Published in IET Computer Vision
Received on 19th October 2011
Revised on 2nd October 2012
Accepted on 22nd December 2012
doi: 10.1049/iet-cvi.2011.0214

ISSN 1751-9632

Vertical edge-based mapping using range-augmented omnidirectional vision sensor

Bladimir Bacca[1,2], Xavier Cufí[1], Joaquim Salví[1]

[1]*Institut d'Informatica I Aplicacions, Universitat de Girona, Building PIV, Girona, Spain*
[2]*Escuela de Ing. Eléctrica y Electrónica, Universidad del Valle, Melendez, Cali, Colombia*
E-mail: bladimir@eia.udg.edu

Abstract: Laser range finder and omnidirectional cameras are becoming a promising combination of sensors to extract rich environmental information. This information includes textured plane extraction, vanishing points, catadioptric projection of vertical and horizontal lines, or invariant image features. However, many indoor scenes do not have enough texture information to describe the environment. In these situations, vertical edges could be used instead. This study presents a sensor model that is able to extract three-dimensional position of vertical edges from a range-augmented omnidirectional vision sensor. Using the unified spherical model for central catadioptric sensors and the proposed sensor model, the vertical edges are locally projected, improving the data association for mapping and localisation. The proposed sensor model was tested using the FastSLAM algorithm to solve the simultaneous localisation and mapping problem in indoor environments. Real-world qualitative and quantitative experiments are presented to validate the proposed approach using a Pioneer-3DX mobile robot equipped with a URG-04LX laser range finder and an omnidirectional camera with parabolic mirror.

1 Introduction

Reliable self-localisation and mapping plays an important role in autonomous robot navigation. Indoor environments are the scenario where service robots move for surveillance, inspection, delivery and cleaning tasks. The perception system of a mobile robot must provide accurate information of the robot environment, taking advantage of its surroundings in order to reconstruct a consistent representation of the environment. In many applications, this environment representation is not known and the mobile robot has to locate itself while it is doing a mapping using a set of sequential observations. This problem is called simultaneous localisation and mapping (SLAM), which has been studied throughout many years [1, 2].

The most relevant solutions to SLAM are focused on the feature-based approach, where feature descriptors are extracted from laser scans or images to solve the problem of matching observations to landmarks. Other alternative approaches are as follows: In [3] two-dimensional (2D) raw range data is used to extract saliency using the iterative closest point (ICP) [4] algorithm regardless of the geometric representation of the environment; [5] computes the camera pose and the scene structure considering illumination changes using the entire image intensities to extract saliency regions, and then perform a non-linear minimisation. Despite the fact that the authors in [5] consider the illumination changes, this hypothesis is valid only locally.

In recent years, appearance-based mapping and localisation has gained special attention, since these methods use a richer description of the environment giving more cues to improve robot mapping and localisation [6, 7]. These approaches are featured-based (e.g. SIFT, SURF, Harris corners etc.) and they present a probabilistic framework to build appearance-based topological maps, assuming environments with enough texture to extract the image features. In [7], the authors include the laser scans in their probabilistic framework in order to extract the environment topology in the presence of poor textured environments. Appearance-based methods also exploit the environment structure given mainly by vertical and horizontal edges, such as doors, planes and so on, to obtain a better representation of the robot surroundings. In this sense, Wongphati *et al.* [8] proposed a solution using omnidirectional images and bearing information of vertical edges to solve SLAM, but this approach needs two consecutive frames to extract the position of the vertical edges, delaying the robot pose estimation. The authors in [9] proposed a vertical edge descriptor that robustly matches catadioptric vertical edges. The descriptor proposed in [9] is valid locally, requiring two or more views to extract the scene vertical edges position.

Besides, common available laser range finders (LRFs) work in a plane parallel to the ground, and then it limits the environment representation to 2D. Combining vision sensors with LRF increases the perceptual information, but monocular or stereo cameras have limited field of view, affecting their perception because of occlusions and feature lifetime observation. Omnidirectional cameras have received special attention recently because of its long-term landmark tracking, its wide field of view, its robustness to occlusions

and its ability to be fused with range data. Therefore combining an omnidirectional camera with a LRF has many advantages: all the laser trace can be used to extract environment features on the image plane, depth information can be embedded into the omnidirectional image, 3D feature information can be recovered, and once the calibration between these sensors is performed, for first time, it can be used in real time. A seminal work in this sense was proposed by Biber *et al.* [10], where the SLAM problem is solved with the LRF and the appearance of the environment is captured using the omnidirectional camera. In [11], a system based on a LRF and an omnidirectional camera is described to obtain a map of the environment using scan matching and vertical edges; however, the authors neither solve the SLAM problem nor define a data association method. In [12], a hybrid sensor is presented and is composed of an omnidirectional camera and a LRF, where the laser trace is projected onto the omnidirectional image and it is used to extract salient features on the image plane using 1D intensity signals around the detected vertical edge as a local data association method; however, pixel intensity-based methods need short base-line movements to be used with custom similarity metrics and do not produce discriminative enough features for data association.

In this paper, we present a sensor model based on the extrinsic calibration between a LRF and an omnidirectional camera [13], in order to robustly extract the 3D position of vertical edges in indoor environments and use them to solve the SLAM problem. Data association is very important for probabilistic frameworks that deal with mapping and localisation. Our approach uses a two-step algorithm to solve data association: First, a joint compatibility branch and bound (JCBB) test [14] is performed; second, a geometric constraint based on the catadioptric projection of the scene vertical edges and a rigid transformation is considered in order to resolve the ambiguous associations obtained from the JCBB test. The experimental validation was performed by integrating our sensor model into the FastSLAM [15, 16] framework and using a data set collected in the indoors of the University of Girona.

This paper is organised as follows: Section 2 describes the extrinsic calibration between the LRF and the omnidirectional camera. Section 3 concerns the description of the sensor model, vertical edge position estimation and data association. Section 4 describes the scenario and the experimental results. The paper ends with conclusions.

2 LRF/omnidirectional camera calibration

A central catadioptric camera consists of a perspective or orthographic camera and a mirror. The latter can be conic, hyperbolic or parabolic. Projective models for these cameras have been developed by [17, 18]. In the remainder of this paper, we adopt the model described in [18], because it is related to the toolbox described in [19], and it was used to calibrate our central catadioptric camera.

The URG-04LX 2D LRF used in this work was previously calibrated in order to decrease the range error following the procedure described in [13]. The raw 2D laser scan data are previously processed using a median filter to discard spurious readings. Our approach was tested using a data set with the robot in motion; for this reason motion compensation is performed in the 2D laser scan data. For self-containment, the following are the main steps performed to find the extrinsic calibration parameters between the omnidirectional camera and the LRF.

