}(k+1)\hat{\mathbf{P}}(k)\mathbf{A}^{\mathrm{T}}(k+1) + \mathbf{Q}(k), \\ &\mathbf{A}_{ij}(k+1) = \left.\frac{\partial \mathbf{f}_i}{\partial \mathbf{x}_{p\,j}(k)}\right|_{(\hat{\mathbf{x}}_p(k), \mathbf{u}(k))}, \end{aligned} \tag{2}$$

where $\hat{\mathbf{x}}_p(k)$ denotes the state estimate at time instant k obtained in the last correction step of the EKF and $\hat{\mathbf{P}}(k)$ denotes the covariance matrix of the corresponding estimation error. $\tilde{\mathbf{x}}_p(k+1)$ denotes the state prediction and $\tilde{\mathbf{P}}(k+1)$ denotes the covariance matrix of the state-prediction error.

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. The robot then builds a local environment map, which is also composed of line segments, from the current LRF scan.

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

coordinates

$$x_R \cos\psi_i + y_R \sin\psi_i = r_i. \tag{3}$$

The line segments in 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. The matching strategy is adopted from [16] and some other matching approach can be found in [15,24]. Each line segment of the local map is through the overlapping rate 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. If the overlapping rate between the most similar local and global line segment is below the threshold, the line segments are paired. The matching line parameters ψ_i and r_i from the current local map are collected in vector $\mathbf{z}(k) = [r_1, \psi_1, \ldots, r_N, \psi_N]^{\mathrm{T}}$, which is used as the input for the correction step of the EKF to update the vehicle's state. The parameters p_i and α_i of the matching 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

$$\begin{aligned} Ci &= p_i - \tilde{x}_r(k+1)\cos(\alpha_i) - \tilde{y}_r(k+1)\sin(\alpha_i), \\ \begin{bmatrix} \hat{r}_i \\ \hat{\psi}_i \end{bmatrix} &= \mu_i(\tilde{\mathbf{x}}_p(k+1), p_i, \alpha_i) \\ &= \begin{bmatrix} |Ci| \\ \alpha_i - \left(\tilde{\varphi}_r(k+1) - \dfrac{\pi}{2}\right) + (-0.5 \cdot \mathrm{sign}(Ci) + 0.5)\pi \end{bmatrix}, \end{aligned} \tag{4}$$

where $\tilde{\mathbf{x}}_p(k+1)$ (2) denotes the prediction of the robot's pose and the operator $|.|$ denotes the absolute value. The output model of the process (1) can then be defined by the vector

$$\begin{aligned} &\boldsymbol{\mu}(\tilde{\mathbf{x}}_p(k+1)) \\ &\quad = \left[\mu_1(\tilde{\mathbf{x}}_p(k+1), p_1, \alpha_1), \ldots, \mu_N(\tilde{\mathbf{x}}_p(k+1), p_N, \alpha_N)\right]^{\mathrm{T}}. \end{aligned} \tag{5}$$

The correction step of the EKF is performed as follows

$$\begin{aligned} \mathbf{K}(k+1) &= \tilde{\mathbf{P}}(k+1)\mathbf{H}^{\mathrm{T}}(k+1)(\mathbf{H}(k+1)\tilde{\mathbf{P}}(k+1)\mathbf{H}^{\mathrm{T}}(k+1) \\ &\quad + \mathbf{R}(\mathbf{k}+\mathbf{1}))^{-1}, \\ \hat{\mathbf{P}}(k+1) &= \tilde{\mathbf{P}}(k+1) - \mathbf{K}(k+1)\mathbf{H}(k+1)\tilde{\mathbf{P}}(k+1), \\ \mathbf{H}_{ij}(k+1) &= \left.\frac{\partial \boldsymbol{\mu}_i}{\partial \tilde{\mathbf{x}}_{p\,j}(k+1)}\right|_{\tilde{\mathbf{x}}_{p\,j}(k+1)}, \end{aligned} \tag{6}$$

$$\hat{\mathbf{x}}_p(k+1) = \tilde{\mathbf{x}}_p(k+1) + \mathbf{K}(k+1)(\mathbf{z}(k+1) - \boldsymbol{\mu}(\tilde{\mathbf{x}}_p(k+1))). \tag{7}$$

When applying the EKF, the noise arising from the LRF's distance and angle measurements affects the line parameters $\mathbf{z}(k) = [r_1, \psi_1, \ldots, r_N, \psi_N]^{\mathrm{T}}$ of the local map. The covariance matrix of the vector $\mathbf{z}(k)$ is the output-noise covariance matrix $\mathbf{R}(k+1)$ of the EKF and has a block-diagonal structure, where i-th block

$$\mathbf{R}_i(k+1) = \begin{bmatrix} \mathrm{var}(r_i) & \mathrm{cov}(r_i, \psi_i) \\ \mathrm{cov}(\psi_i, r_i) & \mathrm{var}(\psi_i) \end{bmatrix} \tag{8}$$

represents the covariance matrix of the line parameters (r_i, ψ_i). The $\mathbf{R}(k+1)$ matrix is as in [16] block diagonal, since it is assumed that there is no correlation between the estimated parameters r_i and ψ_i representing different environment line segments. In the following sections the formulas to define the $\mathbf{R}_i(k+1)$ covariance matrix are given.

2.2. Identification of the line parameters

The line segments are extracted from the laser range finder's (LRF) reflection points. The LRF in each environment scan (Fig. 1b) gives the set of distances $\mathbf{d}_s = [d_{s0^\circ}, \ldots, d_{s180^\circ}]$ to the obstacles (e.g., a wall) at the angles $\boldsymbol{\theta}_s = [0^\circ, \ldots, 180^\circ]$. The Sick LMS200 laser range finder, which is very often used in mobile robotics, is also taken into consideration in this paper. The LRF consists of a 1D laser range measurement device and a constantly rotating mirror with a rotation speed of 75 Hz. This LRF has according to the Sick LMS200 technical description [25] the following characteristics. It has a selectable scanning angle (field of vision) of 100° or 180° at selectable angular resolutions of 0.25° (100° scanning angle only), 0.5° and 1°, with the times for scanning one cycle 53.3 ms (100° scanning angle only), 26.6 ms and 13.13 ms, respectively. The maximum measurement distances are 8 m in the mm- mode and 80 m in the cm-mode, with a measurement resolution of 10 mm and typical measurement accuracy of ±35 mm. The systematic error in the mm-mode is typically ±15 mm at a range of 1 to 8 m and in the cm-mode it is typically ±40 mm at a range 1 to 20 m. The statistical error in the mm-mode is typically 5 mm at a range of ≤ 8 m. When the vehicle moves while scanning, the movement imposes a distortion on individual points of each environment scan. This effect is non-negligible for low mirror speeds and in [26] it is shown how each time when a new environment scan arrives, it gets immediately compensated by reading out the odometry. As in [27] it is assumed that this effect is negligible, due to using a SICK LMS200 laser range finder, which has high mirror speed (75 Hz). This assumption is however justified only when the robot moves with a low translational and rotational speed [27], e.g. a few meters per second.

All the consecutive points

$$\begin{aligned} &\mathbf{x}_{scan}(m) = \mathbf{d}_s(m)\cos\boldsymbol{\theta}_s(m), \qquad \mathbf{y}_{scan}(m) = \mathbf{d}_s(m)\sin\boldsymbol{\theta}_s(m), \\ &\quad m = 1, \ldots, Np, \end{aligned} \tag{9}$$

by which the reflections have occurred ($\mathbf{d}_s(m) \le R_{\mathrm{LRF}}$) are clustered; the other points ($\mathbf{d}_s(m) > R_{\mathrm{LRF}}$) are ignored, where R_{LRF} denotes the range of the LRF (e.g., 80 m). Each cluster is then split into more clusters if the distance between the two consecutive points is greater than the threshold, which in a particular environment is set according to the expected smallest distance (e.g., 15 cm) required to distinguish between two different consecutive line segments. If there is less than $N_{\min}$ points in a cluster, the cluster is ignored. $N_{\min}$ is a minimum number of scan points for a line segment (e.g., $N_{\min} = 5$). 2 points already define a line, but due to the LRF's noise more points must be taken into account to reliably describe the environment line segment with the calculated line segment. Each cluster is then split with the split-and-merge algorithm [13,24] in the consecutive sets of scan points, where each set of points $(\mathbf{x}, \mathbf{y})$ corresponds to a certain environment line segment. If there are less than $N_{\min} = 5$ points that correspond to certain environment line segment, the set of line segment $(\mathbf{x}, \mathbf{y})$ points is again discarded. According to a comparison of line-extraction algorithms using a 2D LRF done in [13], the split-and-merge algorithm is fast and has good correctness. Besides the line parameters the edge points of the line segments can also be computed, when using this algorithm. The line parameters are very often computed using the Hough transform [15,28–30]; however, this 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.

