Robotica (2008) volume 26, pp. 129–141. © 2007 Cambridge University Press
doi:10.1017/S0263574707003657 Printed in the United Kingdom

Mobile robot localization using the Hausdorff distance

F. Donoso-Aguirre, J.-P. Bustos-Salas, M. Torres-Torriti* and A. Guesalaga

Department of Electrical Engineering, Pontificia Universidad Católica de Chile, Vicuña Mackenna 4860, Casilla 306-22, Santiago, Chile.

(Received in Final Form: June 4, 2007. First published online: July 12, 2007)

SUMMARY

This paper presents a novel method for localization of mobile robots in structured environments. The estimation of the position and orientation of the robot relies on the minimisation of the partial Hausdorff distance between ladar range measurements and a floor plan image of the building. The approach is employed in combination with an extended Kalman filter to obtain accurate estimates of the robot's position, heading and velocity. Good estimates of these variables were obtained during tests performed using a differential drive robot, thus demonstrating that the approach provides an accurate, reliable and computationally feasible alternative for indoor robot localization and autonomous navigation.

KEYWORDS: Mobile robot localization; Hausdorff distance; Map-matching.

1. Introduction

Fully autonomous mobile robots rely significantly on self-localization techniques to accomplish navigation tasks.[1,2,3] It is well known that methods based on propioceptive sensors alone, such as encoders or inertial measurement units, cannot robustly solve the localization problem.[4] This is because errors from odometry readings accumulate, leading to unbounded position errors. In view of this limitation, several alternatives to dead reckoning techniques have been developed employing exteroceptive sensors, such as sonar,[5,6,7] ladar,[8,9,10] visual sensors,[11] or their combination through sensor fusion techniques.[12,13,14] In general, these techniques involve a process commonly known as *scan matching* or *model matching*, which consists of matching features extracted from sensor measurements to features in a global map of the environment.

This paper presents a novel approach for mobile robot localization and pose estimation based on matching *raw* ladar range information to a relatively simple user-created floor plan of a building. Although the idea of matching raw sensor measurements directly to a model of the environment is not new (see, for example, the work by R. Chatila and J.-P. Laumond,[15] and J. Crowley *et al.*[16]), our approach differs from exising methods in that it relies on the computation of a set of transformations (translation, rotation and scaling) that minimise the partial Hausdorff distance (HD) between the observations and the model as proposed by W. Rucklidge[17] or D. Sim and R. Park[18] for 2D pattern matching. The localization procedure also draws on ideas from our earlier work on radar-based ship positioning.[19]

The contribution of this approach is that it does not require a complex perception process to extract the features of the environment, such as different forms of line detection and association procedures predominant in vision-based approaches; the reader is referred to the survey paper by G. DeSouza and A. Kak[11] for an exhaustive review on vision-based navigation techniques. Furthermore, the experimental results obtained using a differential-drive robot demonstrate that the method can be implemented to perform in real time and that it is robust to sensor noise and dynamic environment perturbations, such as people crossing the sensor's field of view or arbitrary position changes of small furniture, such as chairs or cabinets.

The only assumptions required by the proposed approach are that the dimensions of the rooms in the building do not exceed the sensor's maximum scanning range and that all the rooms have a uniquely identifiable shape. These assumptions are not really restrictive, since they are often satisfied by existing commercial range sensors and regular home or indoor office environments. On the other hand, it is possible to combine the proposed method and the traversed path history to deal with ambiguous or out of scan-range situations.

The paper is organized as follows. Section 2 reviews the existing localization approaches. Section 3 presents the mathematical background associated to the HD and explains how it can be employed to measure the dissimilarity between two shapes. The robustness and accuracy of the approach is demonstrated in Section 4 using simulated data. Section 5 presents the sensor model and explains the solution of the localization problem using the HD. Experimental results confirming the good performance of the proposed approach applied in a real environment are presented in Section 6. Finally, Section 7 presents the conclusions of this work and discusses some aspects of ongoing and future research.

2. Existing Localization Approaches

A large number of localization techniques have been proposed in the last few decades. These approaches can be classified according to different criteria, such as sensor type, environment of applicability, map type, supporting mathematical model, or globality of the solution, to name a few. J.-S. Gutmann *et al.*[20] suggest three basic localization

* Corresponding author. E-mail: mtorrest@ing.puc.cl

approaches: behavioral-based, landmark-based, and those based on scan-matching techniques. This section will focus on scan-matching techniques for reasons of brevity and because behavioral approaches have limited localization abilities, while landmark-based techniques often require modifying the environment to provide reliable landmarks.[20] The reader is referred to the work by S. Thrun *et al.*,[1] D. Filliat and J.-A. Meyer,[21] G. Dudek and M. Jenkin,[2] C. F. Olson,[22] J. Borenstein *et al.*[3] and R. Talluri and J. Aggarwal[33] for in-depth reviews of the different localization techniques.

An important characteristic of dense sensor-matching techniques is that they rely on probabilistic models to take into account not only sensor noises and process disturbances, but also uncertainty in the environment. The general formulation of the robot localization problem in a probabilistic framework can be stated as an estimation problem in which the state of the robot (typically given by its position, velocity, heading and rate of turn) must be inferred at every time-step from the knowledge about its initial state and all measurements up to the current time. More formally, let $\mathbf{x}_k \stackrel{def}{=} [x_k, \dot{x}_k, y_k, \dot{y}_k, \theta_k, \dot{\theta}_k]^T$ denote the state of the robot at a time instant $t_k = kT$, $k = 0, 1, 2, \ldots$, where x_k, y_k are the robot's global position coordinates in the Cartesian plane, $\dot{x}_k$, $\dot{y}_k$ are the robot's velocity components along the x and y coordinates, θ_k is the robot's orientation, $\dot{\theta}_k$ is the turning rate, and T is the sampling period. Denote the set of all sensor measurements up to time t_k by $Z_k \stackrel{def}{=} \{\mathbf{z}_i, i = 1, 2, \ldots, k\}$. The robot localization problem can then be defined as the problem of finding a state estimate $\hat{\mathbf{x}}_k$, which maximizes the posterior density $p(\mathbf{x}_k|Z_k)$ of the current state conditioned over all measurements. To this end, it will be necessary to recursively compute at each time instant k, the posterior probability density $p(\mathbf{x}_k|Z_k)$ in two steps:

- Prediction step: In this first step, the *motion model* of the robot is employed to *predict* its current position in the form of a predictive probability density function $p(\mathbf{x}_k|Z_{k-1})$. The motion model is defined by a conditional density function $p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ assuming that the current state $\mathbf{x}_k$ is Markovian (i.e. only depends on the previous state $\mathbf{x}_{k-1}$) and the control input $\mathbf{u}_{k-1}$. The predictive density is computed by integration:

$$p(\mathbf{x}_k|Z_{k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) p(\mathbf{x}_{k-1}|Z_{k-1})\, d\mathbf{x}_{k-1}. \quad (1)$$

- Update step: The second step involves incorporating information from the sensors using a *measurement model* to obtain the posterior density $p(\mathbf{x}_k|Z_k)$. The usual assumptions are that the measurement $\mathbf{z}_k$ at the kth time instant is conditionally independent of previous measurements Z_{k-1} given $\mathbf{x}_k$, and that the measurement model is given by a likelihood $p(\mathbf{z}_k|\mathbf{x}_k)$, which expresses the probability of observing $\mathbf{z}_k$ given a predicted robot location $\mathbf{x}_k$. Using the Bayes rule, the posterior density of the current state $\mathbf{x}_k$ given all measurements Z_k is obtained as:

$$p(\mathbf{x}_k|Z_k) = \frac{p(\mathbf{z}_k|\mathbf{x}_k) p(\mathbf{x}_k|Z_{k-1})}{p(\mathbf{z}_k|Z_{k-1})}. \quad (2)$$

Considering this Bayesian formulation of the localization problem, the existing algorithms can be classified into the following general categories depending on the proposed representation of the posterior density (2):

- Kalman filtering approaches: In this case, the motion and measurement model, as well as the initial state, are assumed to be Gaussian. Hence, Eq. (2) remains Gaussian at all times. Furthermore, linearization of the motion and measurement equations yields closed-form expressions for Eqs. (1) and (2), which constitute the classical Kalman filter.[6,24–26]
- Markov localization: This approach is based on the computation of a discrete approximation of a probability distribution over all possible states. The distribution also evolves according to Eqs. (1) and (2). It is possible to distinguish two types of Markov localization strategies depending on whether the state space of possible robot positions is represented in terms of the environment's topological structure or using a grid. *Topological Markov localization* approaches are often employed in landmark-based navigation strategies.[27–31] *Grid-based Markov localization* methods involve discretizing the state space and approximating the posterior density in Eq. (2) by, for example, piece-wise constant functions.[32–34]
- Sampling-based methods: These approaches represent the posterior density $p(\mathbf{x}_k|Z_k)$ by a set of samples that are randomly drawn from it.[35–37] It is posible to reconstruct the density from the samples using a histogram or kenel-based density estimation techniques. Recursive algorithms to compute the samples have been proposed by several authors. These algorithms are known as bootstrap filters,[38] Monte Carlo filters[39] condensation algorithms,[40,41] or generically as particle filters.[42]