Fig. 1 depicts the extrinsic calibration between the omnidirectional camera and the LRF. There are three coordinate systems: those of the laser, the calibration pattern and the camera. The problem focuses on finding $\boldsymbol{R}$ and $\boldsymbol{T}$, the rotation and translation matrices of the camera with respect to the LRF, so that laser points $\boldsymbol{P}_{\mathrm{L}}$ can be represented in the camera coordinate system, and then projected onto the omnidirectional image. According to [13], the laser points ($\boldsymbol{P}_{\mathrm{L}}$) that belong to the calibration plane satisfy a geometric constraint based on the distance between the camera and the calibration plane. This constraint can be expressed by (1)

$$\boldsymbol{N}_{\mathrm{C}}\left(\boldsymbol{R}\boldsymbol{P}_{\mathrm{L}}+\boldsymbol{T}\right)=\boldsymbol{N}_{\mathrm{C}}^{2} \tag{1}$$

where $\boldsymbol{N}_{\mathrm{C}}=\boldsymbol{R}_{3,\mathrm{C}}\times\left(\boldsymbol{R}_{3,\mathrm{C}}^{\mathrm{T}}\,\boldsymbol{T}_{\mathrm{C}}\right)$ is a vector parallel to the normal vector of the calibration plane extracted from the omnidirectional camera calibration; $\boldsymbol{R}_{3,\mathrm{C}}$ and $\boldsymbol{T}_{\mathrm{C}}$ correspond to the third column vector of the calibration plane extrinsic rotation matrix and its translational vector, respectively; $\boldsymbol{R}$ and $\boldsymbol{T}$ are the parameters to estimate; and $\boldsymbol{P}_{\mathrm{L}}$ is a laser point. Using (1), a non-linear minimisation function that simultaneously estimates $\boldsymbol{R}$ and $\boldsymbol{T}$ can be expressed as in (2) and (3).

$$f\left(\boldsymbol{R},\ \boldsymbol{T},\ \boldsymbol{P}_{\mathrm{L}}\right)=\boldsymbol{N}_{\mathrm{C}}\left(\boldsymbol{R}\boldsymbol{P}_{\mathrm{L}}+\boldsymbol{T}\right)-\boldsymbol{N}_{\mathrm{C}}^{2} \tag{2}$$

$$\nabla_{Q,T}f\left(\boldsymbol{R},\ \boldsymbol{T},\ \boldsymbol{P}_{\mathrm{L}}\right)=\boldsymbol{N}_{\mathrm{C}}\left[\nabla\boldsymbol{R}_{Q,P_{\mathrm{L}}}\nabla n_{Q},\ \ I_{3x3}\right] \tag{3}$$

where $\nabla_{\mathrm{Q,T}}\boldsymbol{f}\left(\boldsymbol{R},\ \boldsymbol{T},\ \boldsymbol{P}_{\mathrm{L}}\right)\boldsymbol{P}_{\mathrm{L}}$ is the gradient of the minimisation function using quaternions, $\nabla\boldsymbol{R}_{Q,P_{\mathrm{L}}}$ is the quaternion's gradient evaluated at point $\boldsymbol{P}_{\mathrm{L}}$, and $\nabla\boldsymbol{n}_{\boldsymbol{Q}}$ is the gradient of the quaternion's normalisation factor. A Levenberg–Marquardt algorithm was used to minimise (3), which need an initial guess. In [13], three different ways to obtain it were presented; however, the best results were obtained using the linear least squares version of (2) applied to two different calibration planes. Afterwards, the initial guess gave a rank-2 rotation matrix, since the resulting matrix is not a proper rotation matrix, because it does not satisfy $\boldsymbol{R}\boldsymbol{R}^{\mathrm{T}}=\boldsymbol{I}$. This happens because the laser points are

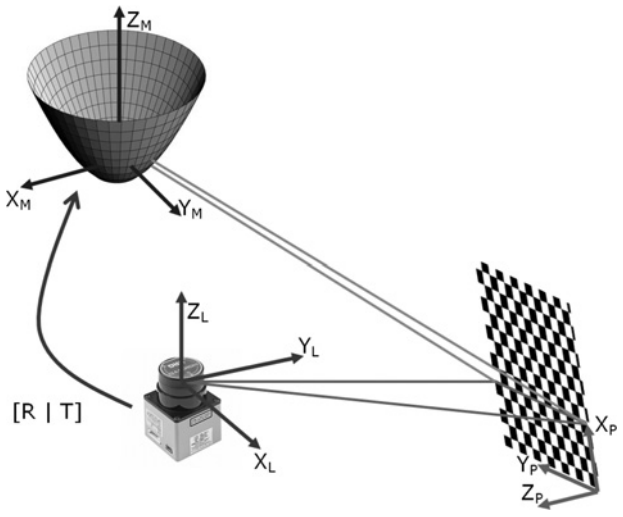

Fig. 1 *Omnidirectional camera and LRF extrinsic calibration*

doi: 10.1049/iet-cvi.2011.0214

constrained on a plane. For this reason, the nearest rotation matrix was found as proposed in [20]. At the end of the non-linear minimisation process, and using 14 calibration planes, the following results were obtained: $\boldsymbol{T} = [-0.0069 \pm 0.0007,\ -0.2100 \pm 0.0017,\ 0.5557 \pm 0.027]^{\mathrm{T}}$ in meters and $0.7644° \pm 0.002°$, $-4.5928° \pm 0.0005°$ and $-105.0235° \pm 0.033°$ for the Roll, Pitch and Yaw angles of $\boldsymbol{R}$.

3 Sensor model

Once the extrinsic parameters of the LRF and the omnidirectional camera were obtained, the laser points can be projected onto the omnidirectional image using the projection model described in [19]. The sensor model proposed uses the laser points and the vertical edges on the omnidirectional image to obtain a set of observations in terms of range, azimuth and elevation of the scene vertical edges. Section 3.1 presents the vertical edge extraction; the observation model is presented in Section 3.2 and the data association method in Section 3.3.

3.1 Vertical edge detection

For edge detection, we used a modified version of [21]. Popular solutions consider the Hough transform; however, these solutions require a sampling of the search space and the accuracy depends on it. On the other hand, the method proposed in [21] allows computing the edges analytically, which is needed for our proposal as it will be further explained in Section 3.2. For self-containment, the main idea of the method proposed in [21] is described. This algorithm starts detecting edges in the image (e.g. using Canny or Sobel detectors) and building chains of connected edge pixels. Then these chains are projected onto the sphere, where the great circle constraint is verified to be considered as an edge. Otherwise, it is iteratively cut into two sub-chains until a sub-chain is considered as an edge or its length is too small. Afterwards, a merging step is applied because an edge might be decomposed into more than a single chain. The approach presented in [21] detects any edge; hence we modified the pipeline process in order to detect only the vertical edges. First, instead of validating an edge using the normal vector extracted from the extreme points, which is inaccurate since it does not keep in mind the other edge points, our implementation of the algorithm considers an improved geometric constraint shown in (4). It comes from the relationship between the normal vector of a plane and the points belonging to this plane.

$$\begin{bmatrix} x_1 & y_1 & z_1 \\ \dots & \dots & \dots \\ x_n & y_n & z_n \end{bmatrix} \boldsymbol{N}_{\mathrm{S}} = \boldsymbol{A} \times \boldsymbol{N}_{\mathrm{S}} = 0 \tag{4}$$

where $\boldsymbol{P}_{\mathrm{S}} = [x_n\ y_n\ z_n]$ is a point on the sphere, n is the total number of points belonging to a chain, and $\boldsymbol{N}_{\mathrm{S}}$ is the normal vector. Using a singular value decomposition of $\boldsymbol{A} = \boldsymbol{USV}^{\mathrm{T}}$, ordering the eigenvalues of $\boldsymbol{S}$ in decreasing order, the third column of $\boldsymbol{V}$ contains the least-squared solution of (4). Secondly, once the edge was validated, an additional constraint given by the cross-product between the normal vector of the edge $\boldsymbol{N}_{\mathrm{S}}$ and the normal ground plane $\boldsymbol{z} = [0\ 0\ 1]^{\mathrm{T}}$ was added. This allows detecting only the vertical edges, since their dot product is close to zero. Last, to improve the computing time the edge and chain detection were implemented in MEX files; and using the camera calibration a look-up table was computed to accelerate the image point projection onto the sphere.