Each set of line-segment points $(\mathbf{x}, \mathbf{y})$ is reduced to the parameters ψ and r (Fig. 1a) of the line equation in normal form, according to the robot's coordinates (3) and to the edge points of the line segment. 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 lest-square estimated line parameters to the given set of data $(\mathbf{x}, \mathbf{y})$, the slope of the line is estimated first. This slope is estimated from the edge points

$(Ta_x,\ Ta_y)$ and $(Tb_x,\ Tb_y)$ of the set $(\mathbf{x}, \mathbf{y})$. If the absolute value of the slope is greater than 1

$$|Ta_y - Tb_y| > |Ta_x - Tb_x|, \tag{10}$$

all the points $(\mathbf{x}, \mathbf{y})$ are rotated by $-\frac{\pi}{2}$ 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 into the parameters r and ψ of the line equation in normal form according to the robot's coordinates (3) as follows

$$\mathbf{y} = [\mathbf{y}(1), \ldots, \mathbf{y}(n)]^{\mathrm{T}}, \qquad \mathbf{U} = \begin{bmatrix} \mathbf{x}(1) & \cdots & \mathbf{x}(n) \\ 1 & \cdots & 1 \end{bmatrix}^{\mathrm{T}},$$

$$\hat{\boldsymbol{\theta}} = [\hat{k}_l, \hat{c}_l]^{\mathrm{T}} = (\mathbf{U}^{\mathrm{T}}\mathbf{U})^{-1}\mathbf{U}^{\mathrm{T}}\mathbf{y}, \tag{11}$$

$$r(k_l, c_l) = \frac{c_l}{\sqrt{k_l^2 + 1}}\mathrm{sign}(c_l),$$

$$\psi(k_l) = \arctan 2\left(\frac{\mathrm{sign}(c_l)}{\sqrt{k_l^2 + 1}}, \frac{-k_l}{\sqrt{k_l^2 + 1}}\mathrm{sign}(c_l)\right), \tag{12}$$

where n denotes the number of points that corresponds to the line segment. $\hat{k}_l$ and $\hat{c}_l$ are parameters of the explicit line equation $y_R = k_l \cdot x_R + c_l$, estimated with the classic LSQ method. Both parameters are then converted into the parameters $r(\hat{k}_l, \hat{c}_l)$ and $\psi(\hat{k}_l)$ (12) of the line equation in normal form, where the function arctan 2 is a four quadrant arctan function. If condition (10) is satisfied, all the points corresponding to the line segment were rotated for $-\frac{\pi}{2}$. In this case $\frac{\pi}{2}$ must be added up to the calculated angle ψ (12) to get the right line parameter. The calculated parameter r which denotes the distance of line to the coordinate-frame origin (Fig. 1a) is invariant to the rotation of line-segment points and therefore remains unchanged.

The line parameters r_i and ψ_i (Fig. 1a) of the line equation in normal form (3) can also be computed with the very often used orthogonal least-squares method [16]. The parameters r^* and ψ^* minimizing the cost function $E(r, \psi) = \sum_{i=1}^{N}(r - \mathbf{x}(i)\cos(\psi) - \mathbf{y}(i)\sin(\psi))^2$ are calculated with the function $\mathbf{f} = [f_1,\ f_2]$

$$\begin{bmatrix} r^* \\ \psi^* \end{bmatrix} = \begin{bmatrix} \bar{x}\cos(\psi^*) + \bar{y}\sin(\psi^*) \\ \frac{1}{2}\arctan\left(\frac{-2S_{xy}}{S_{y^2} - S_{x^2}}\right) \end{bmatrix} \triangleq \begin{bmatrix} f_1(\mathbf{x}(1), \mathbf{y}(1), \ldots, \mathbf{x}(n), \mathbf{y}(n)) \\ f_2(\mathbf{x}(1), \mathbf{y}(1), \ldots, \mathbf{x}(n), \mathbf{y}(n)) \end{bmatrix}. \tag{13}$$

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

$$S_{y^2} = \sum_{j=1}^{n}(\mathbf{y}(j) - \bar{y})^2, \qquad S_{xy} = \sum_{j=1}^{n}(\mathbf{x}(j) - \bar{x})(\mathbf{y}(j) - \bar{y}). \tag{14}$$

2.3. Estimation of line parameters' covariances

Besides the line parameters r_i and ψ_i (12), the variances of both parameters and the covariances between them, which compose the covariance matrix of vector $[r_i,\ \psi_i]$

$$\mathbf{C} = \begin{bmatrix} \mathrm{var}(r) & \mathrm{cov}(r, \psi) \\ \mathrm{cov}(\psi, r) & \mathrm{var}(\psi) \end{bmatrix}, \tag{15}$$

must 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 range-sensor (e.g., a LRF) readings $\mathbf{d}_s(m)$ and $\boldsymbol{\theta}_s(m)$ affects both of the extracted line parameters.

A method for estimating the line parameters' covariances resulting from classic LSQ is proposed here. The covariance matrix of the line parameters' vector $[k_l,\ c_l]$ must be calculated first. The error between the y_R coordinates of the line-segment points $(\mathbf{x},\ \mathbf{y})$ and the estimated line arises from the noise of the LRF. Assuming that this error is white noise, the covariance matrix of the line parameters' vector $[k_l,\ c_l]$ can be calculated according to the least-squares theory by

$$\mathbf{Ce} = \mathrm{var}(\mathbf{y}(j))(\mathbf{U}^{\mathrm{T}}\mathbf{U})^{-1} = \begin{bmatrix} \mathrm{var}(k_l) & \mathrm{cov}(k_l, c_l) \\ \mathrm{cov}(c_l, k_l) & \mathrm{var}(c_l) \end{bmatrix}, \tag{16}$$

$$\mathrm{var}(\mathbf{y}(j)) = \frac{\sum_{j=1}^{n}(\mathbf{y}(j) - \hat{\mathbf{y}}(j))^2}{n - 1}, \qquad \hat{\mathbf{y}}(j) = \hat{k}_l \cdot \mathbf{x}(j) + \hat{c}_l, \tag{17}$$

where $\mathrm{var}(\mathbf{y}(j))$ is the vertical-error variance of the points $(\mathbf{x}(j),\ \mathbf{y}(j))$ $(j = 1, \ldots, n)$ according to the estimated line with parameters $\hat{k}_l$ and $\hat{c}_l$. Because $\hat{\mathbf{y}}(j)$ is calculated out of $\mathbf{x}(j)$ coordinate, variance $\mathrm{var}(\mathbf{y}(j))$ is calculated out of both $(\mathbf{x}(j),\ \mathbf{y}(j))$ cartesian coordinates of the LRF's points. Since these two coordinates are calculated as $(\mathbf{d}_s(j)\cos\boldsymbol{\theta}_s(j),\ \mathbf{d}_s(j)\sin\boldsymbol{\theta}_s(j))$ (9), the uncertainties of the LRF's range and angle measurements $(\mathbf{d}_s(j),\ \boldsymbol{\theta}_s(j))$ are both incorporated in vertical error variance $\mathrm{var}(\mathbf{y}(j))$. Knowing the variances and covariances between the parameters k_l and c_l the variances and covariances between the parameters r and ψ can be approximated by a second-order Taylor expansion of the Eqs. (12)

$$\Delta r \approx \frac{\partial r(\hat{k}_l, \hat{c}_l)}{\partial k_l}\Delta k_l + \frac{\partial r(\hat{k}_l, \hat{c}_l)}{\partial c_l}\Delta c_l, \qquad \Delta\psi \approx \frac{\partial \psi(\hat{k}_l)}{\partial k_l}\Delta k_l, \tag{18}$$

