An Application of Omnidirectional Vision to Grid-based SLAM in Indoor Environments

Vito Macchia, Stefano Rosa, Luca Carlone, Basilio Bona

Abstract— In this work we study the use of an omnidirectional camera for the estimation of a consistent metric representation of an indoor scenario. The proposed approach is based on Rao-Blackwellized Particle Filters and allows the robot to use a single vision sensor for estimating an occupancy grid map of the environment. The prediction phase of the filter is performed by means of an accurate visual odometry algorithm, whereas the update phase is based on floor segmentation, allowing to treat the omnidirectional camera in full similarity with laser-based approaches to SLAM. The technique is validated in simulation and through real experiments and it is shown to perform consistent map estimation while reducing the costs and the equipment of the robotic system.

I. INTRODUCTION

The capability of building a map while traveling in an unknown environment represents one of the main prerequisite of truly autonomous mobile robots. For this purpose a robot has to jointly estimate both its pose and a representation of the surrounding objects with respect to a given reference frame. In order to create a consistent map, a crucial issue arises when the robot comes back to already traveled areas (i.e., when *loop closing* occurs), since it is crucial that it recognizes the place revisiting episode. If the loop closing occurrences are neglected and only robot motion is estimated, the positioning error unavoidably accumulates, preventing correct map estimation; this case is often referred to as *relative or odometric localization*. In vision-based setups several techniques do exist for the estimation of odometric motion of the robot from the frames acquired during motion using monocular, omnidirectional [17] or stereo cameras [9]. Some authors tried to alleviate the error accumulation by considering the loop closing episodes in the estimation procedure: in [3] loop closing detection is based on image similarity, whereas in [16] a vocabulary tree is employed to recognize already visited areas and bundle adjustment is applied for correcting the position estimates.

In Probabilistic Robotics the joint estimation of robot pose and world representation is known as *Simultaneous Localization And Mapping* (SLAM). When dealing with landmark-based models of the environment, Extended Kalman Filter (EKF) has been demonstrated to be an effective solution to SLAM: EKF is applied as observer of a dynamic system whose states include both robot pose and landmark positions

[4]. In indoor environments, however, metric representations are often employed, in the form of occupancy grid maps [6], [15]. In a grid map representation, the world model is discretized in a regular grid and for each cell of the grid an information about the probability of the cell being occupied is maintained. Such a model is desirable for several reasons: widespread path planning algorithms (A*, D*, etc., see [12]) can be adapted to deal with grid maps; grids are intuitive models of the environment and can be easily understood by a human user when displayed on a human machine interface (HMI); finally they overcome the problem of *data association* that occurs in landmark-based SLAM and that can be difficult to tackle in symmetric, structured environments. In this last scenario Particle Filters (see [1] and the references therein) constitute a widespread and successful approach for the estimation of SLAM posterior.

Robotic literature encompasses several contributions on Rao blackwellized Particle Filters (RBPF) SLAM in indoor environments, see, e.g., [19] and [23]. In such approaches the wheel odometry is usually employed in the *prediction* phase of the particle filter, whereas exteroceptive measurements are acquired from laser range finders and used for sample weights update. The laser range finder or laser scanner is a commonly used sensor in robotics and allows to acquire accurate measurements of distance from obstacles within a given range (usually 5-16 m). This sensor, however, is not suitable for several applications due to its weight burden, or simply for its cost. Moreover it can be difficult to extract semantic information from laser data as required for high level reasoning in autonomous systems. Odometry estimation by means of wheel encoders, on the other hand, can provide poor motion estimation.

As the aforementioned drawbacks can be technological barriers for low-cost low-power indoor robotic applications, we propose a methodology for estimating grid-based SLAM posterior using a single omnidirectional camera and Rao-Blackwellized Particle Filters. Our technique allows to treat the problem in full similarity with respect to laser-based techniques. The prediction phase of the filter is based on visual odometry, whereas a floor segmentation algorithm is used to extract range information for filter update.