There exist other localization methods that employ probabilistic approaches for sensor and motion modelling which are worth mentioning, but that do not fit in the previous categories as they do not rely on standard Bayesian techniques. The first of such approaches is that of H. Moravec and A. Elfes,[43] who proposed a map-based technique in which the map is defined in terms of an occupancy-grid containing the probabilities of grid elements being occupied or not. The occupancy-grid method differs from the grid-based Markov localization approaches in that it does not make explicit use of conditional probabilities over previous measurements and robot actions. Nonetheless, this approach is important as it opened the way to the more advanced grid-based Markov localization techniques. More recently, P. Jensfelt and K. Kristensen[44] proposed an approach known as multiple-hypothesis localization based on multiple-hypothesis tracking techniques to exploit the accuracy of Kalman filtering localization methods and allow the representation of non-Gaussian probability distributions for the robot's state as a mixture of Gaussians. Alternative refinements of localization methods based on Monte Carlo sampling methods and multiple-hypothesis tracking can be found in the papers by J.-S. Gutmann and D. Fox,[45] and K. Kristensen and P. Jensfelt.[46] It has been demonstrated that grid-based Markov localization is in general more robust than approaches using Kalman filters, but that the latter can be more accurate than the former.[20] On the other hand, Markov localization techniques tend to be computationally more

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

demanding than localization strategies based on Kalman filtering. Recent comparisons indicate that approaches based on Monte Carlo localization or multi-hypothesis localization are both accurate and robust, performing slightly better at a reasonable computational cost.[45,46] For detailed comparative studies, the reader is referred to the work by B. Schiele and J. L. Crowley,[24] J.-S. Gutmann *et al.*,[20] J.-S. Gutmann and D. Fox,[45] K. Kristensen and P. Jensfelt,[46] G. Schaffer *et al.*[47] and L. Marchetti *et al.*[48]

On the same level of importance as the probabilistic approach for tracking the robot's pose is the process of interpreting its sensors' raw measurements. This process is a fundamental part of the measurement model and its role is to extract pose information from the raw data provided by the sensors. In contrast to motion models, which in general are one of the few possible different formulations of the kinematic and dynamic equations for a given robot, the measurement model, and more specifically the process of extracting meaningful information from raw sensor data, can take a variety of forms for a given sensor. Thus, another key aspect that distinguishes the different existing localization methods is the way pose information is obtained from raw measurements. In broad terms, the problem of interpreting measurements to quantify changes in a robot's pose is a problem of finding displacements or geometrical misaligments between current and previous measurements or between current measurements and a map of the environment. The techniques to find the correspondances and compute the degree of misalignment between current scan features and reference features are commonly known as *scan* or *map-matching techniques*. Most of the scan-matching methods originated from template or shape-matching techniques[49] conceived for computer vision applications. The different scan-matching approaches for mobile robot localization can be classified into the following general categories[25]:

- Point-to-segment: This technique relies on asigning current scan points to an existing model consisting of line segments.[8] It is best suited for polygonal environments.
- Point-to-point: This strategy computes changes in pose by finding the correspondance between each point in the current scan and each point in a previous scan. Since this technique does not require the extraction of geometric features from the raw measurements, it is applicable to nonpolygonal environments. The point-to-point correspondence approach was first employed for mobile robot localization by F. Lu and E. Milios[50,51] and is arguably the most popular technique.[25,52–54] Most point-to-point correspondance methods employ the Iterative Closest Point[55] (ICP) algorithm or some variant.[56–58] More recently, A. Diosi and L. Kleeman[59] have proposed a method called Polar Scan Matching, which is faster than ICP since the point-to-point correspondances are sought by matching points with the same bearing in the native polar coordinates of the sensor rather than in a Cartesian coordinate frame.
- Feature-to-feature: These approaches are based on representing the measurements in terms of their properties, which may not necessarily be of geometrical nature. For example, Weiss *et al.*[9,60] proposed a method based on the computation of the cross-correlation between the histograms of the scanned distances and those of the reference scan. However, this particular approach is mostly limited to polygonal environments with perpendicular walls. Modifications of this approach are possible for nonperpendicular environments.[25] Other approaches that do not require explicit geometrical features are those that employ occupancy grids and compute 2D cross-correlations between the scanned grid and the reference map in a manner similar to template-matching in image processing, i.e. by multiplying ovelapping grid cells and summing the result of each multiplication.[30,61,62] Another approach that is as common as the ICP is to perform line-to-line matching.[10,14,25,63,64] Several variants of the latter exist and differ in the way lines are extracted from the raw measurements.[65] The performance of the line-to-line matching approaches is strongly related to each algorithm's ability to correctly identify the line segments. Within this class of matching approaches, it is possible to distinghish two groups. Some of the approaches perform scan-to-scan matching, i.e. the current scan is matched to a previous scan in order to determine relative displacements of the robot, while other approaches match the features extracted from the scans to the features in a previously existing map created offline or a map created by the robot while navigating the environment. The latter can thus be referred to as scan-to-map matching approaches.

The localization approach proposed in this paper may be regarded as scan-to-map matching approach with the following advantages over other methods:

- It does not require the extraction of geometrical features nor to find explicit point-to-point correspondences, which are harder to establish in the pressence of dynamic changes and disturbances of the environment.
- Unlike point-to-point methods, our approach does not have problems of converence to local minima instead of the global minimum when the initial pose estimate is not close to the actual pose.
- It can be as computationally efficient as point-to-point or line-to-line matching strategies by using the Voronoi matrix of the reference map, which can be computed offline once.
- It is not limited to polygonal environments as many point-to-point or line-to-line matching strategies.
- It is more robust to map errors, non-Gaussian disturbances and occlusions affecting the measurements than line-to-line matching approaches because it seeks to minimise a measure of dissimilarity rather than to maximise similarity between the scan and the reference data.
- It exploits the accuracy and computational efficiency of the extended Kalman filter to update the estimate of the robot's state.

3. Matching Based on the Hausdorff Distance

The computation of the Hausdorff distance (HD) is a technique to measure the degree of dissimilarity among

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

different objects. By taking two sets of points, one being the reference model and the other the actual measurements, the HD between them is small when every point in one of the sets is near to some point in the other. Given two sets, $A = \{a_1, a_2, \ldots, a_p\}$ and $B = \{b_1, b_2, \ldots, b_q\}$, the HD between A and B is defined as:

$$H(A, B) \stackrel{def}{=} \max(h(A, B), h(B, A)) \tag{3}$$

where

$$h(A, B) \stackrel{def}{=} \max_{a \in A} \min_{b \in B} \|a - b\| \tag{4}$$

is the directed HD between sets A and B.

Numerical procedures to compute this distance, sort the points in A according to their distance to the nearest point in B, and then select the farthest one as the result. For instance, if $h(A, B) = h^*$, then every point in A is at the most at a distance h^* of a point in B. The point with distance h^* is the point that most deviates from set B. Figure 1 shows a geometric representation of the HD as applied to pattern recognition. Here, sets A and B are the reference model and measurements, respectively. By rotating and translating the measurements, a satisfactory matching is obtained as the one that minimises $H(A, B)$.

In order to reduce the number of calculations, a distance transform in the form of a Voronoi matrix is computed first. By doing so, set A is processed only once. Further details can be found in the paper by W. Rucklidge.[17]

In most applications, the sets A and B are not identical, as would typically occur in the presence of occlusions, measurement noise and image distortions. The latter is particularly valid for ladar images generated through the application of a transformation of the raw measurements in polar coordinates to a set of measurements in Cartesian coordinates; thus, offsets in range will cause a shrinkage or enlargement of the objects. Sometimes these differences can also be introduced at intermediate stages, such as expansion, rotation and translation, among others. To cope with this problem, some modifications to the previous criteria based on the HD must be introduced as explained next.

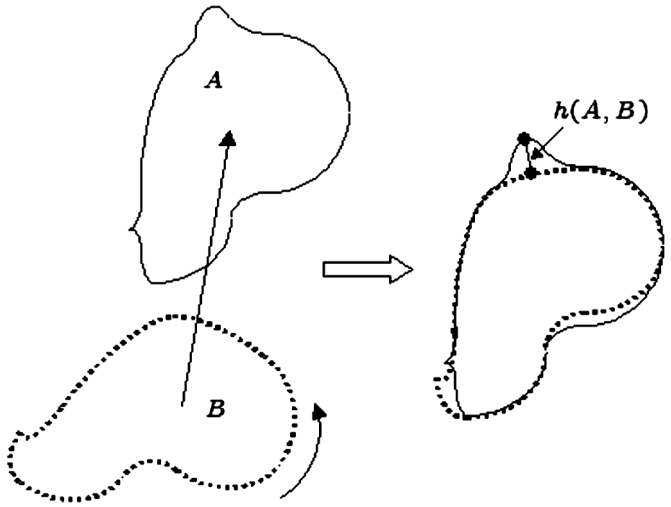

Fig. 1. Pattern matching employing the Hausdorff Distance.