where Δc_l, Δk_l, Δr and $\Delta\psi$ are deviations in a neighborhood of the values $\hat{k}_l$, $\hat{c}_l$, r and ψ, respectively. The partial derivatives are

$$K_{rk} = \frac{\partial r(\hat{k}_l, \hat{c}_l)}{\partial k_l} = \frac{-\hat{c}_l\hat{k}_l}{\sqrt{\hat{k}_l^2 + 1(\hat{k}_l^2 + 1)}}\mathrm{sign}(\hat{c}_l),$$

$$K_{rc} = \frac{\partial r(\hat{k}_l, \hat{c}_l)}{\partial c_l} = \frac{\mathrm{sign}(\hat{c}_l)}{\sqrt{\hat{k}_l^2 + 1}}, \qquad K_{\psi k} = \frac{\partial \psi(\hat{k}_l)}{\partial k_l} = \frac{1}{\hat{k}_l^2 + 1}, \tag{19}$$

where in the derivation of the $K_{\psi k}$ the partial derivation $\frac{\partial \arctan\frac{|\sin\psi(k_l)|}{|\cos\psi(k_l)|}}{\partial k_l} = \frac{\partial \arctan\frac{1}{|k_l|}}{\partial k_l}$ and the definition of the four quadrant arctan 2 function is considered. Considering the approximations of Δr and $\Delta\psi$ (18) and the statistical properties of the random variables the estimations of the variances and covariances between the line parameters r and ψ are derived as follows

$$\begin{aligned} &\mathrm{var}(\psi) = K_{\psi k}^2\mathrm{var}(k_l), \\ &\mathrm{var}(r) = K_{rk}^2\mathrm{var}(k_l) + K_{rc}^2\mathrm{var}(c_l) + 2K_{rk}K_{rc} \cdot \mathrm{cov}(k_l, c_l), \\ &\mathrm{cov}(r, \psi) = K_{rk}K_{\psi k}\mathrm{var}(k_l) + K_{rc}K_{\psi k} \cdot \mathrm{cov}(k_l, c_l), \\ &\mathrm{cov}(\psi, r) = \mathrm{cov}(r, \psi). \end{aligned} \tag{20}$$

If condition (10) is satisfied all the line-segment points are rotated for the angle $-\frac{\pi}{2}$. Then the parameters r and ψ (12) and their covariances (20) are calculated. To get the right line parameter, $\frac{\pi}{2}$ must be added up to the calculated angle ψ (12). The calculated parameter r denotes the distance of the line to the coordinate-frame origin (Fig. 1a). The parameter is invariant to the rotation of line-segment points and therefore remains unchanged. Therefore the variance of parameter r also remains unchanged and equals the already calculated $\mathrm{var}(r)$ in Eq. (20). The variance of the right angle $\mathrm{var}(\psi + \frac{\pi}{2})$ equals the already calculated variance $\mathrm{var}(\psi)$ (20), since $\mathrm{var}(\frac{\pi}{2}) = 0$. And the covariance $\mathrm{covar}(r,\ \psi + \frac{\pi}{2})$ equals the already calculated covariance $\mathrm{covar}(r,\ \psi)$ (20), since $\mathrm{covar}(r,\ \frac{\pi}{2}) = 0$.

If the line parameters r_i and ψ_i (Fig. 1a) of the line equation in normal form (3) are calculated using the orthogonal least-squares method (13) and (14), then the covariance matrix of the line parameters (15) can be, as shown in [16], calculated as follows

$$\mathbf{C}^* = \sum_{j=1}^{n} (\mathbf{A}_j\mathbf{B}_j)\mathbf{C}_{m_j}(\mathbf{A}_j\mathbf{B}_j)', \tag{21}$$

$$\mathbf{C}_{m_j} = \begin{bmatrix} \sigma_{d_j}^2 & \sigma_{d_j\,\theta_j} \\ \sigma_{\theta_j\,d_j} & \sigma_{\theta_j}^2 \end{bmatrix},$$

$$\mathbf{B}_j = \begin{bmatrix} \dfrac{\partial g_1}{\partial \mathbf{d}_s(j)} & \dfrac{\partial g_1}{\partial \boldsymbol{\theta}_s(j)} \\ \dfrac{\partial g_2}{\partial \mathbf{d}_s(j)} & \dfrac{\partial g_2}{\partial \boldsymbol{\theta}_s(j)} \end{bmatrix} = \begin{bmatrix} \cos\boldsymbol{\theta}_s(j) & -\mathbf{d}_s(j)\sin\boldsymbol{\theta}_s(j) \\ \sin\boldsymbol{\theta}_s(j) & \mathbf{d}_s(j)\cos\boldsymbol{\theta}_s(j) \end{bmatrix}, \tag{22}$$

$$\mathbf{A}_j = \begin{bmatrix} \dfrac{\partial f_1}{\partial \mathbf{x}(j)} & \dfrac{\partial f_1}{\partial \mathbf{y}(j)} \\ \dfrac{\partial f_2}{\partial \mathbf{x}(j)} & \dfrac{\partial f_2}{\partial \mathbf{y}(j)} \end{bmatrix},$$

$$\begin{bmatrix} \mathbf{x}(j) \\ \mathbf{y}(j) \end{bmatrix} = \begin{bmatrix} \mathbf{d}_s(j)\cos(\boldsymbol{\theta}_s(j)) \\ \mathbf{d}_s(j)\sin(\boldsymbol{\theta}_s(j)) \end{bmatrix} \triangleq \begin{bmatrix} g_1(\mathbf{d}_s(j), \boldsymbol{\theta}_s(j)) \\ g_2(\mathbf{d}_s(j), \boldsymbol{\theta}_s(j)) \end{bmatrix}, \tag{23}$$

where $\mathbf{C}_{m_j}$ denotes the covariance matrix of a point in the polar coordinates $[\mathbf{d}_s(j), \boldsymbol{\theta}_s(j)]$, which is a raw sensor measurement (e.g., a LRF). The polar coordinates $[\mathbf{d}_s(j), \boldsymbol{\theta}_s(j)]$ are transformed (23) to cartesian coordinates $[\mathbf{x}(j),\ \mathbf{y}(j)]$ by the relations g_1 and g_2. These transformations are linearized according to the polar coordinates to obtain the matrix $\mathbf{B}_j$. $\mathbf{A}_j$ is the Jacobian matrix of $\mathbf{f} = [f_1,\ f_2]$ according to the cartesian coordinates. The partial derivatives from the $\mathbf{A}_j$ matrices are shown in [16].

2.4. Estimation of the computational complexity

To solve the localization or SLAM problem the vectors of the two line parameters $[r_i,\ \psi_i]$ and their covariance matrices must be calculated every time an observation is made with a range sensor for all the observed environment line segments. Both, SLAM and localization algorithm, must run in real time. SLAM is known as a computationally much more complex algorithm. Therefore, the computational efficiency of the same algorithms that are used in a SLAM and a localization algorithm plays more important role in SLAM than in the localization algorithm. The proposed method for estimating the covariances of the observed line parameters resulting from the classic LSQ method and the method resulting from orthogonal LSQ [16] were described.

The computational complexity of both methods will be compared relative to each other. 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 (12) and their covariance matrix $\mathbf{C}$ (15). These operations are addition, subtraction, multiplication, division, square rooting and the trigonometric functions cosine, sine and arctan. The elementary operations in the same terms that are used a number of times are counted only once. The algorithms used to perform the multiplication, division and square-root operations are optimized on modern computers to reach the time complexity of the addition and subtraction. For this reason all these elementary operations are counted up together for each method. The time of calculating the cosine, sine and arctan functions and the time to perform the multiplication operation in a C++ language were measured on laptop computer. Measurements have shown that calculating the cosine or sine function is about 4 times longer than performing the multiplication operation. Calculating the arctan function is about 8 times longer than the multiplication operation. Each cosine or sine calculation can then be denoted as 4 multiplications and each arctan calculation can be denoted as 8 multiplications in terms of the time complexity.