3.2 Vertical edge position estimation

The vertical edge position computation with respect to the robot frame takes place as follows: First, the LRF data are divided into line segments using a Kalman-based breakpoint detector and a split-and-merge approach; afterwards, using the omnidirectional camera projection model, the extrinsic calibration described in Section 2, the line projection model [22] and the intrinsic camera parameters, the laser segments are projected as conics on the image plane. Despite the fact that the LRF trace is compensated for the robot motion, the LRF trace does not exactly coincide with the scene vertical edges as depicted in Fig. 2*a*. For this reason, a robust data association is implemented and described in the following stages. Secondly, using the projection model the breakpoint uncertainties are projected onto the image plane as depicted in (5)–(8)

$$\Sigma_{Ci}^{I} = \boldsymbol{J}_{\mathrm{K}}\boldsymbol{J}_{\mathrm{D}}\boldsymbol{J}_{\mathrm{S}}\boldsymbol{J}_{\mathrm{R}}\boldsymbol{J}_{\mathrm{P}} \begin{bmatrix} \sigma_{\rho}^2 & 0 & 0 \\ 0 & \rho\sigma_{\theta_{Ci}}^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{J}_{\mathrm{P}}^{\mathrm{T}}\boldsymbol{J}_{\mathrm{R}}^{\mathrm{T}}\boldsymbol{J}_{\mathrm{S}}^{\mathrm{T}}\boldsymbol{J}_{\mathrm{D}}^{\mathrm{T}}\boldsymbol{J}_{\mathrm{K}}^{\mathrm{T}} \tag{5}$$

$$\boldsymbol{J}_{\mathrm{P}} = \begin{bmatrix} \cos(\theta_{Ci}) & -\sin(\theta_{Ci}) & 0 \\ \sin(\theta_{Ci}) & \cos(\theta_{Ci}) & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{6}$$

$$\boldsymbol{J}_{\mathrm{S}} = \frac{1}{r_p(z_s + \xi r_p)^2} \times \begin{bmatrix} r_p z_s + \xi(y_s^2 + z_s^2) & -\xi x_s y_s & -x_s(r_p + \xi z_s) \\ -\xi x_s y_s & r_p z_s + \xi(x_s^2 + z_s^2) & -y_s(r_p + \xi z_s) \end{bmatrix} \tag{7}$$

$$\boldsymbol{J}_{\mathrm{K}} = \begin{bmatrix} \gamma_1 & s\gamma_1 \\ 0 & \gamma_2 \end{bmatrix} \tag{8}$$

where σ_r^2 and $\sigma_{\theta c}^2$ are the variances of the range and orientation of the LRF, respectively, ρ is the range of the ith breakpoint in the LRF frame, $\boldsymbol{J}_{\mathrm{P}}$ is the Jacobian of the polar to Cartesian coordinate transformation, θ_{Ci} is the breakpoint

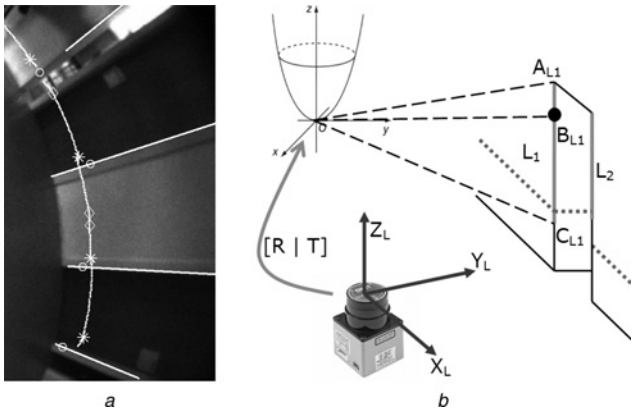

Fig. 2 *Vertical edge position estimation*

a Vertical edges and their LRF corner associations
b Vertical edge position measurement

orientation in the LRF frame; $\boldsymbol{J}_{\mathrm{R}}$ is the Jacobian of the standard Roll–Pitch–Yaw rotation matrix obtained in Section 2 (not shown for reasons of space); $\boldsymbol{J}_{\mathrm{S}}$ is the Jacobian of the projection function to the image plane at infinity, ζ is the mirror parameter, $[x_{\mathrm{S}}\ y_{\mathrm{S}}\ z_{\mathrm{S}}]^{\mathrm{T}}$ is the point on the sphere, and r_{P} its magnitude; $\boldsymbol{J}_{\mathrm{D}}$ is the Jacobian of the distortion function which was taken from [19]; $\boldsymbol{J}_{\mathrm{K}}$ is the Jacobian of the camera projection function, and γ_1, γ_2 and s are the intrinsic camera parameters.

Thirdly, using the conic projections on the image plane [22] corresponding to the LRF segments, the intersecting points with the vertical edge are computed solving the second-order equation given by $\boldsymbol{P}_{\mathrm{L}}^{\mathrm{T}}\boldsymbol{C}_{\mathrm{Im}}\boldsymbol{P}_{\mathrm{L}} = 0$, where $\boldsymbol{P}_{\mathrm{L}}$ is the intersection point that resides on the conic defined by $\boldsymbol{C}_{\mathrm{Im}}$. By doing this, two intersects are yielded for each vertical edge. Fourthly, the data association between the projected breakpoints and the vertical edge intersects is performed using the JCBB test. Unlike [11], where no details were described with respect to the vertical edge position estimation, the method presented here robustly associates each vertical edge to a range/bearing pair in the camera frame. Fig. 2*a* shows an example of the resulting data association described above. In this figure, the conic intersects are shown in circle-shaped points and the associated LRF breakpoints are shown in star-shaped points. The remaining LRF breakpoints are also shown in diamond-shaped points. Lastly, considering Fig. 2*b*, where B_{L1} corresponds to the XY position of the vertical edge with respect to the camera frame, the 3D ending point A_{L1} and the 3D initial point C_{L1} are computed using the associated range and bearing information and the projection of the vertical edge onto the sphere. These points can be expressed as depicted in (9) and (10).

$$\boldsymbol{A}_{\mathrm{L1}} = \begin{bmatrix} x_A \\ y_A \\ z_A \end{bmatrix} = \begin{bmatrix} r_{Ci}\cos(\theta_{Ci}) \\ \rho_{Ci}\sin(\theta_{Ci}) \\ \dfrac{\rho_{Ci}}{\tan(\pi - \theta_{A1})} \end{bmatrix} \tag{9}$$

$$\boldsymbol{C}_{\mathrm{L1}} = \begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} = \begin{bmatrix} r_{Ci}\cos(\theta_{Ci}) \\ \rho_{Ci}\sin(\theta_{Ci}) \\ \dfrac{\rho_{Ci}}{\tan(\pi - \theta_{C1})} \end{bmatrix} \tag{10}$$

where r_{Ci} and θ_{Ci} are the associated range and bearing of the vertical edge with respect the camera, respectively, and θ_{A1} and θ_{C1} are the angles of the ending and initial points with respect to the Z-axis of the camera frame. Using these scene points and the inverse transformation of the extrinsic calibration parameters (Section 2), the observation model with respect to the LRF frame can be computed as depicted in (11).