3.1. Partial Hausdorff distance

The previously mentioned sources of error will generate some false-positive matches with HDs significantly larger than the one of any true-positive match. Taking advantage of the fact that the HD computation procedure determines the distance of the farthest point in B by ranking the distances of its points to points in A, a way of reducing erroneous matches is to select the Kth distance in the ranking, instead of the largest one.[17] In other words, some of the the points in B with the largest distances are ignored and only a subset is used. The HD with respect to A of this subset of B is the so-called *partial (directed)* HD. In order to formally define the partial HD, it is convenient to introduce first a mapping d_Ω:

$$d_\Omega : x \to d_\Omega(x) = \min_{\omega \in \Omega} \|x - \omega\|$$

that *measures* the distance of the closest point ω in a set Ω to some point x. Then, the partial HD of the K *best matching* points of the measurements set B to the model set A can be defined recursively for $K = q, q-1, q-2, \ldots, 2, 1$ as:

$$h_K(B, A) = \max_{b \in B^K} d_A(b) \tag{5}$$

where

$$B^K = B^{K+1} - \{b^*_{K+1}\}$$

$$b^*_K = \arg\max_{b \in B^K} d_A(b)$$

and the initial values $B^{q+1} = B$, $b^*_{q+1} = \{\emptyset\}$. It is worth noting that $h_q(B, A) = h(B, A)$, and that $h_q(B, A) \geq h_{q-1}(B, A) \geq \cdots \geq h_1(B, A)$, since $B^q \supset B^{q-1} \supset \cdots \supset B^1$. Hence, this definition automatically implies that there are K measurement points in B within a distance $h_K(B, A)$ from A (the Kth partial HD). Since more than one transformation of the image associated with the measurements set may result in similar values for $h_K(B, A)$, an effective criteria for successful matching is to minimise the average of partial HDs smaller or equal to $h_K(B, A)$ for some chosen K. This average, also known as *modified* HD,[18] is given by:

$$\bar{h}_K(B, A) = \frac{1}{K} \sum_{i=1}^{K} h_i(B, A). \tag{6}$$

Employing the modified HD ensures that more points in the image will resemble the model. For practical purposes, it is convenient to define the ratio of model points employed in the calculation of the average as $\lambda = K/q$. Then, K may be selected in terms of the ratio λ, simply as $K = \lambda q$, with $1/q \leq \lambda \leq 1$. The value of λ is found empirically as the one that minimises the matching error for a set of image and model pairs.

3.2. Matching strategy and its computational cost

The search for the optimal solution is based on the application of translation, rotation and scaling transformations to the sets

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

or images in such a way as to minimise Eq. (6), thus yielding a set of transformations that maximises the similarity between A and B. Formally, the matching problem can be stated as that of finding a transformation $T : \Omega \rightarrow T(\Omega)$ such that it minimises:

$$\min_T h_K(T(B), A). \tag{7}$$

Here, T was chosen to be $T_{\delta,\theta,\alpha} : \Omega \rightarrow \alpha\, R_\theta(\Omega + \delta)$, i.e. a transformation that first applies a translation δ in the plane, followed by a rotation of angle θ and finally a scaling transformation of magnitude α. In order to solve Eq. (7), a reliable and rapidly converging approach to minimise Eq. (6) was implemented in terms of standard gradient methods.

The computational cost of the matching strategy is the product of the computational effort involved in the calculation of the modified HD, the complexity of the minimization procedure and the number of iterations required to attain a minimum. Assuming the Voronoi matrix of the reference map has been pre-computed offline, a closed-form expression for the cost of the modified HD computation is given by:

$$C_{HD} = N_s(7C_1 + 2C_2 + C_3) + C_4 N_s \log(N_s) + C_5 \tag{8}$$

where N_s is the number of samples per scan, C_1, C_2, C_3, C_4 and C_5 are the number of CPU cycles required by additions/subtractions, transcendental functions, rounding operations, memory fetch operations and multiplications/divisions, respectively. The term $C_4 N_s \log(N_s)$ in Eq. (8) is the cost of sorting the list of partial HDs. Thus for large number of samples N_s, the complexity of the HD computation is $C_{HD} \approx O(N_s \log(N_s))$. This complexity is comparable to that of the split and merge algorithm for line extraction, which has been reported to be the fastest of the line extraction algorithms for mobile robot localization.[65]

The minimisation strategy requires the computation of the modified HD eight times, once in each direction of the eight adjacent neighbors in a Cartesian grid, in order to obtain a numerical approximation of the gradient along the x and y coordinates. Similarly, to find the proper direction of rotation that minimises Eq. (7), the modified HD needs to be computed twice. Initial iterations do not attempt to minimise the possible error in the orientation angle between the current scan and the reference map. This is because correct orientations may yield a larger modified HD than incorrect ones when the initial translation error is large. Hence, attempting to reduce the angular misaligment before having reduced the position misalignment to a minimum may result in wasted effort.

The rate of convergence will be determined by the distance function employed to compute the Voronoi matrix. Convergence in a finite number of iterations can be guaranteed if a city-block (ℓ_1) or Euclidean (ℓ_2) metric is employed.[66] In order to reduce the computational burden in calculating the modified HD, it is common to employ the city-block metric.[67] This will guarantee a constant convergence rate with respect to translation for a fixed step size. A way to reduce the number of iterations required for convergence is to adjust the step size as the solution approaches the optimal value. In our implementation, a simple mechanism was adopted in which initial steps of size $\bar{\Delta} = \Delta_x = \Delta_y > 1$ grid units are taken in the directions x and y until no further decrease of the objective function can be achieved. The minimisation process is then repeated using a step size half of that in the previous set of iterations. This action is repeated until the step size cannot be further reduced, i.e. until $\bar{\Delta} = 1$ grid unit, unless sub-pixel accuracy is sought. A similar procedure is employed to minimise the angular error and the scaling corrections employing an initial angle step size $\bar{\Delta}_\Theta$ and a scaling factor step $\bar{\Delta}_s$. Assuming that the range sensor has been properly calibrated offline, then scaling errors can be neglected and the matching problem reduces to finding optimal translation and rotation, thus reducing the computational burden. Thus, if the city-block distance is employed in the computation of the Voronoi matrix, and if the initial positional and angular errors are given by $\bar{D} = \|\hat{p} - p\|_{\ell_1}$ and $\bar{\Theta} = \|\hat{\theta} - \theta\|_{\ell_1}$, then the minimisation approach will converge in at most $N_I = N(\bar{D}, \bar{\Delta}) + N(\bar{\Theta}, \bar{\Delta}_\Theta)$ iterations, where:

$$N(D, \Delta) \stackrel{def}{=} \sum_{i=0}^{n} \text{round}\,(y_i(D, \Delta)) \tag{9}$$

and

$$y_i(D, \Delta) \stackrel{def}{=} \frac{y_{i-1}(D, \Delta) - \text{trunc}(y_{i-1}(D, \Delta))}{\frac{\Delta}{2^i}}$$

$$y_0(D, \Delta) \stackrel{def}{=} \frac{D}{\Delta}.$$

with $n = \sup\{i \in \mathbb{N} : \frac{\Delta}{2^i} \geq 1\}$, i.e. n is the largest integer which satisfies $\frac{\Delta}{2^n} \geq 1$.

Considering the total number of iterations, the total computational cost of the matching strategy satisfies

$$C_T \leq (8N(\bar{D}, \bar{\Delta}) + 2N(\bar{\Theta}, \bar{\Delta}_\Theta))C_{HD} \tag{10}$$

where C_{HD} is given in Eq. (8).

It is important to note that by Eqs. (9) and (10), the total computational cost directly depends on the initial errors $\bar{D}$ and $\bar{\Theta}$, as well as the number of range samples per scan. This cost may initially be large, since the true robot's position and orientation may have been badly guessed. However, if the initial guess required by the minimisation strategy is based on the predictions obtained from the robot's dynamic model and the knowledge of the steering inputs through an estimation filter, such as an extended Kalman filter, then subsequent iterations of the localization strategy will incur very small computational costs. This is because, given good predictions, the misalignments $\bar{\Delta}$ and $\bar{\Theta}$ will be kept small throughout the following iterations as the robot moves. Hence, the application of the strategy for real-time localization is made possible, as will be shown in Section 6.

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

Table I. Simulation results of the positioning accuracy.

Noise percentage [%]	Noise level σ [pixels]	Final match position error [pixels]	Final match heading error [°]	Initial $\bar{h}_K(A, B)$ [pixels]	Final $\bar{h}_K(A, B)$ [pixels]
0	0	0	0	165.16	0
20	10	0	0	165.02	0
20	20	0	0	165.77	0
50	10	0	0	165.27	0.66
50	20	0	0	164.50	1.22
100	10	0	0.17	164.60	4.33
100	20	1.00	0	163.03	8.42

4. Robustness and Accuracy of the Matching Method

In this section, the robustness and accuracy of the matching method based on the modified HD is tested in terms of simulated data. The simulations consider a 500 × 500 pixels reference image containing a 116 × 116 pixels square in its center. The simulated data representing the contour obtained by the ladar is generated by: (i) adding noise to a given percentage of samples of the original reference square, (ii) rotating this simulated measurements 10° counter-clockwise, and (iii) translating the set of points 150 pixels to the left and 150 pixels up. Two different parameters are adjusted for the different scenarios of measured data: (i) the percentage of outliers or measurements affected by noise and (ii) the average magnitude of the noise. The noise affecting each measurement sample, if any, is assumed to be zero-mean Gaussian with standard deviation equal to the scenario's noise level.