Table 1
Computational costs of the method resulting from orthogonal LSQ in the case of nonzero and zero LRF's angular variance $\sigma_{\theta_j}^2$.

	N_{eq}
$\mathbf{B}_j$ matrices n times (22)	$11n/8n$
$\mathbf{A}_j$ matrices n times (23)	$18n + 19$
$(\mathbf{A}_j\mathbf{B}_j)\mathbf{C}_{m_j}$ n times (21)	$16n$ / $8n$
$(\mathbf{A}_j\mathbf{B}_j\mathbf{C}_{m_j}) * (\mathbf{A}_j\mathbf{B}_j)'$ n times (21)	$9n$
$n-1$ matrix additions (21)	$3n-3$
$\bar{x},\ \bar{y},\ S_{x^2},\ S_{y^2},\ S_{xy}$ (14)	$12n-3$
$r^*,\ \psi^*$ (13)	23
Overall computational costs	$C_{ols1}(n) = 69n + 36/$ $C_{ols2}(n) = 58n + 36$

First, the computational complexity of the method resulting from the orthogonal LSQ is examined. In Table 1 are shown the computational costs of all the terms involved in the calculation of the line parameters r and ψ (13) and their covariance matrix $\mathbf{C}^*$ (21). N_{eq} denotes the equivalent number of operations, that are necessary to calculate certain mathematical term, taking into account that each cosine (sine) calculation or each arctan calculation is equivalent to 4 or 8 multiplications, respectively. The computational costs obviously depend on the number of line-segment points n. The covariance matrix of a point in the polar coordinates $\mathbf{C}_{m_j}$ should be a priori given from a LRF's noise model. The covariances between the LRF's distance and angle measurement $\sigma_{d_j\theta_j}$ and $\sigma_{\theta_j d_j}$ (22) are here set to zero, since it is as in [19] assumed that there is no correlation between the distance and angle measurement at the LRF sensors. The error on the LRF's beam-angle $\boldsymbol{\theta}_s(j)$ is in some papers neglected [20,31,17], whereas in some papers it is considered [24,19,18,16]. Thus, here two cases of setting the variance of the LRF's beam angle $\sigma_{\theta_j}^2$ are considered: the noise case with nonzero $\sigma_{\theta_j}^2$ and the noise-free case with zero variance $\sigma_{\theta_j}^2 = (0\ \text{rad})^2$. If the computational cost of some term is in the noise-free case different than in the noise-case, then the computational cost before the sign '/' (Table 1) refers to the noise case and the cost behind the sign '/' refers to the noise-free case. Since the covariances $\sigma_{d_j\theta_j}$ and $\sigma_{\theta_j d_j}$ in the matrices $\mathbf{C}_{m_j}$ equals zero, the upper-right and the lower-left terms in the matrix products $(\mathbf{A}_j\mathbf{B}_j\mathbf{C}_j) * (\mathbf{A}_j\mathbf{B}_j)'$ (21) are equal. In the computational cost of the product $(\mathbf{A}_j\mathbf{B}_j\mathbf{C}_j) * (\mathbf{A}_j\mathbf{B}_j)'$ (21) (Table 1) is therefore considered that only one of this two terms is calculated. Since the covariances $\text{covar}(r,\ \psi) = \mathbf{C}^*(1,2)$ and $\text{covar}(\psi,\ r) = \mathbf{C}^*(2,1)$ are equal, in the computational costs of the $n-1$ additions of 2×2 matrices (21) (Table 1) it is considered that only one of the two terms $\mathbf{C}^*(1,2)$ and $\mathbf{C}^*(2,1)$ is calculated. $C_{ols1}(n)$ and $C_{ols2}(n)$ denote the overall computational costs of the method resulting from orthogonal LSQ in the noise cases, with nonzero and zero LRF's angular variance $\sigma_{\theta_j}^2$, respectively. $C_{ols1}(n)$ and $C_{ols2}(n)$ are obtained by summing up the equivalent numbers of computations (N_{eq}) of all the mathematical terms shown in Table 1.

In Table 2 are shown the computational costs of the method resulting from classic LSQ. At the computational costs of the term $\mathbf{U^TU}$ (16) it is considered that this matrix product can be written as

$$\mathbf{U^TU} = \begin{bmatrix} \sum_{j=1}^{n} \mathbf{x}(j)^2 & \sum_{j=1}^{n} \mathbf{x}(j) \\ \sum_{j=1}^{n} \mathbf{x}(j) & n \end{bmatrix}, \tag{24}$$

where the upper-right and the lower-left terms are equal. It is considered that the 2×2 matrix inversion in the term $\mathbf{Ce} = \text{var}(\mathbf{y}(j)) * (\mathbf{U^TU})^{-1}$ (16) is calculated analytically with cofactors

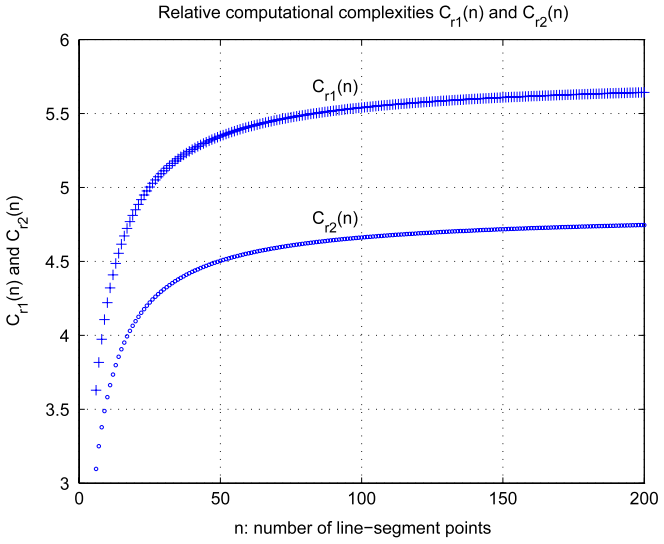

Fig. 2. The relative computational complexities $C_{r1}(n) = \frac{C_{ols1}(n)}{C_{ls}(n)}$ and $C_{r2}(n) = \frac{C_{ols2}(n)}{C_{ls}(n)}$ as a function of the number of line-segment points $n \geq 6$.

Table 2
Computational costs of the method resulting from classic LSQ.

	N_{eq}
var($\mathbf{y}(j)$) (17)	$5n+1$
Matrix product $\mathbf{U^T U}$ (16)	$3n-2$
var($\mathbf{y}(j)$) $*$ $(\mathbf{U^T U})^{-1}$ (16)	10
$K_{rk}, K_{rc}, K_{\psi k}$ (19)	10
var(ψ), var(r), cov(r, ψ) (20)	16
$(\mathbf{U^T U})^{-1}\mathbf{U^T y}$ (11)	$4n+4$
r, ψ (12)	13
Overall computational costs	$C_{ls}(n) = 12n+52$

and determinant and that covariances $\text{cov}(c_l, k_l) = \mathbf{Ce}(2,1)$ and $\text{cov}(k_l, c_l) = \mathbf{Ce}(1,2)$ are equal. In the computational costs of the term $\mathbf{Ce} = \text{var}(\mathbf{y}(j)) * (\mathbf{U^T U})^{-1}$ (16) is therefore considered that only one of the upper-right and lower-left term of the matrix inversion $(\mathbf{U^T U})^{-1}$ is calculated and multiplied with variance var($\mathbf{y}(j)$). If condition (10) is satisfied, each line segment point $(\mathbf{x}(j),\ \mathbf{y}(j))$ $(j = 1, \ldots, n)$ was replaced with rotated point $(\mathbf{y}(j),\ -\mathbf{x}(j))$. That takes n subtractions which is in Table 2 considered in the computational cost of calculating the term $(\mathbf{U^T U})^{-1}\mathbf{U^T y}$ (11). In the computational cost of the line parameters r and ψ (12) the worst case of calculating the arctan $2(y_a,\ x_a)$ function $(\pi - \arctan(\frac{|y_a|}{|x_a|})) * \text{sign}(y_a)$ is considered. If condition (10) is satisfied, then $\frac{\pi}{2}$ must be added to the calculated angle ψ (12), which is also considered in this cost. $C_{ls}(n)$ (Table 2) denote the overall computational costs of the method resulting from the classic LSQ.