This work is structured as follows. In Section II-A we introduce some notions on Rao-Blackwellized Particle Filter SLAM. Then, in Section II-B and II-C, we describe the proposed approach, whereas in Section III we present the results of simulations and real tests which validate the approach. Conclusions are drawn in Section IV

II. RBPF-SLAM AND OMNIDIRECTIONAL VISION

A. SLAM with Rao-Blackwellized Particle Filters

Although the high dimensionality of state space in grid-based SLAM makes challenging the application of sample-based representations of the posterior of robot pose and occupancy grid map, an elegant solution to reduce dimensionality of the sampling space can be obtained through Rao-Blackwellization [5]. Since the map probability can be computed analytically given the robot path, it is possible to factorize the joint probability as follows:

$$p(\mathbf{x}_{1:k}, m \mid \mathbf{z}_{1:k}, \mathbf{u}_{0:k-1}) = p(m \mid \mathbf{x}_{1:k}, \mathbf{z}_{1:k})$$
$$\cdot p(\mathbf{x}_{1:k} \mid \mathbf{z}_{1:k}, \mathbf{u}_{0:k-1}) \quad (1)$$

In (1) the state includes the robot trajectory $\mathbf{x}_{1:k} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k\}$ and the map m, both estimated from the measurements $\mathbf{z}_{1:k} = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_k\}$ and the commands $\mathbf{u}_{0:k-1} = \{\mathbf{u}_0, \mathbf{u}_1, \ldots, \mathbf{u}_{k-1}\}$. Equation (1) provides the basis for RBPF SLAM: the particle filter is applied to the problem of estimating potential trajectories and a map hypothesis is associated to each sample. According to particle filter framework the posterior of robot trajectory is approximated by a set of weighted random samples:

$$p(\mathbf{x}_{1:k} \mid \mathbf{z}_{1:k}, \mathbf{u}_{0:k-1}) \approx \sum_{i=1}^{n} \omega_k^{[i]} \delta(\mathbf{x}_{1:k} - \mathbf{x}_{1:k}^{[i]}) \quad (2)$$

where n is the particle set size, $\mathbf{x}_{1:k}^{[i]}$ is the pose of the i-th particle at time k, $\omega_k^{[i]}$ is the corresponding weight $\left(\sum_{i=1}^{n} \omega_k^{[i]} = 1\right)$, and $\delta(\cdot)$ is the Dirac delta function. Filter prediction is obtained by drawing particles from the *proposal distribution* $\pi(\mathbf{x}_{k+1} \mid \mathbf{x}_k, \mathbf{u}_k)$, which is often approximated with a Gaussian density, whose mean and covariance depend on the odometric information $\mathbf{u}_k$, whereas the weights are updated according to [20]:

$$\omega_k^{[i]} = \omega_{k-1}^{[i]} p(\mathbf{z}_k \mid \mathbf{x}_{k-1}^{[i]}, \mathbf{u}_{k-1}, m_{k-1}^{[i]}), i = 1, \ldots, n. \quad (3)$$

Particles degeneracy (i.e., the situation in which most part of the sample set has negligible weight) is then prevented by a resampling phase that randomly chooses the samples which best fit current and past observations, according to particles weights. A common condition for resampling is based on the *effective sample size* [1], which is an approximated measure of particle diversity:

$$\tilde{N}_{eff} = \frac{1}{\sum_{i=1}^{n} \left(\omega_k^{[i]}\right)^2}. \quad (4)$$

Particles are re-sampled if the previous quantity drops below a given threshold, usually fixed to $n/2$, see [21].

In the following sections we will discuss how information from omnidirectional vision can be included in the filter for the SLAM posterior estimation.

B. Visual Odometry and Filter Prediction

According to the previous section, in order to perform the prediction phase of the particle filter, the robot needs to estimate the relative motion from the previous pose $\mathbf{x}_{k-1} = [x_{k-1} \ y_{k-1} \ \theta_{k-1}]^\top$ to the current pose $\mathbf{x}_k$, i.e., the odometric information $\mathbf{u}_{k-1} = [\Delta x_{k-1} \ \Delta y_{k-1} \ \Delta \theta_{k-1}]^\top$. As pointed out in Section I, however, several approaches allow to estimate robot motion from visual information. Here we recall a simple approach based on feature matching, between consecutive frames acquired at time $k-1$ and k, that resembles the procedure proposed in [17]. We assume that a camera model and the camera height h are available and the robot motion is planar.