The results obtained throughout the simulations are summarised in Table I. The robustness of matching method based on the modified HD is apparent from this results, which show practically no variation in the position or heading error for different percentage values of noisy samples and noise levels. The quality of the match may be appreciated from Figs. 2 and 3, which show the initial measurements and their final position after matching, respectively. Figure 4 shows the convergence of the modified HD to a minimum in an almost linear fashion for all the different scenarios. The final value of the averaged partial HD increases proportionally to the noise level. However, it is worth noting that this value remains zero for the simulations in which only 20% of the samples are corrupted by noise. This is because the ratio λ for calculation of the modified HD was set at 0.7, which means that only 70% of the total number of samples that best match the reference set are used, while the remaining 30% samples that include those corrupted by noise are automatically discarded from the calculation of the modified HD. Figure 5 clearly shows that the regular HD curves have largest values beyond the threshold. While scenarios with noise percentages below 30% yield final $\bar{h}_K(A, B)$ equal to zero, scenarios with noise percentages above 30% yield final $\bar{h}_K(A, B)$ which are non-zero and are proportional to the standard deviation of the noise for a fixed threshold λ; see Table I. The results in Table I also confirm that the accuracy is high, as expected according to the theoretical result by D. P. Huttenlocher *et al.*[67] (Claim 3), which states that the matching error due to spatial sampling is at most one rasterization unit for the noise-free situation. This result may also be extended to the zero-mean Gaussian noise situation.

5. Localization Problem

For convenience of exposition, let x and y denote the position coordinates of the robot in the global 2D Cartesian frame of reference, and let θ denote its orientation with respect to the vertical axis. The first step to determine the robot's position and heading is to express each of the N raw range measurements $z_i^s = [r_i, \theta_i^s]$, $i = 1, 2, \ldots, N$, obtained in polar coordinates relative to the sensor coordinate frame O_s, as *Cartesian coordinates relative to the global coordinate frame* O_w. To this end, the following measurement model is employed:

$$z_i^w = \begin{bmatrix} (r_i^s + \beta_r)\cos(\theta_i^s + \theta + \beta_\theta) + x + \eta_x \\ (r_i^s + \beta_r)\sin(\theta_i^s + \theta + \beta_\theta) + y + \eta_y \end{bmatrix} \quad (11)$$

where η_x and η_y are assumed to be zero-mean Gaussian noises arising from errors in the matching, as will be

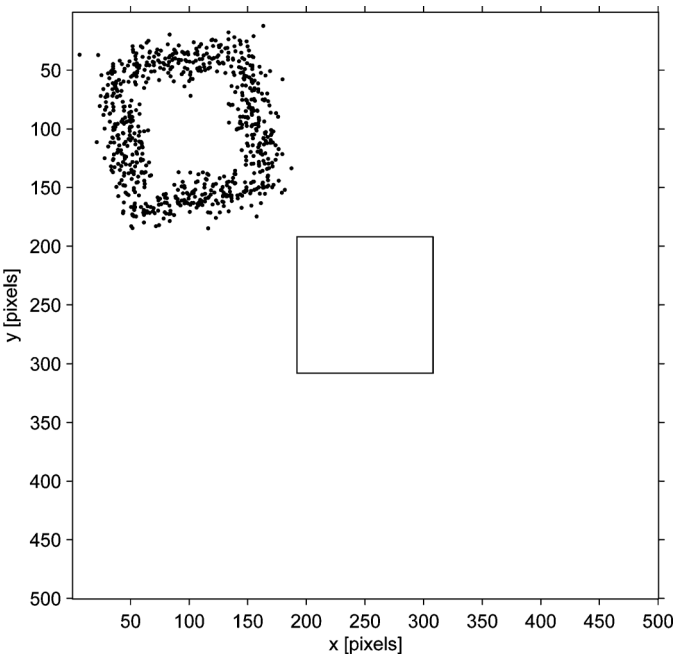

Fig. 2. Initial misalignment between simulated data (dots) and the reference data (line).

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

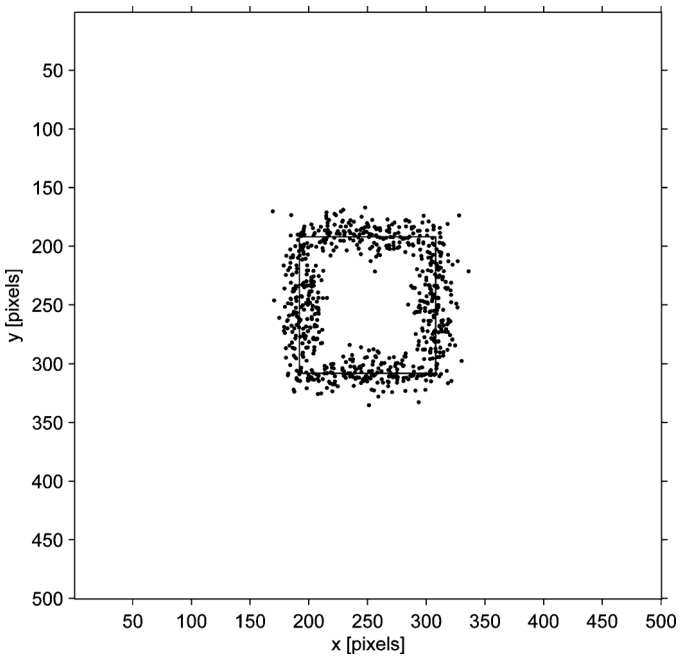

Fig. 3. Final match of simulated data (dots) and the reference data (line).

explained next. The variables β_r and β_θ are assumed to be common to all points and account for sensor measurement errors induced by biases in range and bearing, respectively. The range bias β_r results from errors in leveling the sensor (azimuth error). The bearing bias β_θ describes the rotation that suffers the measurement image due to misalignments of the sensor base with respect to the robot. Since normally ladar sensors are fixed and do not rotate with respect to the platform, it becomes difficult to determine β_θ without an additional heading sensor, such as a compass, because the main rotation of the measurement image results from the rotation θ of the robot as it moves and turns. Thus, it is assumed here that $\beta_\theta = 0$.

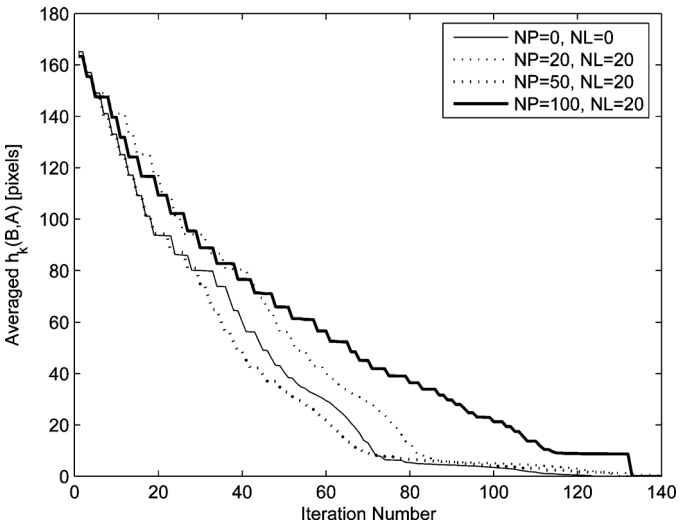

Fig. 4. Convergence of the modified Hausdorff distance for the different scenarios with percentages of noisy samples NP and noise levels NL.

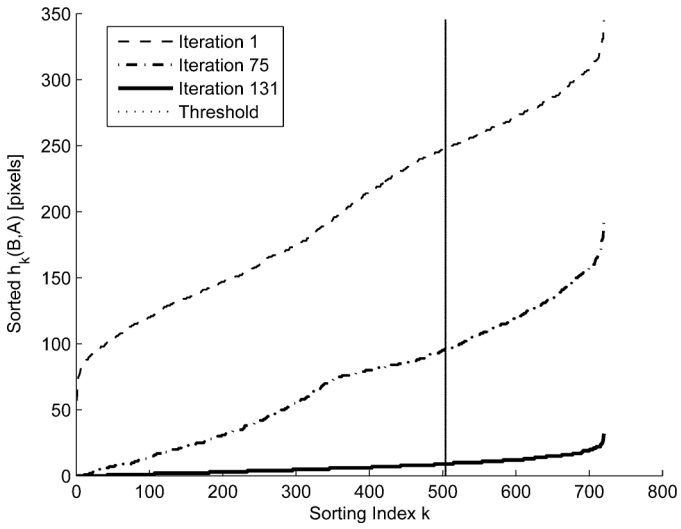

Fig. 5. Sorted Hausdorff distances for the simulation with NP = 100% and NL = 10.

Employing Eq. (11), a *measurement image* can be created from the N measurement samples by setting all pixels to zero except for those at the coordinates z_i^w, $i = 1, \ldots, N$, which must be set to one. Similarly, the CAD floor plan drawing must be rasterized, i.e. the drawing must be sampled discretely to create a *model image* in which pixels are set to one, wherever there is a geometric element, or to zero otherwise. The resolution of both (measurement and model) images should be sufficiently large in order to allow for accurate position estimates, but should not exceed a value such that each pixel represents a dimension far smaller than the precision of the sensor itself.