The computational complexities of both methods are compared relative to each other by

$$C_{r1}(n) = \frac{C_{ols1}(n)}{C_{ls}(n)} = \frac{69n+36}{12n+52},$$
$$C_{r2}(n) = \frac{C_{ols2}(n)}{C_{ls}(n)} = \frac{58n+36}{12n+52}, \qquad (25)$$

where the relative computational complexities $C_{r1}(n)$ and $C_{r2}(n)$ refer to the the noise cases with nonzero and zero LRF's angular variance $\sigma^2_{\theta_j}$, respectively. Fig. 2 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 25 points or 25 to 200 points, the method resulting from classic LSQ has in the noise case with nonzero LRF's angular variance $\sigma^2_{\theta_j}$ about 4 to 5 times or 5 to 5.6 times fewer operations than the method resulting from orthogonal LSQ, respectively. The computational costs of the method resulting from orthogonal LSQ are in the noise case with zero LRF's angular variance $\sigma^2_{\theta_j} = (0\,\text{rad})^2$ somewhat lower. If the line parameters and their covariance matrix are calculated from 9 to 50 points or 50 to 200 points, the method resulting from classic LSQ has in this noise case about 3.5 to 4.5 times or 4.5 to 4.7 times fewer operations than the method resulting from orthogonal LSQ, respectively.

Table 3
Three environment line segments, which correspond to $n = 36$ LRF's points.

$r_{e1} = 2$ m,	$\psi_{e1} = \frac{\pi}{2}$ rad,	$\boldsymbol{\theta}_s(j)$: 60°, 61°, …, 95°
$r_{e2} = 50$ m,	$\psi_{e2} = 130\frac{\pi}{180}$ rad,	$\boldsymbol{\theta}_s(j)$: 80°, 81°, …, 115°
$r_{e3} = 10$ m,	$\psi_{e3} = 170\frac{\pi}{180}$ rad,	$\boldsymbol{\theta}_s(j)$: 97°, 98°, …, 132°

To solve the localization or SLAM problem the vectors of the two line parameters $[r_i,\ \psi_i]$ and their covariance matrices must be calculated in each environment scan for all the observed environment line segments. It was shown that the use of classic LSQ instead of orthogonal LSQ reduces the number of computations in the process of estimating the two normal line-equation parameters and their covariance matrix. Exactly the same algorithms for calculating these line parameters and covariance matrices that can be applied in a localization algorithm would then also be applied in a SLAM algorithm, which is extended out of the localization algorithm. The reduction of the number of computations (if using classic LSQ instead of using orthogonal LSQ) would therefore be in the SLAM algorithm exactly the same (from the absolute point of view) as in the localization algorithm.

3. Statistical validation of both methods

The correctness of both methods for estimating the covariance matrix of line parameters will be compared using statistical analysis. Three different environment line segments are simulated in Matlab. They are defined with the normal line equation parameters $(r_{ei}, \psi_{ei};\ i = 1, 2, 3)$ (Tables 3–5) and laser beam angles $\boldsymbol{\theta}_s(j)$, at 1° resolution. Line segments are extracted out of $n = 36$ points.

To test both methods for estimating the covariance matrix the following LRF's noise is assumed

$$\boldsymbol{\theta}_s(j) = \boldsymbol{\theta}(j) + N(0, \sigma_{\theta_j}), \qquad \mathbf{d}_s(j) = \mathbf{d}(j) + N(0, \sigma_{d_j}), \qquad (26)$$

where $\boldsymbol{\theta}(j)$ denotes the true laser–beam angle and $N(0, \sigma_{\theta_j})$ denotes and the normally distributed noise with zero mean and variance of the laser–beam angle error $\sigma^2_{\theta_j}$. $\mathbf{d}(j)$ is the true distance between the LRF sensor and the simulated environment line segment at the true LRF's beam angle $\boldsymbol{\theta}(j)$. $N(0, \sigma_{d_j})$ denotes the normally distributed noise with zero mean and variance of the LRF's distance-measurement error $\sigma^2_{d_j}$. The covariance between the LRF's distance and angle measurement $\sigma_{d_j\theta_j}$ (22) is in this model and in the following experiments set to zero, since it is as in [19] assumed that there is no correlation between the distance and angle measurement at the LRF sensors.

As already mentioned the error on the LRF's beam-angle $\boldsymbol{\theta}_s(j)$ is in some papers neglected [20,31,17], whereas in some papers it is considered [24,19,18,16]. Thus, here two cases of setting the standard deviation of the LRF's beam angle σ_{θ_j} are considered: the noise case with $\sigma_{\theta_j} = 0.0017$ rad (Table 5) and the noise-free case with $\sigma_{\theta_j} = 0$ rad (Table 4). The standard deviation of the LRF's distance-measurement error is set to $\sigma_{d_j} = 30$ mm (Tables 4 and 5), which is comparable to the distance–measurement error of the LRF used. It is assumed here that the standard deviations σ_{d_j} and σ_{θ_j} of all the line-segment points are equal.

Experiments for estimating the line parameters from the LRF's points by using the classic LSQ (cLSQ) and the orthogonal LSQ (oLSQ) methods are repeated many times (e.g., $N_{tr} = 10\,000$) for each of the three environment line segments and both noise

Table 4
Comparison of the accuracy of the method resulting from classic (cLSQ) and orthogonal LSQ (oLSQ) considering zero LRF's angular noise ($\sigma_{\theta_j} = 0$ rad), where three different environment line segments (r_{ei}, ψ_{ei}; $i = 1, 2, 3$) are simulated.

	r_{e1}, ψ_{e1}		r_{e2}, ψ_{e2}		r_{e3}, ψ_{e3}	
	cLSQ	oLSQ	cLSQ	oLSQ	cLSQ	oLSQ
σ_{r_u} (mm)	7.8	7.9	11.9	11.9	7.9	7.9
mean(σ_r) (mm)	7.3	7.5	11.0	11.5	7.2	7.9
std(σ_r) (mm)	0.9	0.9	1.4	1.4	1.0	1.0
a-priori : *mean*(σ_r) (mm)	7.4	7.7	11.2	11.8	7.4	8.0
a-priori : *std*(σ_r) (mm)	0.2	0.2	9×10^{-4}	0.001	0.006	0.006
σ_{ψ_u} (rad)	0.012	0.012	3×10^{-4}	3×10^{-4}	4×10^{-4}	4×10^{-4}
mean(σ_ψ) (rad)	0.012	0.012	3×10^{-4}	3×10^{-4}	4×10^{-4}	4×10^{-4}
std(σ_ψ) (rad)	0.001	0.001	4×10^{-5}	4×10^{-5}	6×10^{-5}	5×10^{-5}
a-priori : *mean*(σ_ψ) (rad)	0.012	0.012	3×10^{-4}	3×10^{-4}	4×10^{-4}	4×10^{-4}
a-priori : *std*(σ_ψ) (rad)	8×10^{-5}	8×10^{-5}	10^{-7}	10^{-7}	4×10^{-7}	4×10^{-7}
cov(r_u, ψ_u) (mm rad)	−0.076	−0.077	−0.003	−0.003	−0.003	−0.003
mean(cov(r, ψ)) (mm rad)	−0.068	−0.071	−0.003	−0.003	−0.003	−0.003
std(cov(r, ψ)) (mm rad)	0.017	0.018	8×10^{-4}	8×10^{-4}	7×10^{-4}	7×10^{-4}
a-priori : *mean*(cov(r, ψ)) (mm rad)	−0.070	−0.073	−0.003	−0.003	−0.003	−0.003
a-priori : *std*(cov(r, ψ)) (mm rad)	0.003	0.003	10^{-6}	10^{-6}	5×10^{-6}	5×10^{-6}

Table 5
Comparison of the accuracy of the method resulting from classic (cLSQ) and orthogonal LSQ (oLSQ) considering nonzero LRF's angular noise ($\sigma_{\theta_j} = 17 \times 10^{-3}$ rad), where three different environment line segments (r_{ei}, ψ_{ei}; $i = 1, 2, 3$) are simulated.