Feature keypoints (SIFT, SURF, Shi-Tomasi, etc. [10]) are first extracted from the frames, restricting the search to the annulus between a radius r_{min} and a maximum distance r_{max} from the camera center (see Figure 1). Features detected from this search region have high probability to belong to the ground plane, which, as explained later, is a crucial prerequisite for retrieving the relative motion of the rover. The keypoints in the two images are then pairwise matched using suitable criteria on the Euclidean distance between feature descriptor vectors (double consistency check, KD-tree, etc.). Matched feature points are reprojected on the normalized image coordinate system and used for estimating the planar roto-translation that describes robot motion between each image pair [17]. Let $\mathcal{F}_{k-1} = \{x_{f,k-1}^{[1]}, x_{f,k-1}^{[2]}, \ldots, x_{f,k-1}^{[m]}\}$ and $\mathcal{F}_k = \{x_{f,k}^{[1]}, x_{f,k}^{[2]}, \ldots, x_{f,k}^{[m]}\}$ be the sets containing the normalized coordinates of features extracted from the $(k-1)$-th and k-th frames, respectively. Hence the homography describing the transformation that has to be applied to the features in $\mathcal{F}_{k-1}$, in order to obtain the corresponding keypoints in $\mathcal{F}_k$ can be estimated as the minimum of the following cost function:

$$\min_{R,t} \sum_{j=1}^{m} \|x_{f,k-1}^{[j]} - \hat{x}_{f,k-1}(R, t, x_{f,k}^{[j]})\|^2 +$$
$$+ \|x_{f,k}^{[j]} - \hat{x}_{f,k}(R^{-1}, -t, x_{f,k-1}^{[j]})\|^2 \quad (5)$$

where $R \in \mathbb{R}^{2 \times 2}$ is a rotation matrix, $t = [t_x \ t_y]^\top \in \mathbb{R}^2$ is a translation vector, $\hat{x}_{f,t}(R, t, x_{f,k}^{[j]})$ are the features in the k-th frame, roto-translated according to homography $\mathcal{H}(R, t)$, and $\hat{x}_{f,k}(R^{-1}, -t, x_{f,k-1}^{[j]})$ are the features in the $(k-1)$-th frame, once the reverse transformation $\mathcal{H}^{-1}(R, t)$ is applied. As suggested in [16], RANSAC is applied for robust homography estimation, in order to restrict the estimation to coplanar features and reduce the effect of outliers. Since feature matching can be inaccurate for rotation estimation, the aforementioned procedure can be further improved by using a more precise compass estimation using appearance-based techniques [17]. According to [11] it is possible to extract the rotational component of the homography by comparing the appearance of patches from two frames. If we call $\Delta \theta_{k-1}^*$ the compass obtained from the appearance-
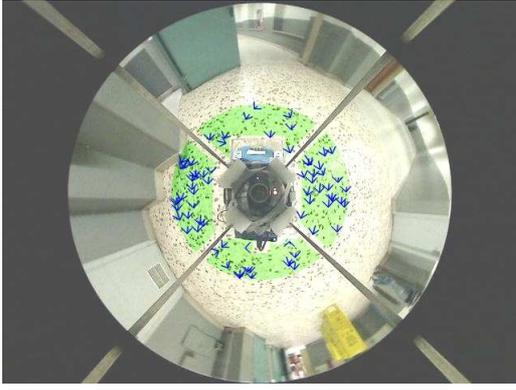
Fig. 1. Feature extraction for visual odometry. Arrows indicate the motion of the features between subsequent frames. The green mask corresponds to the annulus from which features are extracted.

based approach, the optimization problem (5) reduces to:

$$\min_t \sum_{j=1}^{m} \|x_{f,k-1}^{[j]} - \hat{x}_{f,k-1}(R^*, t, x_{f,k}^{[j]})\|^2 +$$
$$+ \|x_{f,k}^{[j]} - \hat{x}_{f,k}((R^*)^{-1}, -t, x_{f,k-1}^{[j]})\|^2 \qquad (6)$$

where R^* is the rotation matrix:

$$R^* = \left[\begin{array}{cc} \cos(\Delta\theta_{k-1}^*) & -\sin(\Delta\theta_{k-1}^*) \\ \sin(\Delta\theta_{k-1}^*) & \cos(\Delta\theta_{k-1}^*) \end{array} \right]. \qquad (7)$$

Finally the relative robot motion information $\mathbf{u}_{k-1}$ can be extracted from the homography by simply rescaling the translational components of $\mathcal{H}$:

$$\mathbf{u}_{k-1} = \left[\begin{array}{c} \Delta x_{k-1} \\ \Delta y_{k-1} \\ \Delta\theta_{k-1} \end{array} \right] = \left[\begin{array}{c} ht_x^* \\ ht_y^* \\ \Delta\theta_k^* \end{array} \right], \qquad (8)$$

where h is the camera height and t^* is the solution of the optimization problem (6). Once this information is computed the robot can apply the prediction phase of the particle filter: samples describing the pose at time $k-1$ are predicted according to $\mathbf{u}_{k-1}$ and provide the prior for the estimation of the current pose $\mathbf{x}_k$. In order to sample from the proposal distribution $\pi(\mathbf{x}_k \mid \mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ the robot also requires some guess on the uncertainty in the odometry estimation process, that can be evaluated experimentally.