Labeling the model image by A and the measurement image by B, the matching algorithm presented previously can be employed to estimate the values of x, y and θ, as well as η_r, through the solution of Eq. (7), which yields a transformation T^* that minimises Eq. (6). The transformation T^* can be parameterized by a translation $\delta = (x, y)$, a rotation by θ, and scaling factor α, such that $\alpha\, r_i^s = r_i^s + E(\eta_r)$, where $E(\eta_r)$ denotes the expected value of η_r. Thus, the solution of Eq. (6) that minimises the difference between the model and measurement image naturally results in a measurement of the robot's location and orientation. In practice, the matching process is not perfect, since the sensor and the model have finite resolutions, and because sensor measurements are always subject to clutter and occlusions, which limit the precision of the results. On the other hand, the actual transformation from polar to Cartesian coordinates in Eq. (11) also produces deformations of the objects in the image. Thus, to account for these sources of error, the noises η_x and η_y need to be included in the measurement model given in Eq. (11).

Despite these errors, the accuracy of the position and orientation estimates can be further improved using an extended Kalman filter (EKF), which also allows to obtain estimates of the robot's velocity and the sensor range bias β_r, as shown in the next section in terms of experimental results.

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

Fig. 6. Magellan differential drive robot and Sick PLS-101.

6. Experimental Results

The proposed approach for localization and pose estimation was tested using a WRI Magellan Pro® differential drive robot equipped with a Sick® PLS-101 Laser range finder, shown in Fig. 6. The sensor is set to scan 180° with a range span of 0–50 m. The range accuracy is of the order of ±0.05 m with a resolution of 0.07 m or better, while the angular resolution is 0.5°. With this configuration, the 360 samples stretching along the full distance range and angular sweep have proven sufficient to ensure a reasonably good matching and estimation accuracy.

In order to implement the EKF, the following simplified model equation is employed to describe the motion of the robot:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v}_R \\ \dot{v}_L \\ \dot{\beta}_r \end{bmatrix} = \begin{bmatrix} \frac{v_R+v_L}{2}\sin(\theta)+\xi_x \\ \frac{v_R+v_L}{2}\cos(\theta)+\xi_y \\ \frac{v_R-v_L}{L}+\xi_\theta \\ u_1+\xi_{u_1} \\ u_2+\xi_{u_2} \\ u_3+\xi_{u_3} \end{bmatrix} \stackrel{def}{=} f(x,u) \quad (12)$$

where ξ_x, ξ_y, ξ_θ, ξ_{u_1}, ξ_{u_2} and ξ_{u_3} are all assumed to be zero-mean i.i.d. Gaussian process noises. The above model corresponds to the standard kinematic model for a differential drive robot,[2] augmented to include the accelerations $\dot{v}_R$ and $\dot{v}_L$ of the left and right wheels, which are controlled through the commands u_1 and u_2, respectively. It is to be noted that using the velocities of each wheel is equivalent to including terms for the longitudinal velocity v and the angular rate of change ω of the mass center of the robot, since $v=(v_R+v_L)/2$ and $\omega=(v_R-v_L)/L$. Inertia moments and masses are not explicitly considered in the model since the EKF can automatically compensate and adjust the gain dynamically as needed. The model has also been augmented to include the unknown range bias, which is assumed to be constant throughout the experiment, and hence, the *virtual* input u_3 is set to zero.

A simple measurement model is assumed considering that the matching strategy automatically outputs the observations that feed the EKF. Therefore, the measurement model is given by:

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} x+\zeta_x \\ y+\zeta_y \\ \theta+\zeta_\theta \\ \beta_r+\zeta_{\beta_r} \end{bmatrix} \stackrel{def}{=} h(x) \quad (13)$$

where ζ_x, ζ_y, ζ_θ and ζ_{δ_r} are assumed to be zero-mean i.i.d. Gaussian measurement noises.

From Eqs. (12) and (13), the process and measurement Jacobians can be easily calculated as:

$$\nabla_x f = \begin{bmatrix} 0 & 0 & \frac{v_R+v_L}{2}\cos(\theta) & \frac{1}{2}\sin(\theta) & \frac{1}{2}\sin(\theta) & 0 \\ 0 & 0 & -\frac{v_R+v_L}{2}\sin(\theta) & \frac{1}{2}\cos(\theta) & \frac{1}{2}\cos(\theta) & 0 \\ 0 & 0 & 0 & \frac{1}{L} & -\frac{1}{L} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\nabla_u f = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\nabla_x h = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

To evaluate the performance of the proposed approach, a reference trajectory of known geometry was first defined. The experiment consisted in scanning the environment every second while the robot followed the reference trajectory at a constant velocity of 5 cm/s. Thus, the change in position between scans was about 5 cm. Figure 7 shows the model (floor plan) image to which the measurement (ladar) image must be matched. The superposition of the first measurement image to the model image in Fig. 8 clearly shows a mismatch due to the large error in the initial position and heading values, as would occur if the

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

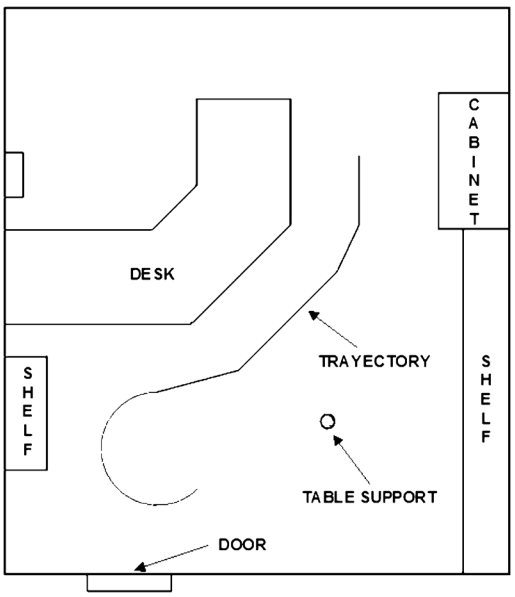

Fig. 7. Model image from the rasterization of the floor plan.

initial estimate is wrongly or poorly chosen. The initial misalignment was deliberately made larger by starting the HD-based matching procedure with a significantly larger initial estimated angle $\theta_0 = 30^\circ$, even if it was known to be around 0°. In order to find the transformation that minimises the modified HD, $\bar{h}_K(B, A)$ given in Eq. (6), the matching strategy described in Section 3 is applied with $\lambda = 0.7$, resulting in an almost perfect match, as shown in Fig. 9. This may also be corroborated in Fig. 10, which shows the convergence of the modified HD from an initial value of 47.87 to a value of 2.43 pixels after 11 iterations. The convergence of the set $\{h_K(B, A); K = 1, 2, \ldots, q\}$ of partial HDs to the resulting optimal set $\{h_K(T^*(B), A); K = 1, 2, \ldots, q\}$ is

Fig. 8. Initial misalignment of the floor plan model (thin lines) and ladar measurement (scattered points).

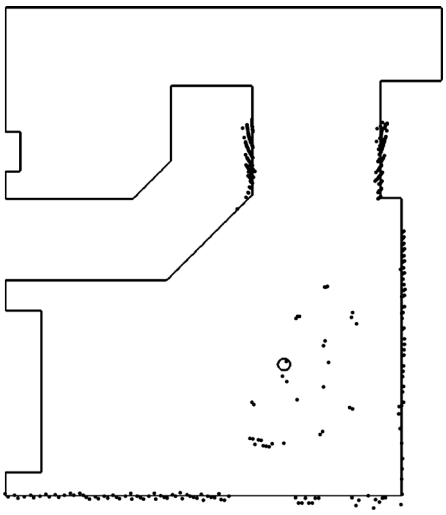

Fig. 9. Matching of the measurements to the model.

shown in Fig. 11. The latter figure clearly shows that there is a relatively small reduction in the standard *directed* HD, i.e. $h(B, A) = h_{360}(B, A)$, which decreases only from 238 to 154 in the third iteration, and none thereafter. Moreover, while the modified HD, $\bar{h}_K(B, A)$, exhibits a decrease of 94.92%, the standard HD, $h(B, A)$, has a reduction of only 35.29%. These facts support the use of the averaged partial HDs as a better matching criteria, which is more robust to outliers arising from the presence of unmodeled objects or noise, as was also demonstrated by the simulations presented in Section 4. The chosen value of $\lambda = 0.7$ implies that only 70% of the measurements with the smallest partial HDs are employed in the calculation of the modified HD; this amounts to 252 measurements of the 360 measurements that are obtained per scan, as shown by the threshold mark in Fig. 11. It is worth pointing out that the matching procedure is carried out successfully in spite of the large initial estimated state error and the presence of noise due to furniture features and chairs not considered in the original floor plan drawing. These

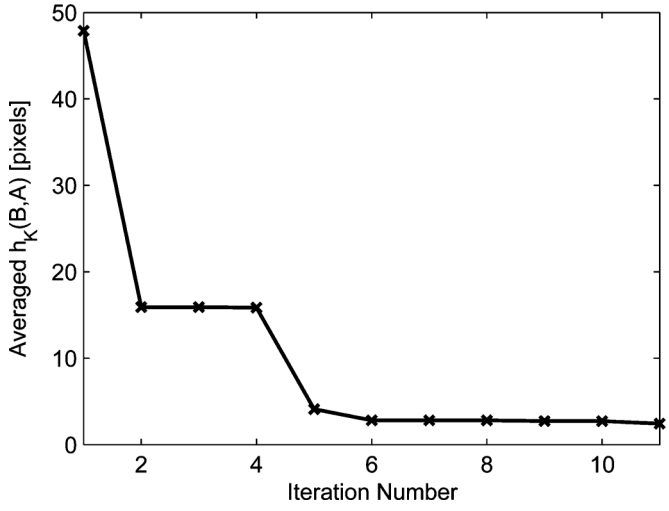

Fig. 10. Convergence of the averaged partial Hausdorff distance $\bar{h}_K(B, A)$ using $\lambda = 0.7$.