	r_{e1}, ψ_{e1}		r_{e2}, ψ_{e2}		r_{e3}, ψ_{e3}	
	cLSQ	oLSQ	cLSQ	oLSQ	cLSQ	oLSQ
σ_{r_u} (mm)	7.7	7.7	26.9	26.9	15.6	15.6
mean(σ_r) (mm)	7.3	/	28.7	/	14.4	/
std(σ_r) (mm)	0.9	/	4.0	/	2.0	/
a-priori : *mean*(σ_r) (mm)	7.5	7.7	29.5	26.9	14.8	15.5
a-priori : *std*(σ_r) (mm)	0.2	0.2	0.1	0.1	0.01	0.02
σ_{ψ_u} (rad)	0.012	0.012	9×10^{-4}	9×10^{-4}	0.001	0.001
mean(σ_ψ) (rad)	0.012	/	8×10^{-4}	/	8×10^{-4}	/
std(σ_ψ) (rad)	0.001	/	10^{-4}	/	10^{-4}	/
a-priori : *mean*(σ_ψ) (rad)	0.012	0.012	8×10^{-4}	9×10^{-4}	8×10^{-4}	0.001
a-priori : *std*(σ_ψ) (rad)	7×10^{-5}	8×10^{-5}	3×10^{-7}	6×10^{-7}	2×10^{-6}	1×10^{-6}
cov(r_u, ψ_u) (mm rad)	−0.074	−0.075	−0.023	−0.023	−0.015	−0.015
mean(cov(r, ψ)) (mm rad)	−0.068	/	−0.022	/	−0.011	/
std(cov(r, ψ)) (mm rad)	0.017	/	0.006	/	0.003	/
a-priori : *mean*(cov(r, ψ)) (mm rad)	−0.070	−0.074	−0.022	−0.023	−0.011	−0.015
a-priori : *std*(cov(r, ψ)) (mm rad)	0.003	0.003	5×10^{-5}	7×10^{-5}	4×10^{-5}	4×10^{-5}

cases. The line parameters r_u and ψ_u ($u = 1, \ldots, N_{tr}$) calculated by cLSQ (11), (12) and oLSQ method (13) and (14) slightly differ in each experiment due to the influence of the LRF's noise (26). The standard deviations of both line parameters and the covariance between them are calculated for both methods by

$$\sigma_{r_u} = \sqrt{\frac{1}{N_{tr}} \sum_{u=1}^{N_{tr}} (r_u - r_{ei})^2}, \qquad \sigma_{\psi_u} = \sqrt{\frac{1}{N_{tr}} \sum_{u=1}^{N_{tr}} (\psi_u - \psi_{ei})^2},$$
$$\mathrm{cov}(r_u, \psi_u) = \frac{1}{N_{tr}} \sum_{u=1}^{N_{tr}} (r_u - r_{ei}) * (\psi_u - \psi_{ei}); \quad i = 1, 2, 3, \tag{27}$$

and are shown in Tables 4 and 5. These statistically obtained standard deviations and covariances are taken as a reference for a comparison of the accuracy of the already-described methods for calculating the variances and the covariance of the line parameters from one set of line-segment points.

First, the standard deviations $\sigma_r = \sqrt{\mathrm{var}(r)}$, $\sigma_\psi = \sqrt{\mathrm{var}(\psi)}$ and the covariance $\mathrm{covar}(r, \psi)$ are calculated with the method resulting from classic LSQ (20). Here, the vertical-error variance $\mathrm{var}(\mathbf{y}(j))$ (16) is calculated from the LRF's points, as shown in (17). If this variance is estimated from a very small (e.g., 5) number of line-segment points, the estimation is not very accurate. If an accurate variance of the LRF's distance-measurement error $\sigma_{d_j}^2$ and variance of the laser–beam angle error $\sigma_{\theta_j}^2$ are given from a LRF's noise model, a better estimation of the vertical-error variance $\mathrm{var}(\mathbf{y}(j))$ can be calculated by

$$\mathrm{var}(\mathbf{y}(j)) = \frac{1}{n} \sum_{j=1}^{n} (\sigma_{d_j} \sin \boldsymbol{\theta}_s(j) + (\mathbf{y}_p(j) - \mathbf{y}_{\sigma d}(j)))^2 + \frac{1}{n} \sum_{j=1}^{n} (\Delta d_{\sigma\theta}(j) \sin \boldsymbol{\theta}_s(j) + (\mathbf{y}_p(j) - \mathbf{y}_{\sigma\theta}(j)))^2, \tag{28}$$

$$\mathbf{x}_p(j) = \frac{r \cos \boldsymbol{\theta}_s(j)}{\cos\psi \cos \boldsymbol{\theta}_s(j) + \sin\psi \sin \boldsymbol{\theta}_s(j)},$$
$$\mathbf{y}_p(j) = \frac{r \sin \boldsymbol{\theta}_s(j)}{\cos\psi \cos \boldsymbol{\theta}_s(j) + \sin\psi \sin \boldsymbol{\theta}_s(j)}, \tag{29}$$
$$\mathbf{y}_{\sigma d}(j) = \frac{r - (\mathbf{x}_p(j) + \sigma_{d_j} \cos \boldsymbol{\theta}_s(j)) \cos\psi}{\sin\psi},$$

$$\Delta d_{\sigma\theta}(j) = \sqrt{\mathbf{x}_p^2(j) + \mathbf{y}_p^2(j)} - \sqrt{\mathbf{x}_{p\sigma\theta}^2(j) + \mathbf{y}_{p\sigma\theta}^2(j)} \tag{30}$$

where $P1 = (\mathbf{x}_p(j), \mathbf{y}_p(j))$ (Fig. 3) is the intersection between the j-th laser–beam line with the equation $y_R = \tan \boldsymbol{\theta}_s(j) x_R$ and the environment line with the parameters r and ψ (3). $\mathbf{y}_{\sigma d}(j)$ is the y_R coordinate of the point on the environment line (r, ψ) at $\mathbf{x}_p(j) + \sigma_{d_j} \cos \boldsymbol{\theta}_s(j)$ coordinate (Fig. 3). $\sigma_{d_j}^2$ is the variance of the distance of each LRF's point from the real-environment line segment in the direction of a certain LRF's beam. $(\sigma_{d_j} \sin \boldsymbol{\theta}_s(j) + (\mathbf{y}_p(j) - \mathbf{y}_{\sigma d}(j)))^2$ is the variance of the distance of the j-th LRF's

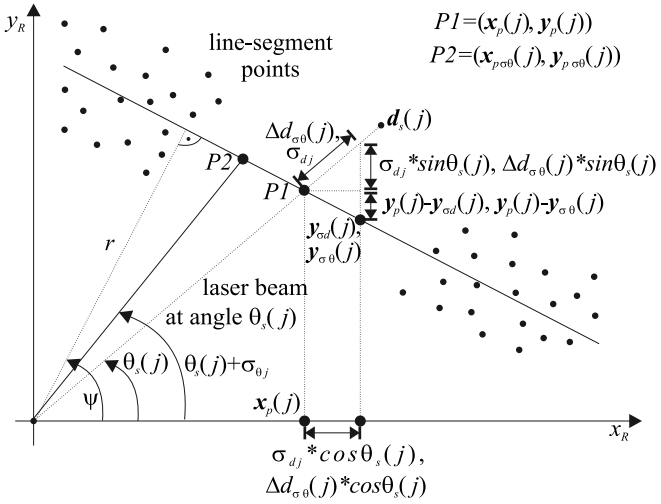

Fig. 3. Estimating the vertical-error variance var($\mathbf{y}(j)$) from an a priori known LRF's distance and angular variance $\sigma_{d_j}^2$ and $\sigma_{\theta_j}^2$.