$$Z_n = \begin{bmatrix} r_n \\ \theta_n \\ \psi_n \end{bmatrix} = \begin{bmatrix} \sqrt{(m_x - x_R)^2 + (m_y - y_R)^2 + m_z^2} \\ \operatorname{atan2}\left(\dfrac{m_y - y_R}{m_x - x_R}\right) - \theta_R \\ \cos^{-1}\left(\dfrac{m_z}{r_n}\right) \end{bmatrix} \tag{11}$$

where r_n, θ_n and ψ_n are the observed range, azimuth and elevation of the middle point $\boldsymbol{m}_{\mathrm{L1}} = [m_x,\ m_y,\ m_z]$ of edge L_1 given the current robot position $\boldsymbol{X}_t = [x_R,\ y_R,\ \theta_R]$, which is assumed the same frame as the LRF. In Section 4.1, the linearisation of the measurement model with respect to the feature coordinates and the robot state will be needed. The Taylor approximation of this measurement model is depicted in (12).

$$z_t\left(X_t, m_{H_t}\right) \simeq z_t\left(X_t, \overline{m_{H_t,t-1}}\right) + \mathbf{J}_t\left(m_{H_t} - \overline{m_{H_t,t-1}}\right) \tag{12}$$

$$\mathbf{J}_t = \begin{bmatrix} \dfrac{\mathrm{d}x}{r_n} & \dfrac{\mathrm{d}y}{r_n} & \dfrac{\mathrm{d}z}{r_n} \\ \dfrac{-\mathrm{d}y}{d} & \dfrac{\mathrm{d}x}{d} & 0 \\ \dfrac{\mathrm{d}x}{\mathrm{d}f} & \dfrac{\mathrm{d}y}{\mathrm{d}f} & \dfrac{-1}{\sqrt{d - \mathrm{d}z^2}} \end{bmatrix} \tag{13}$$

$$z_t\left(X_t, m_{H_t}\right) \simeq z_t\left(\overline{X_t}, m_{H_t}\right) + \mathbf{F}_t\left(X_t - \overline{X_t}\right) \tag{14}$$

$$\mathbf{F}_t^k = \begin{bmatrix} \dfrac{-\mathrm{d}x}{r_n} & \dfrac{-\mathrm{d}y}{r_n} & 0 \\ \dfrac{\mathrm{d}y}{d} & \dfrac{-\mathrm{d}x}{d} & -1 \\ \dfrac{-\mathrm{d}x}{\mathrm{d}f} & \dfrac{-\mathrm{d}y}{\mathrm{d}f} & 0 \end{bmatrix} \tag{15}$$

where $\boldsymbol{J}_t$ is the Jacobian with respect to the feature coordinates, $\boldsymbol{X}_t$ is the robot state, $\overline{\boldsymbol{m}_{H_t,t-1}}$ is the feature mean location in the previous time step, $\mathrm{d}x = m_x - x_R$, $\mathrm{d}y = m_y - y_R$, $\mathrm{d}z = m_z$, $d = (m_x - x_R)^2 + (m_y - y_R)^2 + m_z{}^2$ and $f = \sqrt{(d/\mathrm{d}z^2) - 1}$, $\boldsymbol{F}_t$ is the Jacobian with respect to the robot state ($\boldsymbol{X}_t$), and $\overline{\boldsymbol{X}_t}$ is the predicted robot state.

3.3 Data association

In our implementation, we considered the JCBB test to solve the data association problem [14], such that the correlations between innovations are explicitly taken into account to determine the joint compatibility of a set of pairings. The sensor model proposed in this paper is focused on finding the range, azimuth and elevation of the vertical edge middle point. However, JCBB associate features depending on the innovations, their covariance and the innovation gate distance metric used (Mahalanobis distance), which are no longer valid when the uncertainty becomes important. Therefore in this paper, we propose a method for distinguishing vertical edge features which are close to each other when the measure of uncertainty is not sufficient to obtain a better data association.

First, the JCBB test is used. If a current vertical edge feature is associated with two or more landmarks in the map, the second step is performed. Fig. 3 shows a typical situation, where two views are involved: the current view with one observation associated with two map landmarks. This view is placed at the predicted robot pose $\boldsymbol{O}_{\mathrm{C}}$, where a scene vertical edge ${}^{Z}L_n$ is observed and projected onto the sphere as ${}^{Z}L_{ns}$ using the unified projection model; a plane is defined between ${}^{Z}L_{ns}$ and $\boldsymbol{O}_{\mathrm{C}}$, which can be parameterised by the normal vector $\boldsymbol{N}_{ns}$, since for all points ${}^{Z}L_k$ belonging to ${}^{Z}L_{ns}$ the dot product property $({}^{Z}L_k{}^{\mathrm{T}})\cdot(N_{ns}) = 0$ is satisfied. On the other hand, two landmarks candidates (i.e. ${}^{m}L_i$ and ${}^{m}L_j$) corresponding to the map view placed at $\boldsymbol{O}_m$ have their analogous spherical projections (i.e. ${}^{m}L_{is}$ and ${}^{m}L_{js}$) and planes parameterised by the vector normal ${}^{m}\boldsymbol{N}_{is}$ and ${}^{m}\boldsymbol{N}_{js}$, respectively. In our implementation, the robot pose in which a feature was seen for the first time is saved; this allows us to relate $\boldsymbol{O}_{\mathrm{C}}$ and $\boldsymbol{O}_m$ through a transformation defined by $\boldsymbol{R}$ and $\boldsymbol{T}$.

doi: 10.1049/iet-cvi.2011.0214

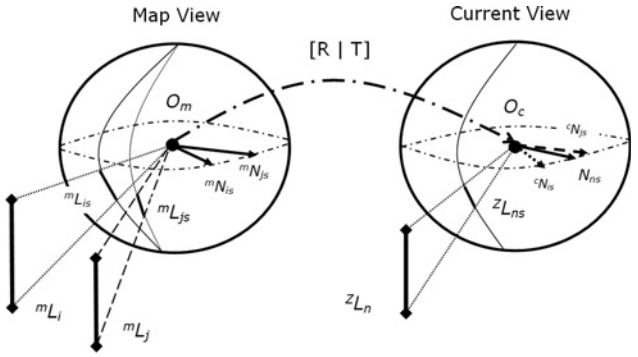

Fig. 3 *Resolving ambiguous data association*

The intuitive idea behind our method is – given the current observation parameterised by the vector normal $\boldsymbol{N}_{ns}$ and the map landmarks candidates, in this case parameterised by ${}^m\boldsymbol{N}_{is}$ and ${}^m\boldsymbol{N}_{js}$ in the map view – associate the observation ${}^Z\boldsymbol{L}_{ns}$ with the closer landmark map parameterised by the transformed vector normal ${}^C\boldsymbol{N}_{is}$ and ${}^C\boldsymbol{N}_{js}$ in the current view. This is formally described by (16).