C. Floor Segmentation and Weights Update

The update phase of the particle filter is based on some visual information from the omnidirectional device. As mentioned above our purpose is to treat the RBPF-SLAM problem in full similarity with laser-based approaches, hence we are looking for some distance measurement that can be extracted from the camera. We first notice that a range sensor measures distances from nearby obstacles, providing l range measurement for each discrete angle $\alpha_i,\ i = 1, \ldots, l$. If we assume that floor looks reasonably different from the walls and obstacles and obstructions in the scenario start near the floor level (no overhanging obstacle), one can retrieve this distance information by simply detecting the boundary between the ground plane and the surrounding obstacles.

In order to extract this boundary from the omnidirectional frame, the robot has to distinguish regions (in the image) that belong to the ground floor, from areas which are expected to be occupied by obstacles. Image segmentation provides a possible solution to perform such a partition: each frame, acquired by the robot, is segmented into regions which are distinguishable with respect to some characteristic or property [18]: regions that are close to current robot position are classified as belonging to the floor, and the boundary of such region provides the searched distance information. According to [7], in this work we applied a segmentation algorithm which is based on graph formalism: Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with vertices $\mathcal{V} = \{v_1, \ldots, v_{N_p}\}$ corresponding to each pixel in the image, and edges $e_{ij} = (i,j) \in \mathcal{E}$, that are incident on pairs of neighboring vertices, namely v_i and v_j. Each edge e_{ij} has a corresponding weight w_{ij}, which is a measure of the similarity between v_i and v_j. Hence image segmentation reduces to a partition of $\mathcal{G}$ in subgraphs sharing similar characteristics.

Edge weights can be computed by evaluating color or intensity difference. If the function $\Lambda : \mathbb{R}^2 \rightarrow \mathbb{R}^5$ associates each node (pixel) v_i with the corresponding feature vector containing both node coordinates v_{i_x}, v_{i_y} and the RGB values v_{i_r}, v_{i_g}, v_{i_b}, then the edges weights are computed as:

$$w_{ij} = \|\Lambda(v_i) - \Lambda(v_j)\|, \ \forall\, (i,j) \in \mathcal{E}. \qquad (9)$$

Let us define the *internal difference* within the region R_a as:

$$\mathcal{I}(R_a) = \max_{(i,j)\in\mathcal{E},\ i,j\in R_a} w_{ij}, \qquad (10)$$

and the *mutual difference* between region R_a and R_b as:

$$\mathcal{M}(R_a, R_b) = \min_{(i,j)\in\mathcal{E},\ i\in R_a, j\in R_b} w_{ij}, \qquad (11)$$

The segmentation procedure starts by considering each pixel as a different region. Such regions are pairwise compared and two regions are merged together in a bigger cluster if the following condition holds:

$$\mathcal{M}(R_a, R_b) \le \min\left(\mathcal{I}(R_a) + \frac{\gamma}{|R_a|}, \mathcal{I}(R_b) + \frac{\gamma}{|R_b|}\right), \qquad (12)$$

otherwise a boundary exist between them. In (12) γ is a constant parameter and the operator $|\cdot|$ returns the size of the region.

The described approach considers the internal characteristics of each region in the pairwise comparison, hence it is effective in segmenting image scenes with texture or non-uniform colors; moreover it is efficient, requiring a complexity of $\mathcal{O}(N_p \log N_p)$, where N_p is the number of pixels in the image [7].