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

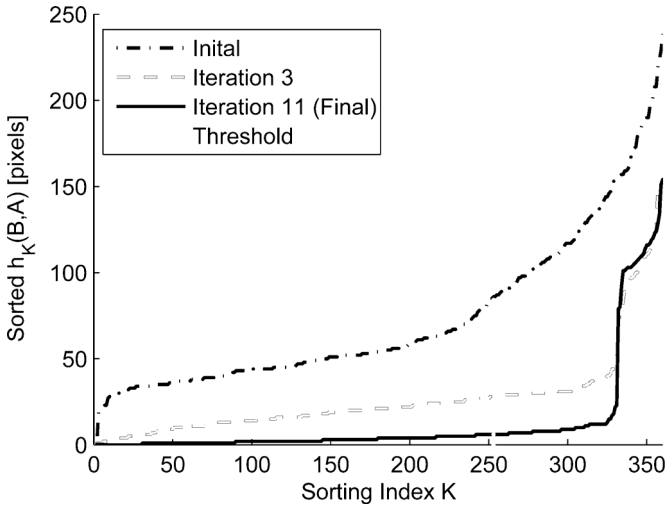

Fig. 11. Sorted partial Hausdorff distances $h_K(B, A)$ and their convergence to the optimal set $h_K(T^*(B), A)$ using $\lambda = 0.7$.

objects appear as spurious measurements mainly around the table support as can be seen in Fig. 9.

The matching procedure is repeated after every ladar scan, yielding position, orientation and radar bias measurements that are fed to the EKF, in order to obtain smoothed estimates of the robot's state vector. The estimated trajectory traversed in 100 s is shown in Fig. 12. The magnitude of the error between the estimated position and the actual trajectory coordinates is shown in Fig. 13. Similarly, the estimated heading angle is compared to the true heading that the robot should have followed in Fig. 14. The heading error shown

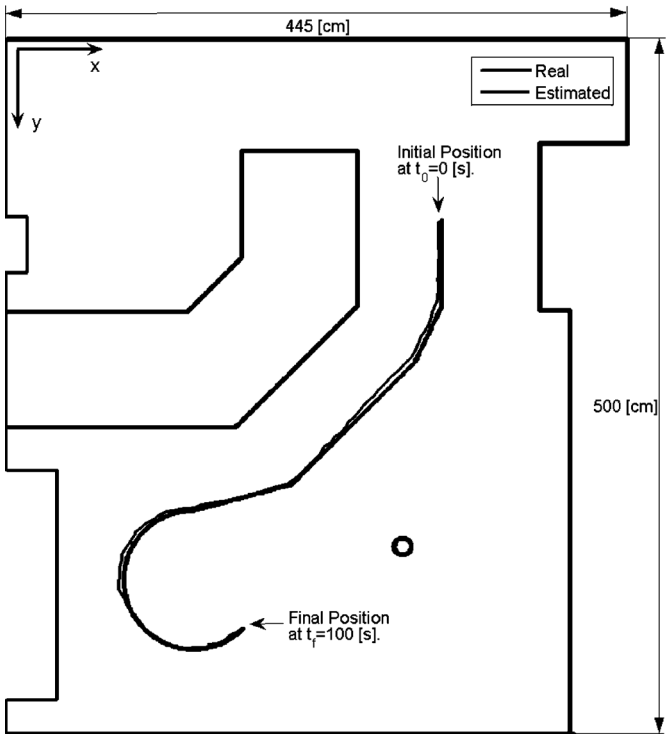

Fig. 12. Estimated trajectory (black) and reference trajectory (gray) starting from $(x_0, y_0, \theta_0) = (312\,\text{cm}, 135\,\text{cm}, 0^\circ)$ in $t_0 = 0\,\text{s}$, and ending at $(x_f, y_f, \theta_f) \approx (169\,\text{cm}, 428\,\text{cm}, 130^\circ)$ in $t_0 = 100\,\text{s}$, with constant forward speed $v \approx 5\,\text{cm/s}$.

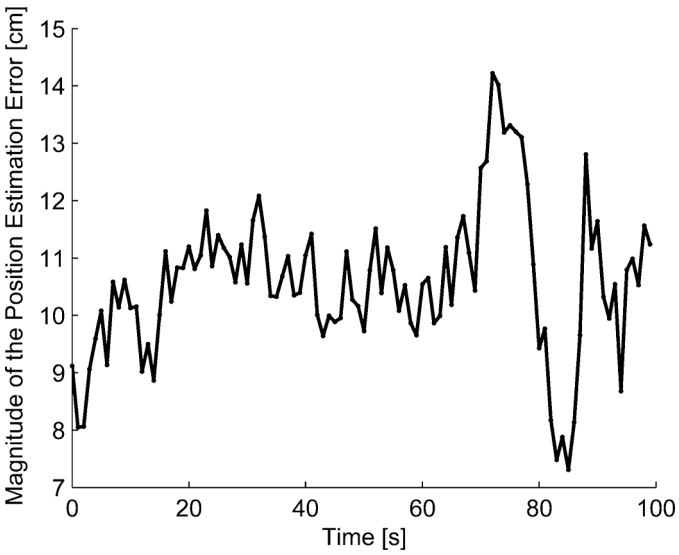

Fig. 13. Magnitude of the error between the estimated position and the real trajectory coordinates.

in Fig. 15 confirms that the estimation is relatively good for most part of the trajectory except at points in which the heading changes on $t = 13, 22, 46, 60\,\text{s}$. The heading error becomes larger when the robot starts the constant turning ratio maneuver as it follows the circular trajectory.

The velocity estimates can be derived using the model equations (12). Thus, the forward velocity estimate is given by $\hat{v}(k|k) = (\hat{x}_4(k|k) + \hat{x}_5(k|k))/2$, while the turning velocity estimate is given $\hat{\omega}(k|k) = (\hat{x}_4(k|k) - \hat{x}_5(k|k))/L$, where L is the distance between wheels, and $\hat{x}_i(k|k)$, $i = 4, 5$, are the right and left wheel estimated velocities, respectively. Since the commands issued to the robot are constant throughout the piecewise linear trajectory and the circular trajectory, the estimates remain constant and close to zero for most part of the trajectory, except at points where there are heading step changes. The turning ratio estimate is also constant, but non-zero for the circular part of the trajectory. The latter is equivalent to a ramp change in the heading as shown in Fig. 14. Similarly, the estimated range bias remained constant

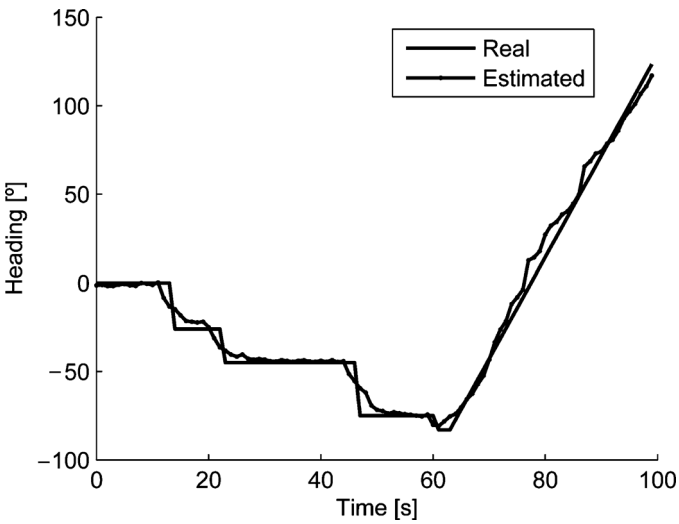

Fig.14. Estimated robot heading angle θ.

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

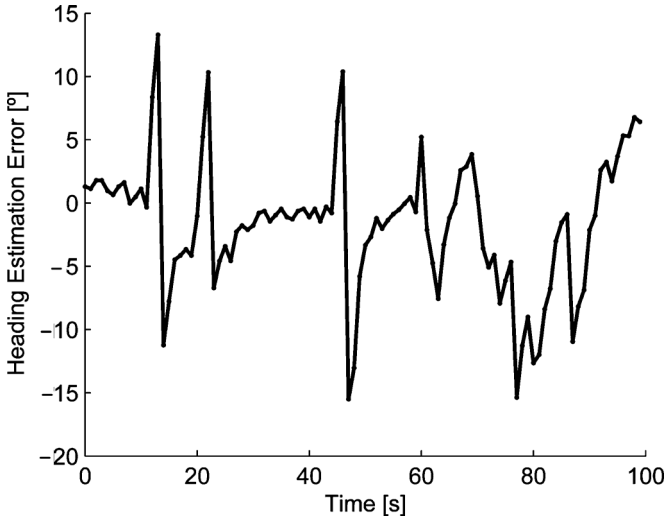

Fig. 15. Error between the estimated heading angle and the real trajectory direction.

throughout the experiment because the resolution of the rasterized model is equal to 5 cm per pixel and the resolution of the scanner is better than 7 cm, while its precision is within ±5 cm.