$$c_i = \min_i \sum_{k=1}^{FZ} \mathbf{L}_k^{\mathrm{T}} \mathbf{N}_{ns} - {}^C\mathbf{L}_{i,k}^{TC} \mathbf{N}_{is} \tag{16}$$

where F is the total number of points of the observed feature, ${}^Z\boldsymbol{L}_k$ is the kth observed point, ${}^C\boldsymbol{L}_{i,k}$ is the kth point of the ith landmark candidate with respect to $\boldsymbol{O}_C$ and c_i is the resulting associated map feature index. Equation (16) selects the landmark candidate with the minimum difference between the projection of the observed points ${}^Z\boldsymbol{L}_k$ on the plane parameterised by $\boldsymbol{N}_{ns}$, and the projection of the ith map feature on the plane parameterised by ${}^C\boldsymbol{N}_{is}$ as the final feature association.

4 Results

As we mentioned in the introduction, mapping and localisation for mobile vehicles is a challenging task, which depends on the filter (EKF, UKF or particle filter [1]), but in which the sort of perception system used and the method to extract the salient features are crucial. This reason motivates us to integrate our sensor model within a popular SLAM algorithm as FastSLAM described in Section 4.1, which presents our test platform and the data set collected. Section 4.2 concerns the resulting map obtained using our approach, which are qualitatively compared with the resulting map using a standard scan-matching technique [4], and quantitatively through the robot pose error with respect to the G^{2}O framework [23].

4.1 Experimental conditions

We tested our approach on a Pioneer 3DX mobile robot equipped with an onboard computer at 1.5 GHz, an omnidirectional vision setup composed of a RemoteReality parabolic mirror with a diameter of 74 mm, a UI-2230SE-C camera with a resolution of 1024×768 pixels, and a URG-04LX LRF (Fig. 4*a*).

We collected our data set at the PIV building of the University of Girona, which has three floors. These data sets include the LRF readings, the corresponding omnidirectional images and the robot odometry. Fig. 4*b* shows two columns of omnidirectional images, each column corresponding to the first and third floor of the PIV building, respectively. These images show the environmental conditions in which the data sets were collected. It can be observed that there are illumination changes and occlusions caused by pedestrians assuring in this way a real-world experiment. The data set corresponding to the first floor has a path length of ~ 80 m and that on the third floor a length of ~ 55 m.

In general, the SLAM problem can be solved using the iterative methods (Kalman filters, particle filters and occupancy grids), or using the global minimisation methods (Bundle adjustment and expected maximisation) [1, 2]. The global minimisation methods require the entire data set, which is not acceptable for real-time operation, besides these methods have data association issues [12]. The family

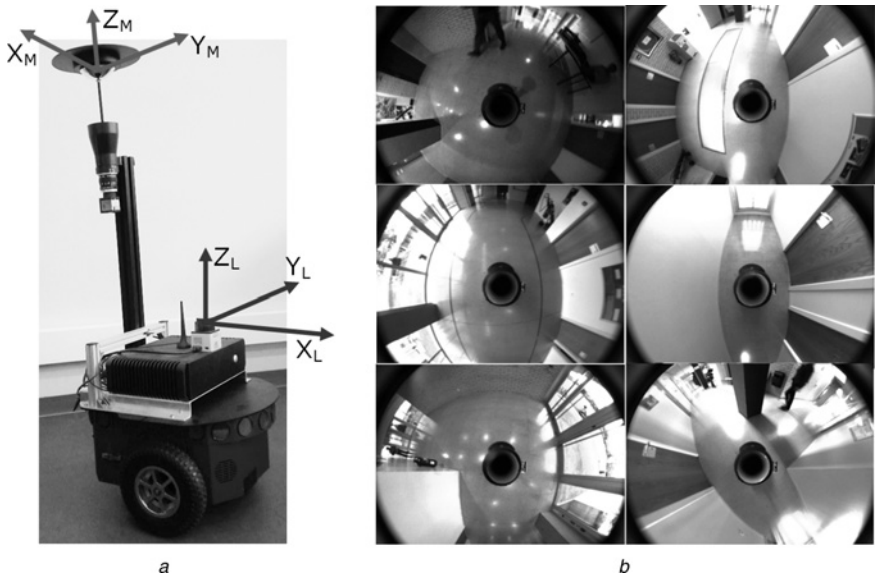

Fig. 4 *Experimental conditions*

a Robot platform and experimental setup
b Omnidirectional images samples taken from the collected data set. Each column corresponds to the first and third floor of the PIV building, respectively

of Kalman filters assumes a high-dimensional Gaussian distribution of the error source, the robot pose and the map features, however, the mapping and localisation problems involve non-linear functions and unknown error sources, the latter cannot be modelled using an exact mathematical representation. In this context, particle filters can handle these kind of problems sampling from an estimate probabilistic distribution and improving it recursively. In this paper, we used the FastSLAM [15, 16] algorithm, which uses particle filtering and we have adapted it to support our sensor model.

In our implementation, the system state is composed of the robot position $\boldsymbol{X} = [x_R, y_R, \theta_R]$ and the vertical edge positions ($\boldsymbol{M}$). The observations are that the range, azimuth and elevation of the vertical edges ($\boldsymbol{Z}$) and the control data ($\boldsymbol{U}$) are obtained from scan matching [4]. As conditional independence can be assumed given the robot poses, the posterior can be factored as depicted in (17).

$$p(x_t, M|z_{1:t}, u_{1:t}) = p(x_t|z_{1:t}, u_{1:t}) \prod_{n=1}^{N} p(m_n|x_t, z_{1:t}) \quad (17)$$

where t is the current time step and N is the current number of features. In FastSLAM, each particle is denoted by $y_t^k = x_t^k, \mu_{1,t}^k, \Sigma_{1,t}^k, \ldots, \mu_{N,t}^k, \Sigma_{N,t}^k$, where k is the particle index, x_t^k is the robot position estimate, and $\mu_{n,t}^k$ and $\Sigma_{n,t}^k$ are the mean and variance of the Gaussian representing the nth landmark location of the kth particle, which are estimated using independent Kalman filters. The filtering process overview is explained as follows:

Getting measures: Vertical edge positions with respect to the LRF are gathered: $z_{n,t} = [\rho_{n,t}\, \phi_{n,t}\, \psi_{n,t}]^T$, where $\rho_{n,t}$, $\phi_{n,t}$ and $\psi_{n,t}$ are the range, the azimuth and elevation of the nth vertical edge at time t (Section 3.2).
Sampling new poses: A new pose x_t is sampled according to the motion posterior $x_t^k \tilde{p}(x_t^k|x_{t-1}^k, u_t)$, where x_{t-1}^k is the previous estimate for the robot location in the kth particle, and u_t is the command motion obtained from scan-matching.
Data association: Given the current set of landmarks for the kth particle, the current set of observations z_t and the current predicted pose x_t^k, the data association vector H_t is computed (Section 3.3).
Update observed features: For each observed feature in each kth particle, the standard EKF expressions were applied to obtain the mean $(\mu_{H_t,t}^k)$ and covariance $(\Sigma_{H_t,t}^k)$ of these features as shown (18), (19) and (20).

$$\boldsymbol{K}_t^k = \sum_{H_t,t-1}^{k} \boldsymbol{J}_t^{kT} \left(\boldsymbol{J}_t^k \sum_{H_t,t-1}^{k} \boldsymbol{J}_t^{kT} + \boldsymbol{Q}_t \right)^{-1} \quad (18)$$

$$\mu_{H_t,t}^k = \mu_{H_t,t-1}^k + \boldsymbol{K}_t^k \left(z_t - \widehat{z_t^k} \right) \quad (19)$$

$$\sum_{H_t,t}^{k} = \left(I - \boldsymbol{K}_t^k \boldsymbol{J}_t^k \right) \sum_{H_t,t-1}^{k} \quad (20)$$

where $\boldsymbol{J}_t^k$ is the Jacobian of the measurement model with respect to the feature coordinates (see Section 3.2) and $\boldsymbol{Q}_t$ is the sensor uncertainty.