Therefore the update phase of the particle filter can be simply implemented according to the following procedure:

1) image segmentation is performed on the omnidirectional image and the region closest to the robot is classified as ground floor;
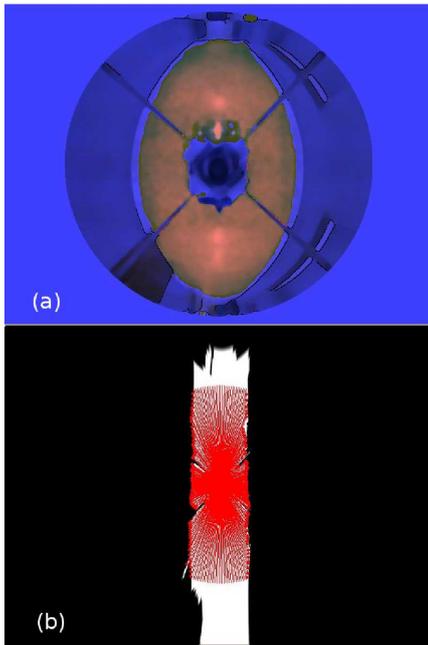2) omnidirectional image is backprojected on the ground plane, using the camera model;

Fig. 2. (a) Ground extraction using a graph-based segmentation algorithm. The red region corresponds to the floor in the omnidirectional image. (b) Ranges obtained from ray casting on the back-projected image.

Fig. 3. Layout of the virtual scenario, created in POV-Ray. The simulated robot is a P3-DX, similar to the rover used in the real tests of Section III-B.

3) a simple ray-casting procedure allows to obtain a distance information for each discrete angle around the robot, i.e., r_i, $i = 1, \dots, l$;
4) sample weights in the particle filters are updated according to the distance measurements.

The weight update is performed according to (3), where the *sensor model* $p(\mathbf{z}_k \mid \mathbf{x}_{k-1}^{[i]}, \mathbf{u}_{k-1}, m_{k-1}^{[i]})$ is usually a Gaussian function of the mismatch between the expected scan (function of estimated robot position and corresponding map hypothesis) and the actual measurement.

An example of floor segmentation is reported in Figure 2-(a) whereas the corresponding scan is shown in 2-(b). Finally we remark that, although the hypotheses for segmentation are in general quite restrictive, they are reasonable in indoor environments and experimental evidence shows that the proposed approach is applicable in common corridor or office-like scenarios.

III. EXPERIMENTAL TEST AND DISCUSSION

A. Simulations

In this section we report the results of the application of the proposed approach in a simulated environment. The scenario was designed in POV-Ray, a freely available 3D modeling software, and it is a rectangular loop with dimensions 60 m $\times$ 30 m. Each corridor is 5 m wide, whereas the camera height is 0.72 m. The layout of the simulation scenario is shown in Figure 3. The calibrated catadioptric system was designed according to the specifications of the omnidirectional device used in real tests, see Section III-B.