The algorithm was implemented in Matlab® running on a 2.8 GHz Pentium computer with 512 MB RAM. Its execution period was verified to be 0.7 ms per scan on average. This amount of processing time is larger than that of many point-to-point algorithms, such as the Polar Scan Matching approach,[59] or line extraction-based techniques, such as the Split and Merge algorithm,[65] which are among the fastest reported in the literature and should, respectively, require around 0.01 ms and 0.2 ms of processing time per scan for the same number of samples when implemented in C/C++ on a computer with similar characteristics. However, the processing time of our approach can be further improved to similar levels by implementing the algorithm in C/C++ rather than an interpreted language. This is because the complexity of our algorithm is similar to that of the best point-to-point or line-to-line localization approaches[65] as mentioned in Section 3.2.

It should be pointed out that the number of iterations required to solve the localization problem after each scan is relatively small as can be observed in Fig. 10, which shows that after the sixth iteration there is no further significant decrease of the modified HD. This is posible because the minimisation is solved starting with an initial guess equal to the position and orientation of the robot predicted by the EKF. In the experiments, it was observed that the minimisation problem is solved in less than eight iterations on average. The number of iterations can be above several hundreds only for the first run of the localization algorithm when the initial guess is far from the true robot's location. In practice, this does not constitute a real problem since the operator can set a reasonable guess of the robot's location at start up, or the robot may be allowed to explore the environment for an initially longer time. By *reasonable guess* it is meant any point in the actual room or area of the building where the robot is.

It is also important to note that even if the method is not an explicit global localization approach, its positioning accuracy does not degrade when people walk around the robot since the matching procedure based on the modified HD takes care of removing spurious measurements as explained in Section 4. On the other hand, the EKF allows to keep track of the robot's position and feed adequate position predictions to the minimisation procedure. If the sensor is almost completely blocked during several scans in such a way that there is not enough information to solve the matching problem, the localization accuracy will not necessarily degrade because under such a situation the collision avoidance schemes would be triggered and the current robot's position would be held until no obstacles appear in the robot's path. The only possible failure mode may arise when there is no unique minimum for the averaged partial HD function $\bar{h}_K(B, A)$. This ill condition may occur when the range scans do not contain enough information to solve ambiguities which may result when there are several similar sections or symmetries in the reference map. To avoid this problem, the search space must be constrained according to the initial guess from a previous prediction and, if necessary, the robot must be allowed to collect more information of the surroundings in order to perform the matching between the reference map and a compounded set of scans that solve possible ambiguities which would otherwise occur if only single scans are employed.

7. Conclusions

An approach for estimating the position, heading and velocity of a mobile robot in a structured environment was presented. The approach relies on matching ladar measurements to an image of a rasterized CAD floor plan drawing, thus freeing the robot from any dependence on landmark-based positioning systems, triangulation-based techniques relying on optic, ultrasonic or RF beacons, or from combinations of these. The latter may be difficult to employ in many indoor environments due to signal path occlusions or interferences.

Central to the matching process is the identification of a set of image transformations that minimise the average of the partial Hausdorff distances given by Eq. (6). These transformations directly translate into raw position and orientation measurements that are then passed to an EKF in order to obtain better estimates of the position, heading and velocity of the robot. The approach also allows to readily obtain the sensor's range bias as part of the matching process, and can be easily modified to obtain the sensor's bearing bias, if a heading sensor is added to the robot. The results obtained demonstrate the effectiveness of the approach as it yields rapidly converging accurate estimates. Furthermore, the approach is amenable for real-time implementation.

Continuing work is concerned with:

- Combining odometric techniques and methods for visual recognition of natural landmarks with the proposed approach in order to improve its performance in the presence of multiple minimising solutions of the modified HD, thus improving the ability of the proposed approach to robustly solve the global localization problem in

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press

ambiguous situations, for example in large buildings with several rooms or storeys with very similar floor plan geometries.
- Investigating alternatives to further reduce the computational cost of the approach. For example, by exploting the characteristics of the city-block metric used in the construction of the Voronoi matrix to reduce the computational complexity of the optimisation procedure and achieve convergence to the optimum in a single step per scan. Other reductions in the total computational cost may be achieved using multiscale techniques, such as the one proposed by Kwon *etal*.[68] to improve the processing time of the modified HD.
- Extending the approach for simultaneous localization and mapping, and exploring possible benefits and costs of using a Markovian or Monte Carlo sampling approach to update the estimate of the robot's state.

Finally, in view of the fact that the vulnerability of global positioning system to spoofing or satellite denial is a matter of particular concern in countries that do not conform to the military elite[69] since this would affect most of the modern navigation systems, the proposed approach can provide a valuable alternative to differential global positioning system for mobile ground platforms in structured environments that are equipped with range sensors, or for ship positioning and maneuvering at locations from the coast within their radar's range. In the latter application, the ship would have to employ the electronic charts as models to be matched by the radar measurements as suggested by A. Guesalaga.[19]

Acknowledgments

This project has been supported by the National Commission for Science and Technology Research of Chile (Conicyt) under Fondecyt Grant 1050666.

References

1. S. Thrun, W. Burgard and D. Fox, *Probabilistic Robotics* (MIT Press, Cambridge, MA, 2005).
2. G. D. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics* (Cambridge University Press, Cambridge, UK, 2000).
3. J. Borenstein, H. Everett, L. Feng and D. Wehe, "Mobile robot positioning: Sensors and techniques," *J. Robot. Syst.* **14**, 231–249 (1997).
4. S. Iyengar and A. Elfes, *Autonomous Mobile Robots* (IEEE Computer Society, Los Alamitos, CA, 1991).
5. A. Elfes, "Sonar-based real-world mapping and navigation," *IEEE Trans. Robot. Autom.* **3**, 249–265 (1987).
6. J. J. Leonard and H. F. Durrant-Whyte, *Directed Sonar Sensing for Mobile Robot Navigation* (Kluwer Academic, Norwell, MA, 1992).
7. J. D. Tardós, J. Neira, P. Newman and J. Leonard, "Robust mapping and localization in indoor environments using sonar data," *Int. J. Robot. Res.* **21**(4), 311–330 (2002).
8. I. J. Cox, "Blanche: An experiment in guidance and navigation of an autonomous robot vehicle," *IEEE Trans. Robot. Autom.* **7** (2), 193–203 (1991).
9. G. Weiss and E. Puttkamer, "A Map Based on Laserscans Without Geometric Interpretation," *Proceedings of the Intelligent Autonomous Systems.* (1995) pp. 403–407.
10. P. Jensfelt and H. Christensen, "Laser Based Position Acquisition and Tracking in an Indoor Environment," *Proceedings of the IEEE International Symposium on Robotics and Automation* (1998) pp. 331–338.
11. G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.* **24**(2), 237–267 (2002).
12. A. Diosi and L. Kleeman, "Advanced Sonar and Laser Range Finder Fusion for Simultaneous Localization and Mapping," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2004) pp. 1854–1859.
13. J. A. Castellanos, J. Neira and J. D. Tardós, "Multisensor fusion for simultaneous localization and map building," *IEEE Trans. Robot. Autom.* **17**(6), 908–914 (2001).
14. J. A. Castellanos and J. D. Tardós, *Mobile Robot Localization and Map Building: A Multisensor fusion Approach* (Kluwer Academic, Norwell, MA, 1999).
15. R. Chatila and J.-P. Laumond, "Position Referencing and Consistent World Modeling for Mobile Robots," *Proceedings of the IEEE International Conference on Robotics and Automation* (1985) pp. 138–145.
16. J. L. Crowley, F. Wallner and B. Schiele, "Position Estimation Using Principal Components of Range Data," *Proceedings of the IEEE International Conference on Robotics and Automation* (1998) pp. 3128–3131.
17. W. J. Rucklidge, "Efficient visual recognition using the Hausdorff distance," *Int. J. Comput. Vis.* **24**, 251–270 (1997).
18. D.-G. Sim and R.-H. Park, "Two-dimensional object alignment based on robust oriented Hausdorff similarity measure," *IEEE Trans. Image Processing* **10**, 475–483 (2001).
19. A. Guesalaga, "Recursive estimation of radar biases using electronic charts," *IEEE Trans. Aerospace Electro. Syst.* **40**(2), 725–733 (2004).
20. J. Gutmann, W. Burgard, D. Fox and K. Konolige, "An Experimental Comparison of Localization Methods," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (1998) pp. 736–743.
21. D. Filliat and J.-A. Meyer, "Map-based navigation in mobile robots. I. A review of localization strategies," *Cogn. Syst. Res.* 243–282 (2003).
22. C. F. Olson, "Probabilistic self-localization for mobile robots," *IEEE Trans. Robot. Autom.* **16**(1), 55–66 (2000).
23. R. Talluri and J. Aggarwal, "Positional Estimation Techniques for An Autonomous Mobile Robot — A Review," **In:** *Handbook of Pattern Recognition and Computer Vision* (C. H. Chen, L. F. Pau and P. S. P. Wang, eds.) (World Scientific Singapore, 1993) Ch. 4.4, pp. 769–801.
24. B. Schiele and J. L. Crowley, "A comparison of position estimation techniques using occupancy grids," *Robot. Autonom. Syst.* **12**, 163–171 (1994).
25. J.-S. Gutmann and C. Schlegel, "Amos: Comparison of Scan Matching Approaches for Self-localization in Indoor Environments," *Proceedings of the 1st Euromicro Workshop on Advanced Mobile Robots* (1996) pp. 61–67.
26. L. Iocchi, D. Mastrantuono and D. Nardi, "A Probabilistic Approach to Hough Localization," *Proceedings of the IEEE International Conference on Robotics and Automation* (2001) pp. 4250–4255.
27. I. Nourbakhsh, R. Powers, and S. Birchfield, "DERVISH an office-navigating robot," *AI Mag.* **16**(2), 53–60 (1995).
28. R. Simmons and S. Koenig, "Probabilistic Robot Navigation in Partially Observable Environments," *Proceedings of the International Joint Conference on Artificial Intelligence* (1995).
29. L. Kaelbling, A. Cassandra and J. Kurien, "Acting Under Uncertainty: Discrete Bayesian Models for Mobile-Robot Navigation," *Proceedings of the IEEE/RSJ International Conference Intelligent Robots and Systems* (1996) pp. 963–972.
30. S. Thrun, "Bayesian landmark learning for mobile robot localization," *Mach. Learning* **33**(1), 41–76 (1998).
31. N. Roy and S. Thrun, "Coastal navigation with mobile robots," *Adv. Neural Processing Syst.* **12**, 1043–1049 (1999).