Re-sampling: We use a low variance re-sampling method, where the importance factor of each particle depends on their measurement probability.

4.2 Real experiments

Fig. 5 shows two maps, the first one (Fig. 5*a*) was built using scan matching and the second one (Fig. 5*b*) depicts our estimated map, both on the CAD map of the first floor of the PIV building in order to see their correspondence. Our estimated map shows the robot path and the landmarks estimated positions of the best particle. By comparing both the maps, it is clear that our approach obtains a better match with respect to the CAD drawing. The data set was taken in the presence of pedestrians, who are shown by the arrows in both maps. Despite of this fact, our approach obtains a coherent representation of the environment. Considering the first column of Fig. 4*b*, it is worth to note that the omnidirectional images correspond to the first loop at the left of Fig. 5, the first corridor above the office No. 005 and the back exit of PIV building below the office No. 015, respectively.

Fig. 6 shows the results for the second experiment, the scan matching-based map (Fig. 6*a*) and our estimated map (Fig. 6*b*). The main goal of the second experiment was to test our approach along long corridors, since it is well known that scan-matching techniques have a tendency to estimate shorter displacements than expected. This fact can be observed in Fig. 6*a* enclosed using discontinuous circle. However, our approach corrects this error and the rotation error introduced after the 180° rotation performed by the robot on the top-right part of the map. The data set was taken in the presence of walking people; despite of this fact, it can be observed that our approach shows that the robot path and the environment representation are in consistent with CAD map of this floor.

Fig. 7*a* shows a zoomed region of Fig. 6*a*. This figure shows the consistency between the laser scan and the corresponding vertical edges in the scene. Using our approach, one omnidirectional image and the corresponding LRF trace are only needed in order to obtain a local 3D reconstruction of the scene. This has many advantages: it avoids dealing with the scale factor problem when only a vision sensor is used, it gives metric information in real time and it solves disambiguation caused by dynamic environments.

In the same way as before, Fig. 7*b* shows a zoomed region of the third floor map in order to visualise the consistency in the vertical edge positions with respect to the laser scan. Observing Fig. 7*b*, it can be noted when groups of vertical edges put together, these edges correspond to those detected in the frame doors. One of the improvements we are working out is using this information and the LRF trace in order to obtain the 3D planes corresponding to the scene, instead of obtaining them only from the catadioptric image analysis as the approach proposed by [21].

Despite the fact that vertical edges are widely present in indoor scenes, using them as observations in a SLAM framework has some drawbacks: the changes in the illumination play an important role in detecting them, which introduces repeated features increasing the data association computation time (see Figs. 7*a* and *b*); in many situations, the vertical edges appear so close to each other (e.g. door jamb) or parallax, since in our experimental setup the LRF is not placed right above the omnidirectional camera. These

doi: 10.1049/iet-cvi.2011.0214

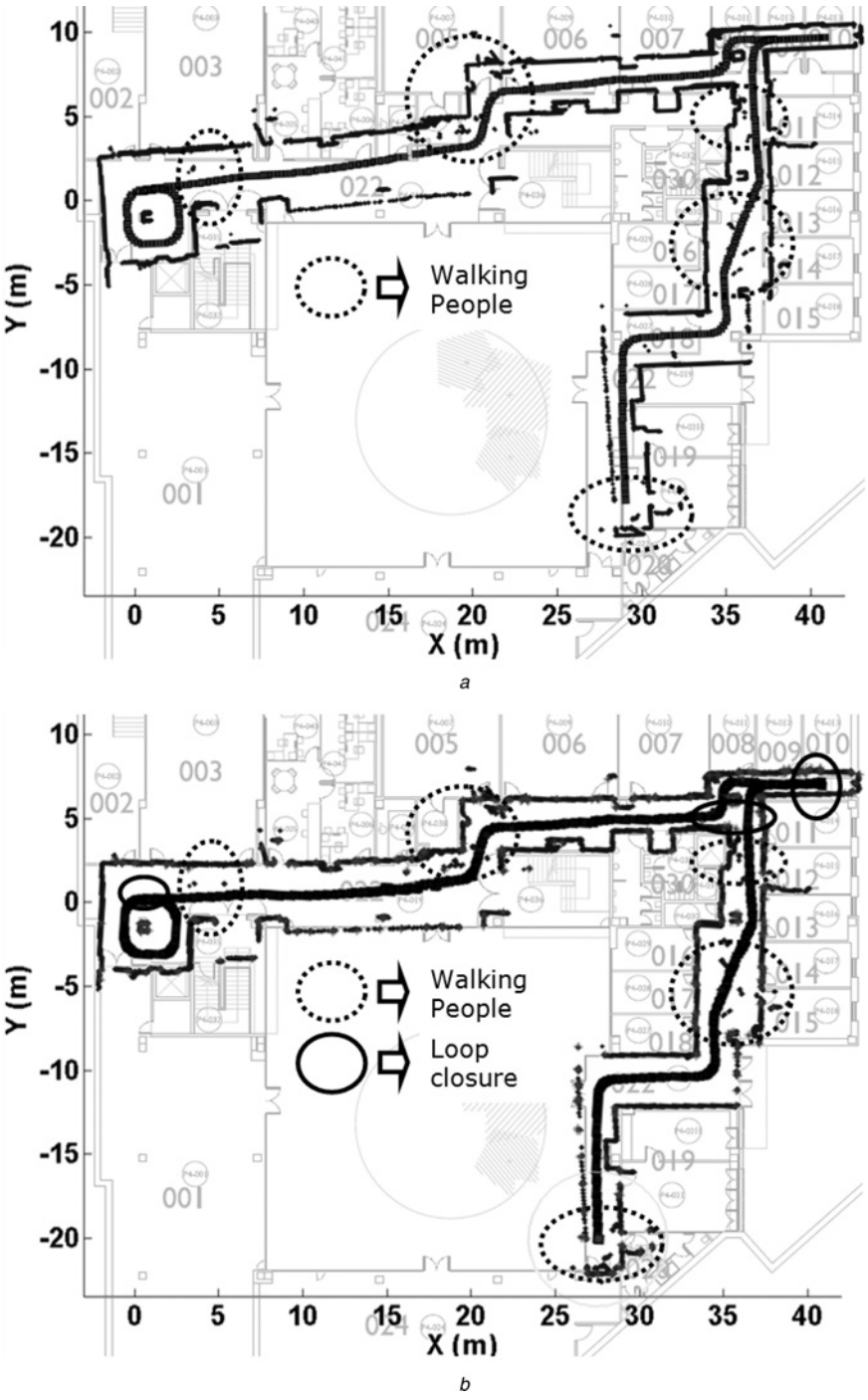

Fig. 5 *First experiment: SLAM at the first floor of PIV building*
a Scan matching map
b Estimated map using our approach

situations cause data association failures and then the robot position error increases. This was another reason to select the FastSLAM algorithm, since in these situations EKF-based SLAM solutions make the EKF filter to diverge [1]. Although the FastSLAM framework holds as many hypotheses as particles exist, however, after the first increment in the robot position error, the probability of further errors increases, temporarily degrading the map quality, but not causing the filter to diverge.