point from the estimated environment line segment (r, ψ) in a vertical direction. This variance depends on the the standard deviation of the LRF's distance measurement σ_{d_j}, the laser–beam angle $\boldsymbol{\theta}_s(j)$ and the slope of the estimated line ψ. Since the environment line was estimated from n LRF's points $(\mathbf{d}_s(j), \boldsymbol{\theta}_s(j))$, all of these n variances that refer to a certain angle $\boldsymbol{\theta}_s(j)$ are averaged (28) to obtain the relevant part of the vertical-error variance var($\mathbf{y}(j)$), which corresponds to the error on distances $\mathbf{d}_s(j)$. $P2 = (\mathbf{x}_{p\sigma\theta}(j), \mathbf{y}_{p\sigma\theta}(j))$ (Fig. 3) is the intersection between the j-th laser–beam line with the equation $y_R = \tan(\boldsymbol{\theta}_s(j) + \sigma_{\theta_j}) * x_R$ and the environment line with the parameters r and ψ (3). $\mathbf{y}_{\sigma\theta}(j)$ is the y_R coordinate of the point on the environment line (r, ψ) at $\mathbf{x}_p(j) + \Delta d_{\sigma\theta}(j) \cos \boldsymbol{\theta}_s(j)$ coordinate (Fig. 3). The laser–beam angular noise with standard deviation σ_{θ_j} is here considered as error on the distance measurement $\mathbf{d}_s(j)$ with the standard deviation $\Delta d_{\sigma\theta}(j)$ (30), since in the algorithm the laser–beam angle $\boldsymbol{\theta}_s(j)(0°, \ldots, 180°)$, not the true laser beam angle $\boldsymbol{\theta}(j)$, is used. In Eq. (30) the point $P1 = (\mathbf{x}_p(j), \mathbf{y}_p(j))$ (Fig. 3) corresponds to the laser–beam angle $\boldsymbol{\theta}_s(j)$ and point $P2 = (\mathbf{x}_{p\sigma\theta}(j), \mathbf{y}_{p\sigma\theta}(j))$ corresponds to the angle $\boldsymbol{\theta}(j) + \sigma_{\theta_j}$. $(\Delta d_{\sigma\theta}(j) \sin \boldsymbol{\theta}_s(j) + (\mathbf{y}_p(j) - \mathbf{y}_{\sigma\theta}(j)))^2$ is the transformation of the standard deviation $\Delta d_{\sigma\theta}(j)$, which is in the direction of the j-th laser beam, to a variance in the vertical direction with regard to the estimated environment line segment (r, ψ) (Fig. 3). This variance depends on the standard deviation of the laser–beam angular noise σ_{θ_j}, the laser–beam angle $\boldsymbol{\theta}_s(j)$ and the slope of the estimated line ψ. All of these n variances that refer to a certain angle $\boldsymbol{\theta}_s(j)$ are averaged to obtain the relevant part of the vertical error variance var($\mathbf{y}(j)$), which corresponds to the error on the laser–beam angle $\boldsymbol{\theta}_s(j)$. If condition (10) is satisfied all the line-segment points were rotated for the angle $-\frac{\pi}{2}$. LRF's beam angles $\boldsymbol{\theta}_s(j)$ must then be replaced with the rotated angles $\boldsymbol{\theta}_s(j) - \frac{\pi}{2}$ in Eqs. (28) and (29). The estimated line parameter ψ must also be replaced with the rotated parameter $\psi - \frac{\pi}{2}$ in these equations. The accuracy of the vertical-error variance var($\mathbf{y}(j)$) estimation (28) depends on the accuracy of the LRF's variances $\sigma_{d_j}^2$ and $\sigma_{\theta_j}^2$ estimation. More accurate estimation of the variance var($\mathbf{y}(j)$) (16) yields a better estimation of the line parameters' variances var(r) (20), var(ψ) and the covariance cov(r, ψ). The variance of each LRF's distance-measurement error $\sigma_{d_j}^2$ and variance of each laser–beam angle error $\sigma_{\theta_j}^2$ must be a priori given from the previously estimated LRF's noise model.

The variances and the covariance of the line parameters can also be estimated with the already-described method resulting from orthogonal LSQ. The uncertainties at the input of this method are the variance of the LRF's distance-measurement error $\sigma_{d_j}^2$ and the variance of the LRF's beam-angle error $\sigma_{\theta_j}^2$. Let us consider the case with zero standard deviation $\sigma_{\theta_j} = 0$ rad (Table 4) and that σ_{d_j} is not a priori known from the LRF's noise model for the sake of comparison of both methods. $\sigma_{d_j}^2$ (Fig. 3) can then be estimated out of the line-segment points as follows

$$\sigma_{d_j}^2 = \frac{1}{n-1} \sum_{j=1}^{n-1} \left(\mathbf{d}_s(j) - \sqrt{\mathbf{x}_p(j)^2 + \mathbf{y}_p(j)^2} \right)^2, \tag{31}$$

where $\sigma_{d_j}^2$ is the same for all the LRF's distances $\mathbf{d}_s(j)$ that correspond to the line-segment points.

In Tables 4 and 5 the results of the experiments are shown where the standard deviations and the covariance of the line parameters r and ψ were calculated with the method resulting from classic LSQ as $\sigma_r = \sqrt{\text{var}(r)}$, $\sigma_\psi = \sqrt{\text{var}(\psi)}$ and cov(r, ψ) (20) and with the method resulting from orthogonal LSQ (21) as $\sigma_r = \sqrt{\mathbf{C}^*(1,1)}$, $\sigma_\psi = \sqrt{\mathbf{C}^*(2,2)}$ and cov(r, ψ) $= \mathbf{C}^*(2,1)$. All three environment line segments (Table 3) and both noise cases were considered. The experiment to calculate these two variances and the covariance for each set of line-segment points was repeated many ($N_{tr} = 10\,000$) times with the same data sets of line-segment points and variances $\sigma_{d_j}^2$ and $\sigma_{\theta_j}^2$ of the LRF's noise model as in the experiments for calculating the reference standard deviations σ_{r_u}, σ_{ψ_u} and the covariance cov(r_u, ψ_u). The variances σ_r and σ_ψ and the covariance cov(r, ψ) are different in each experiment due to the influence of the LRF's distance- and angular-measurement noise (26).

To represent the accuracy of the methods resulting from cLSQ and oLSQ the mean and standard deviation of the $N_{tr} = 10\,000$ calculated standard deviations σ_r and σ_ψ and covariances cov(r, ψ) are shown in Tables 4 and 5, where standard deviations $std(\sigma_r)$, $std(\sigma_\psi)$ and std(cov(r, ψ)) are calculated according to means $mean(\sigma_r)$, $mean(\sigma_\psi)$ and $mean$(cov(r, ψ)), respectively. $mean$(.) and std(.) of the variances σ_r and σ_ψ and covariances cov(r, ψ) in Tables 4 and 5 refer to the case where var($\mathbf{y}(j)$) (cLSQ method) and σ_{d_j} (oLSQ method) were estimated from the LRF's points during each experiment. *a-priori* : $mean$(.) and *a-priori* : std(.) of the variances σ_r and σ_ψ and covariances cov(r, ψ) in Tables 4 and 5 refer to the case where var($\mathbf{y}(j)$) (cLSQ method) and σ_{d_j} (oLSQ method) were estimated from the a-priori known variances of the LRF's noise model $\sigma_{d_j}^2$ and $\sigma_{\theta_j}^2$. Since the variance $\sigma_{d_j}^2$ (31) (calculated from the line-segment points) can only be estimated from the LRF's points in the noise-free case with $\sigma_{\theta_j} = 0$ rad, fields in Table 5 which correspond to the orthogonal LSQ method and the noise-case with $\sigma_{\theta_j} = 0.0017$ rad are empty.