32. W. Burgard, D. Fox, D. Hennig and T. Schmidt, "Estimating the Absolute Position of a Mobile Robot Using Position Probability Grids," *Proceedings of the 14th National Conference on Artificial Intelligence* (1996) pp. 896–901.
33. W. Burgard, A. Derr, D. Fox and A. B. Cremers, "Integrating Global Position Estimation and Position Tracking for Mobile Robots: The Dynamic Markov Localization Approach," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (1998) pp. 730–735.
34. D. Fox, W. Burgard and S. Thrun, "Markov localization for mobile robots in dynamic environments," *J. Artif. Intell. Res.* **11**, 391–427 (1999).
35. F. Dellaert, D. Fox, W. Burgard and S. Thrun, "Monte Carlo Localization for Mobile Robots," *Proceedings of the IEEE International Conference on Robotics and Automation* (1999) pp. 1322–1328.
36. D. Fox, S. Thrun, F. Dellaert and W. Burgard, "Particle Filters for Mobile Robot Localization," **In:** *Sequential Monte Carlo in Practice* (A. Doucet, N. de Freitas and N. Gordon, eds.) (Springer-Verlag, Berlin, Germany, 2001).
37. F. Dellaert, "A Sample of Monte Carlo Methods in Robotics and Vision," *Proceedings of the ISM International Symposium on the Science of Modeling — The 30th Anniversary of the Information Criterion* (2003) pp. 211–222.
38. N. J. Gordon, D. J. Salmond and A. F. M. Smith, "Novel approach to nonlinear/Non-Gaussian Bayesian state estimation," *IEE Proc. F* **140** (2), 107–113 (1993).
39. G. Kitagawa, "Monte Carlo filter and smoother for non-Gaussian nonlinear state space models," *J. Comput. Graphical Statist.* **5**(1), 1–25 (1996).
40. M. Isard and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," *Proceedings of European Conference on Computer Vision* (1996) pp. 343–356.
41. M. Isard and A. Blake, "Condensation–conditional density propagation for visual tracking," *Int. J. Comput. Vis.* **29**(1), 5–28 (1998).
42. B. Ristic, S. Arulampalam and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications* (Artech House Norwood, MA, 2004).
43. H. Moravec and A. Elfes, "High Resolution Maps from Wide Angle Sonar," *Proceedings of the IEEE International Conference on Robotics and Automation* (1985) pp. 116–121.
44. P. Jensfelt and S. Kristensen, "Active global localisation for a mobile robot using multiple hypothesis tracking," *IEEE Trans. Robot. Autom.* **17**(5), 748–760 (2001).
45. J.-S. Gutmann and D. Fox, "An Experimental Comparison of Localization Methods Continued," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2002) pp. 454–459.
46. S. Kristensen and P. Jensfelt, "An Experimental Comparison of Localisation Methods, the MHL Sessions," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2003) pp. 992–997.
47. G. Schaffer, J. Gonzalez and A. Stentz, "Comparison of Two Range-Based Pose Estimators for a Mobile Robot," *Proceedings of the 1992 SPIE Conference on Mobile Robots* (1992) pp. 661–667.
48. L. Marchetti, G. Grisetti and L. Iocchi, "A Comparative Analysis of Particle Filter Based Localization Methods," *Proc. of RoboCup Symposium* (2006).
49. R. C. Veltkamp and M. Hagerdoorn, "State of the Art in Shape Matching," **In:** *Principles of Visual Information Retrieval* (*Advances in Pattern Recognition*) (M. S. Lew ed.) (Springer, Berlin, Germany, 2006) pp. 87–115.
50. F. Lu and E E Milios, "Robot Pose Estimation in Unknown Environments by Matching 2D Range Scans," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1994) pp. 935–938.
51. F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonom. Robots* **4**, 333–349 (1997).
52. T. Bailey and E. Nebot, "Localisation in large-scale environments," *Robot. Autonom. Syst.* **37**(4), 261–281 (2001).
53. S. T. Pfister, K. L. Kreichbaum, S. I. Roumeliotis and J.W. Burdick, "Weighted range Sensor Matching Algorithms for Mobile Robot Displacement Estimation," *Proceedings of the IEEE International Conference on Robotics and Automation* (2002) pp. 1667–1674.
54. J. Minguez, F. Lamiraux and L. Montesano, "Metric-Based Scan Matching Algorithms for Mobile Robot Displacement Estimation," *Proceedings of the IEEE International Conference on Robotics and Automation* (2005) pp. 3557–3563.
55. P. J. Besl and N. D. McKay, "A method for registration of 3-D shapes," *IEEE Trans. Pattern. Anal. Mach. Intell.* **14**(2), 239–256 (1992).
56. Z. Zhang, "Iterative point matching for registration of free-form curves and surfaces," *Int. J. Comput. Vis.* **13**(2), 119–152 (1994).
57. S. Rusinkiewicz and M. Levoy, "Efficient Variants of the ICP Algorithm," *Proceedings of the International Conference on 3D Digital Imaging and Modeling* (2001) pp. 145–152.
58. Y. Liu, "Improving ICP with easy implementation for free form surface matching," *Pattern Recog.* **37**, 211–226 (2004).
59. A. Diosi and L. Kleeman, "Laser Scan Matching in Polar Coordinates with Application to SLAM," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2005) pp. 3317–3322.
60. G. Weiss, C. Wetzler and E. Puttkamer, "Keeping Track of Position and Orientation of Moving Indoor Systems by Correlation of Range-Finder Scans," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (1994) pp. 595–601.
61. A. C. Schultz and W. Adams, "Continuous localization using evidence grids," *Proceedings of the IEEE International Conference on Robotics and Automation* (1998) pp. 2833–2839.
62. K. Konolige and K. Chou, "Markov localization using correlation," *Proceedings of the International Joint Conference on Articial Intelligence* (1999) pp. 1154–1159.
63. L. Zhang and B. K. Ghosh, "Line Segment Based Map Building and Localization Using 2D Laser Rangefinder," *Proceedings of the IEEE International Conference on Robotics and Automation* (2000) pp. 2538–2543.
64. G. A. Borges and M.-J. Aldon, "Line extraction in 2D range images for mobile robotics," *J. Intell. Robot. Syst.* **40**, 267–297 (2004).
65. V. Nguyen, A. Martinelli, N. Tomatis and R. Siegwart, "A Comparison of Line Extraction Algorithms Using 2D Laser Rangefinder for Indoor Mobile Robotics," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* (2005) pp. 1929–1934.
66. D. P. Bertsekas, *Nonlinear Programming* 2nd ed. (Athena Scientific, Belmont, MA, 2004).
67. D. P. Huttenlocher, G. A. Klanderman and W. J. Rucklidge, "Comparing images using the Hausdorff distance," *IEEE Trans. Pattern Anal. Mach. Intell.* **15**(9), (1993) 850–863.
68. O. Kwon, D.-G. Sim and R.-H. Park. "Nonparametric hierarchical Hausdorff distance matching algorithm," *Opt. Eng.* **39**, 1917–1927, (2000).
69. D. W. Taylor, P. N. Johnson and W. T. Faulkner, "Local Area Radio Navigation: A Tool for GPS-denied Geolocation," *Proceedings of the SPIE-Aerosense Conf.* (2003) Vol. 5084, pp. 125–136.

https://doi.org/10.1017/S0263574707003657 Published online by Cambridge University Press