To obtain the ground truth data, we used the G^2O framework [23]. This algorithm provides a solution for batch optimisation of graph-based non-linear error functions. We introduced the robot trajectory of the first experiment as a graph, including nodes, edges and constraints. The output of the algorithm is an estimate of the robot trajectory obtained after a non-linear minimisation using Levenberg–Marquardt. Using this output, we were able to extract the X–Y error and the heading error of the robot along its path (Fig. 8). Observing the X–Y and heading errors shown in Figs. 8*a* and *b*, it is worth noting the error decreasing at steps 50, 150 and 225 approximately because of the loop closure, as depicted in Fig. 5 within the circles with solid line. After step 240 approximately, the particle filter depends on the sensor model to accurately

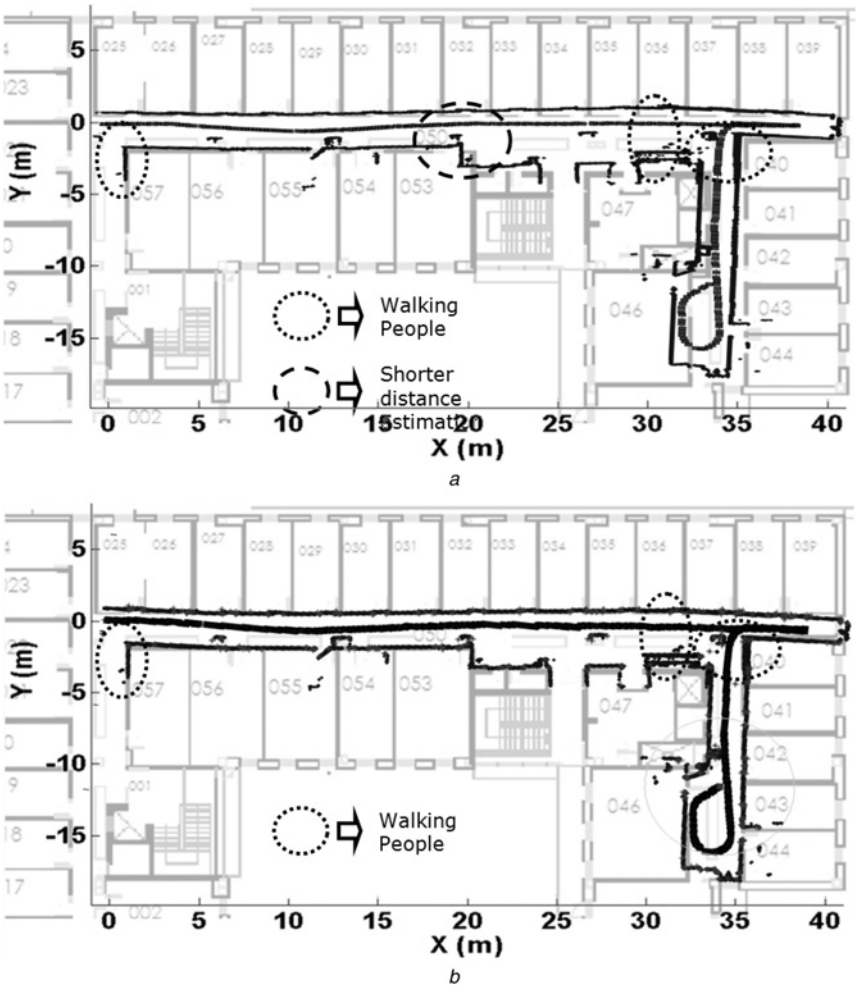

Fig. 6 *Second experiment: SLAM at the third floor of the PIV building*

a Scan matching map
b Estimated map using our approach

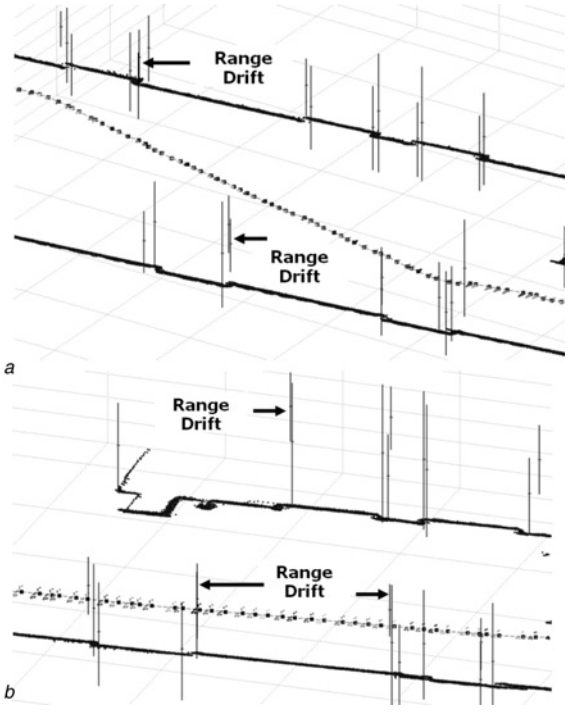

Fig. 7 *Zoomed region of third floor map*

a Details of Fig. 5 with 3D vertical edges
b Detail of Fig. 7 with 3D vertical edges

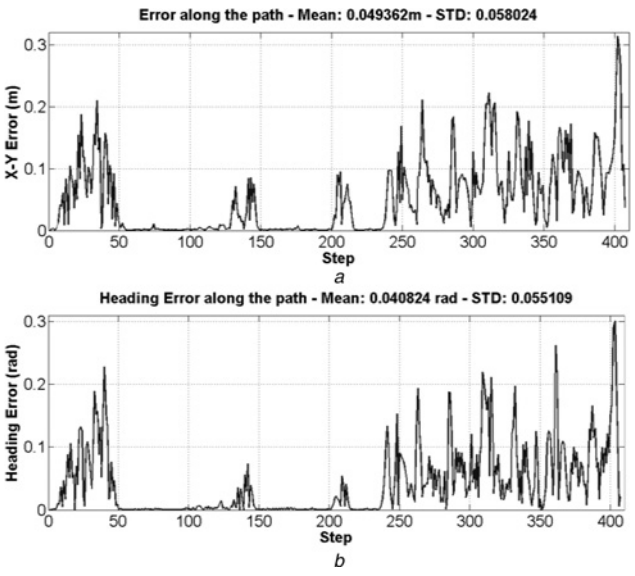

Fig. 8 *X–Y error and the heading error of the robot along its path*

a *X–Y* error in meters along the robot trajectory
b Heading error in radians along the robot trajectory

doi: 10.1049/iet-cvi.2011.0214

detect the vertical edges and associate them with map features. This can be observed in the shape of the error graphs, since periodically there is a rise and fall of the X–Y and heading errors because of vertical edge features coming into and leaving from the field of view.