If the variance var($\mathbf{y}(j)$) (classic LSQ, both noise cases) and the variance $\sigma_{d_j}^2$ (orthogonal LSQ, noise case $\sigma_{\theta_j} = 0$ rad) are estimated from the line-segment points, the standard deviations of the line parameters σ_r, σ_ψ, and the covariances cov(r, ψ) are close to the corresponding reference standard deviations σ_{r_u}, σ_{ψ_u} and to the corresponding reference covariance cov(r_u, ψ_u). The accuracy of the variances var($\mathbf{y}(j)$) and $\sigma_{d_j}^2$ depends on the number of line-segment points from which they are estimated. Consequently, the accuracy of the standard deviations σ_r, σ_ψ, and the covariances cov(r, ψ) also depends on the number of line-segment points. A more accurate variance of the LRF's distance-measurement error $\sigma_{d_j}^2$ and variance of the laser–beam angular error $\sigma_{\theta_j}^2$ can be a priori given from the previously estimated LRF's noise model. The variance var($\mathbf{y}(j)$) for the method resulting from classic LSQ can then also be more accurate if estimated from the variances $\sigma_{d_j}^2$ and $\sigma_{\theta_j}^2$ (28). In experiments to estimate the

standard deviations σ_r, σ_ψ, and the covariances cov(r, ψ) the a-priori known variances $\sigma_{d_j}^2$ and $\sigma_{\theta_j}^2$ were set to the true variances (30 mm)^2 and (0.0017 rad)^2 or 0 rad^2, respectively. This results in the σ_r, σ_ψ and cov(r, ψ) estimated with the method resulting from classic and orthogonal LSQ being closer to the corresponding reference standard deviations r_u, ψ_u and to the corresponding reference covariance cov(r_u, ψ_u) compared to the case where $\sigma_{d_j}^2$ (oLSQ, $\sigma_{\theta_j}^2 = 0$ rad^2) and var($\mathbf{y}(j)$) (cLSQ) were estimated from the line segment points. The accuracy of the standard deviations σ_r, σ_ψ and the covariance covar(r, ψ) of each method is in the noise case with $\sigma_{\theta_j} = 0.0017$ rad (Table 5) practically on the same level as in the noise-free case with $\sigma_{\theta_j} = 0$ rad (Table 4).

4. Conclusion

This paper has proposed a method for estimating the output-noise covariance matrix in the EKF-based localization of a mobile robot equipped with a LRF sensor in an environment described with line segments. To solve the localization or SLAM problem the parameters and their covariance matrices must be calculated in each environment scan for all the observed environment line segments. The output-noise covariance matrix of the EKF is composed of the covariances of the environment-line parameters. The method for estimating these covariances was derived from the classic LSQ method and has been proven to be computationally more efficient than the method resulting from orthogonal LSQ. The use of classic LSQ instead of orthogonal LSQ reduces the number of computations in the process of estimating the parameters and their covariance matrices of all the observed environment line segments by up to about 5.6 times in the noise case with nonzero angular variance and by up to about 4.7 times in the noise case with zero LRF's angular variance. Since a SLAM algorithm is the extension of a localization algorithm, the reduction of the number of computations (if using classic LSQ instead of using orthogonal LSQ) would in the SLAM algorithm be exactly the same (from the absolute point of view) as in the localization algorithm. SLAM and localization are both real time algorithms, but SLAM is known as a computationally much more complex algorithm than the localization algorithm. Therefore, the computational efficiency of the same algorithms that are used in a SLAM and a localization algorithm plays a more important role in the SLAM than in the localization algorithm.

A statistical analysis of the accuracy of both methods using the simulated measurements of the LRF sensor was also performed. The accuracy of the covariances of the line parameters estimated with the method resulting from classic LSQ depends on the number of line-segment points from which these covariances are estimated. The accuracy of the method resulting from orthogonal LSQ depends on the accuracy of the variance of the LRF's distance-measurement error and the variance of the laser–beam angle error which must be a priori given from a LRF's noise model. At each method the estimated standard deviations and covariance of the calculated line parameters were in the simulated experiments close to the reference standard deviations and covariance. In the simulated experiments the accuracy of each method is in the noise case with nonzero LRF's angular variance practically on the same level as in the noise case with zero LRF's angular variance. This paper has focused on solving the problem of mobile-robot localization. Considering all the benefits of the presented solution, the localization algorithm will be extended into the SLAM algorithm in future work.

References

[1] Bailey T, Durrant-Whyte H. Simultaneous localization and mapping (SLAM): Part II. IEEE Robotics & Automation Magazine 2006;13(3):108–17.

[2] Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: Part I. IEEE Robotics & Automation Magazine 2006;13(2):99–110.

[3] Thrun S. Robotic mapping: A survey. In: Lakemeyer G, Nebel B, editors. Exploring artificial intelligence in the new millenium. Morgan Kaufmann; 2002.

[4] Tomatis N, Nourbakhsh I, Siegwart R. Hybrid simultaneous localization and map building: A natural integration of topological and metric. Robotics and Autonomous Systems 2003;44(1):3–14.

[5] 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. Proceedings, vol. 2, 2004. p. 1065–70.

[6] Borges G, 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—Volume 1, vol.1, 2000. p. 1441.

[7] Zhang X, Rad AB, Wong Y-K. A robust regression model for simultaneous localization and mapping in autonomous mobile robot. Journal of Intelligent and Robotic Systems 2008;53(2):183–202.

[8] Latecki L, Lakaemper R, Sun X, Wolter D. Building polygonal maps from laser range data. In: ECAI int cognitive robotics workshop. 2004.

[9] 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. pp. 3317–22.

[10] Rofer T. Using histogram correlation to create consistent laser scan maps. In: IEEE/RSJ international conference on intelligent robots and system, vol. 1, 2002. pp. 625–30.

[11] Crowley J, Wallner F, Schiele B. Position estimation using principal components of range data. In: 1998 IEEE international conference on robotics and automation, 1998. Proceedings. 1998. pp. 3121–8.

[12] Baltzakis H, Trahanias P. Hybrid mobile robot localization using switching state-space models. In: IEEE international conference on robotics and automation, 2002. Proceedings. 2002. pp. 366–73.

[13] 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. pp. 1929–34.

[14] Teslić L, Klančar G, Škrjanc I. Kalman-filtering-based localization of a mobile robot with a lrf in a simulated 2D environment. In: The 14th IEEE Mediterranean electrotechnical conference, 2008. pp. 316–22.

[15] Pfister S, Roumeliotis S, Burdick J. Weighted line fitting algorithms for mobile robot map building and efficient data representation. In: IEEE international conference on robotics and automation, 2003. Proceedings, 1, 2003. pp. 1304–11.

[16] 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. 2005. pp. 2041–6.

[17] Borges G, Aldon M-J. Line extraction in 2D range images for mobile robotics. Journal of Intelligent and Robotic Systems 2004;40(3):267–97.

[18] 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. 2004. pp. 656–61.

[19] Jensfelt P, Christensen H. Pose tracking using laser scanning and minimalistic environmental models. IEEE Transactions on Robotics and Automation 2001; 17(2):138–47.

[20] Arras K, Siegwart R. Feature extraction and scene interpretation for map-based navigation and map building. In: Proceedings of SPIE, mobile robotics XII. vol. 3210. 1997. pp. 42–53.

[21] 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. Autonomous Robots 2008;24(1):13–27.

[22] Chong KS, Kleeman L. Accurate odometry and error modeling for a mobile robot. Tech. rep. Intell. Robot. Res. Centre, Dept. Elect. Comput. Syst. Eng., Monash Univ., Clayton, Australia. 1996.

[23] Arras KO, Tomatis N, Jensen BT, Siegwart R. Multisensor on-the-fly localization: Precision and reliability for applications. Robotics and Autonomous Systems 2001;34(2-3):131–43.

[24] Anousaki G, Kyriakopoulos K. Simultaneous localization and map building of skid-steered robots. IEEE Robotics and Automation Magazine 2007;14(1): 79–89.

[25] SICK, Technical description, LMS200/211/221/291 laser measurement systems.

[26] Arras KO. Feature-based robot navigation in known and unknown environments, Ph.D. thesis. Swiss Federal Institute of Technology Lausanne. 2003.

[27] Nguyen V, Gächter S, Martinelli A, Tomatis N, Siegwart R. A comparison of line extraction algorithms using 2D range data for indoor mobile robotics. Autonomous Robots 2007;23(2):97–111.

[28] 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, 1993. pp. 791–8.

[29] Schiele B, Crowley JL. A comparison of position estimation techniques using occupancy grids. In: IEEE conference on robotics and autonomous systems, 1994.

[30] Giesler B, Graf R, Dillmann R, Weiman C. Fast mapping using the log-hough transformation. In: IEEE/RSJ international conference on intelligent robots and systems, 1998. Proceedings. vol. 3, 1998. pp. 1702–7.

[31] Weingarten J, Gruener G, Siegwart R. Probabilistic plane fitting in 3D and an application to robotic mapping. In: IEEE international conference on robotics and automation, 2004. Proceedings. vol. 1, 2004. pp. 927–32.