5 Conclusions

We have presented a sensor model based on a range-augmented omnidirectional vision sensor and used it to solve the SLAM problem. The proposed sensor model is based on the extrinsic calibration of a LRF and an omnidirectional camera. Using this calibration, the 3D position of vertical edges were extracted and considered as observations in our implementation of the FastSLAM algorithm. Our approach provides metric information and appearance-based environmental description using one omnidirectional image and the corresponding LRF trace. Using our approach, two problems are solved: the scale factor and the data association disambiguation based on the unified spherical model for catadioptric cameras in dynamic environments where the uncertainty becomes important.

Observing the results achieved, alternative research directions can be considered in order to improve the proposed approach, such as exploiting other landmark attributes that could be useful for data association; taking advantage of the metric relationship between the 3D vertical edges and the LRF trace in order to extract the local 3D planes; and improving the loop closure detection using the feature management method proposed in [24, 25], instead of relying only on the data association performance; finally, hierarchical SLAM methods as described in [26] can be used to improve the map consistency in large-scale environments.

6 Acknowledgments

This work has been partially supported by the project RAIMON – Autonomous Underwater Robot for Marine Fish Farms Inspection and Monitoring (Ref. CTM2011-29691-C02-02) funded by the Spanish Ministry of Science and Innovation, the LASPAU-COLCIENCIAS grant no.136-2008, the University of Valle contract no. 644-19-04-95, and the consolidated research group's grant no. SGR2009-00380.

7 References

1 Durrant-Whyte, H., Bailey, T.: 'Simultaneous localization and mapping (SLAM): Part I – the essential algorithms', *Robot. Autom. Mag.*, 2006, **13**, pp. 99–110
2 Bailey, T., Durrant-Whyte, H.F.: 'Simultaneous localisation and mapping (SLAM): Part II – state of the art', *Robot. Autom. Mag.*, 2006, **13**, pp. 108–117
3 Nieto, J., Bailey, T., Nebot, E.: 'Recursive scan-matching SLAM', *Robot. Autonom. Syst.*, 2007, **55**, pp. 39–49
4 Besl, P.J., Mackay, N.D.: 'A method for registration of 3D shapes', *IEEE Trans. Patt. Anal. Mach. Intell.*, 1992, **14**, pp. 239–256
5 Silveira, G., Malis, E., Rives, P.: 'An efficient direct approach to visual SLAM', *IEEE Trans. Robot.*, 2008, **24**, (5), pp. 969–979
6 Cummins, M., Newman, P.: 'Highly scalable appearance-only SLAM – FAB-MAP 2.0'' (Robotics Science and Systems (RSS), Seattle, USA, 2009) pp. 1–8
7 Ranganathan, A., Dellaert, F.: 'Online probabilistic topological mapping', *Int. J. Robot. Res.*, 2011, **30**, pp. 755–771
8 Wongphati, M., Niparnan, N., Sudsang, A.: 'Bearing only FastSLAM using vertical line information from an omnidirectional camera'. IEEE Int. Conf. Robotics and Biomimetics, 2009, pp. 1188–1193
9 Scaramuzza, D.: 'Omnidirectional vision: from calibration to robot motion estimation', PhD thesis, ETH Zurich, 2008
10 Biber, P., Anderson, H., Duckett, T., Schilling, A.: '3D modeling of indoors environments by a mobile robot with a laser scanner and panoramic camera'. Int. Conf. on Intelligent Robots and Systems, IROS, 2004, **4**, pp. 3430–3435
11 Gallegos, G., Meilland, M., Rives, P., Comport, A.I.: 'Appearance-based SLAM relying on a hybrid laser/omnidirectional sensor'. IEEE/RSJ Int. Conf. Intelligent Robots and Systems, Taiwan, Taipei, 2010, pp. 3005–3010
12 Mei, C.: 'Laser-augmented omnidirectional vision for 3D Localisation and Mapping', PhD thesis, INRIA Sophia Antipolis, France, 2007
13 Bacca, E.B., Mouaddib, E., Cufi, X.: 'Embedding range information in omnidirectional images through laser range finder'. IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IROS), Taiwan, Taipei, 2010 pp. 2053–2058
14 Neira, J., Tardos, J.D.: 'Data association in stochastic mapping using the joint compatibility test', *IEEE Trans. Robot. Autom.*, 2001, **17**, (6), pp. 890–897
15 Montemerlo, M., Thrun, S.: 'Simultaneous localization and mapping with unknown data association using FastSLAM'. Proc. IEEE Int. Conf. Robotics and Automation, 2003, ICRA'03, 2003, vol. 2, pp. 1985–1991
16 Thrun, S., Burgard, W., Fox, D.: 'Probabilistic robotics' (MIT Press, Cambridge, MA, 2005)
17 Geyer, C., Daniilidis, K.: 'A unifying theory for central panoramic systems and practical applications'. Eur. Conf. Computer Vision, 2000, vol. 2, pp. 445–461
18 Mei, C., Rives, P.: 'Calibration between a central catadioptric camera and a laser range finder for robotic applications'. Proc. IEEE Int. Conf. Robotics and Automation, ICRA, 2006, pp. 532–537
19 Mei, C., Calibration Toolbox for omnidirectional cameras. Available at http://www.robots.ox.ac.uk/~cmei/Toolbox.html, 2011
20 Eggert, D., Lorusso, D.A., Fisher, R.: 'Estimating 3-D rigid body transformations: a comparison of four major algorithms', *Mach. Vis. Appl.*, 1997, **9**, (5–6), pp. 272–290
21 Bazin, J.-C., Kweon, I., Demonceaux, C., Vasseur, P.: 'Rectangle extraction in catadioptric images'. IEEE 11th Int. Conf. Computer Vision, 2007, ICCV 2007, 2007, pp. 14–21
22 Barreto, J., Araujo, H.: 'Geometric properties of central catadioptric line images and their application in calibration', *IEEE Trans. Patt. Anal. Mach. Intell.*, 2005, **27**, (8), pp. 1327–1333
23 Kuİmmerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: 'G2O: A general framework for graph optimization'. Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), 2011, pp. 3607–3613
24 Bacca, B., Salvi, J., Batlle, J., Cufi, X.: 'Appearance-based mapping and localization using feature stability histograms', *Electron. Lett.*, 2010, **46**, (16), pp. 1120–1121
25 Bacca, B., Salvi, J., Cufí, X.: 'Appearance-based mapping and localization for mobile robots using a feature stability histogram', *Robot. Autonom. Syst.*, 2011, **59**, (10), pp. 840–857
26 Aulinas, J., Salvi, J., Llado, X., Petillot, Y.: 'Local map update for large-scale SLAM', *Electron. Lett.*, 2010, **46**, (8), pp. 564–566