We now compare the outcome of the mapping process based on visual odometry for pose estimation with respect to a full SLAM approach. In the former case the range information provided by floor segmentation are simply used for mapping using a ray tracing procedure from odometric robot pose; mapping with known robot position is straightforward, and well known probabilistic approaches for updating a grid map do exist, see [22] and the references therein. In the SLAM case, instead, the uncertainty in pose estimation is considered, and localization and mapping are solved at the same time, taking into account the correlation between map and pose estimation. In the following test the Shi-Tomasi features are tracked for visual odometry, whereas the parameter γ used in the graph-based segmentation was set to 400. The map obtained using visual odometry-based pose estimation and omnidirectional camera as range device is reported in Figure 4-(a): white cells have a probability of being occupied close to zero (obstacle-free areas), black cells are likely to contain an obstacle, whereas gray cells correspond to the places in the map that have not been visited by the robot. In the simulation the robot started in the bottom-left corner of the scenario and traveled the loop for two times, ending at the bottom-right of the rectangle. When using naive visual odometry, robot pose estimation worsens as time evolves, due to odometric error accumulation; moreover such an approach is not able to detect and exploit loop closing occurrence. Hence, after the robot traveled the whole rectangle in anti-clockwise direction, coming back to the bottom-left corner, the mapping process tends to produce remarkable inconsistencies that become critical as longer distances are traveled. In Figure 4-(b) we report the case in which a SLAM approach is applied. The hypotheses carried on by the particle filter allow the robot to properly model the uncertainty in odometric pose estimation, whereas in the case of Figure 4-(a) the uncertain nature of motion estimation is neglected, leading to an over-confident mapping process. After the robot travels the loop for the first time, it arrives in a region of the environment that was already mapped; therefore the update phase (Section II-C) of the particle filter assigns higher weights to samples that are consistent with the estimated map model. Usually, after the loop closing episode, few samples, which are the best candidates for modeling the SLAM posterior, have high weights and this leads $\tilde{N}_{eff}$ to drop below the threshold for
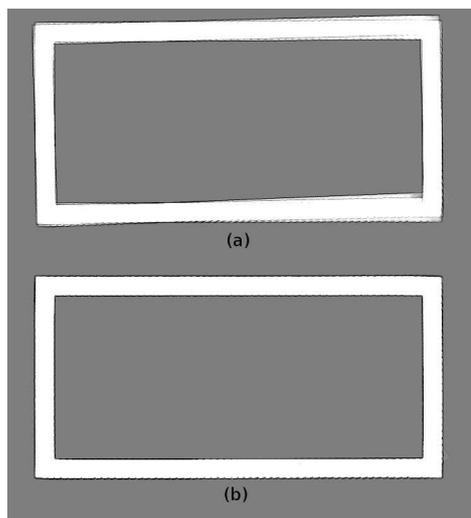
Fig. 4. (a) Estimated grid map using visual odometry for pose estimation and omnidirectional camera as range device: the map is obtained with a ray tracing procedure from the odometric pose estimate. (b) Estimated map with Rao-Blackwellized Particle Filters SLAM using a single omnidirectional camera.
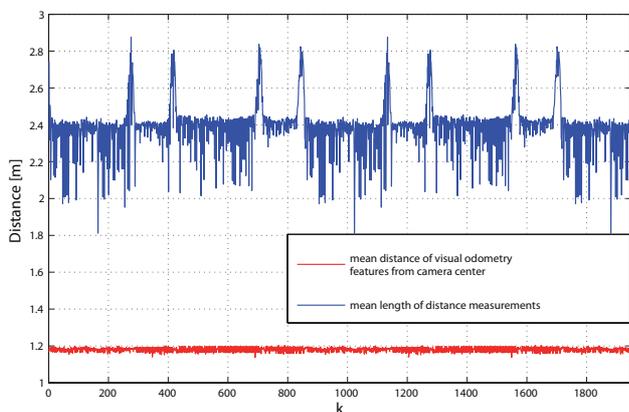

Fig. 5. Red line: mean distances of the Shi-Tomasi features from the camera center for each time step. Blue Line: mean length of the range measurements acquired at each time k.

Fig. 6. Comparison of wheel odometry (blue), laser-stabilized odometry (green), and visual odometry (red) in a real experiment.

B. Real Tests

The technique presented so far was also validated through real experiments, in the environment of Figure 1. The tests were performed with an *ActivMedia Pioneer P3-DX* equipped with omnidirectional camera and laser scanner (the latter is used only for benchmarking). Camera model is obtained using the *omnidirectional camera and calibration toolbox* [14]. We observe that the catadioptric system is a hand-assembled device composed by a cheap USB webcam and an off-the-shelf hyperbolic mirror.

We first report some results on the visual odometry approach described in Section II-B. As mentioned in the Introduction, wheel odometry is inaccurate and provides a pose estimate that quickly diverges from current robot position. For this reason, several authors proposed to improve the accuracy of relative motion estimation, by using of a scan-matching procedure among laser scans acquired at subsequent poses. This approach is usually referred to as *laser-stabilized odometry* [8]. Visual odometry results, compared with the corresponding wheel odometry and the improved approach based on scan-matching are shown in Figure 6. Once relative motion between frames is retrieved, according to (8), the vision-based pose estimate is simply computed by integration. It is possible to observe that the accuracy of visual odometry is comparable with the one obtained with scan-matching.

Finally we compare, in a real test, the SLAM posterior estimation using a traditional laser-based approach (and laser-stabilized odometry), and the corresponding outcome using a single omnidirectional camera for both prediction and update of the particle filter. In Figure 7(a) we report the estimated map with the proposed approach, whereas Figure 7(b) shows (for the same experiment) the map estimated through SLAM with wheel odometry and laser scanner. The overall length of corridor in which the experiments took place was 30 m. As expected the map estimated from camera is less accurate and misses few details of the environment. Moreover the laser has a maximum range of 8 m, whereas the segmentation provides reliable distance measurements within a range of 4 m (as observable from the smaller

resampling. The resampling phase discards the less probable samples hence assuring filter consistency and correct map estimation. It is worth noting that features are extracted in the neighborhood of robot position (we considered $r_{max} = 0.3$ m) whereas range information corresponds to regions in the image plane which are usually far from robot location (the robot is supposed to respect some safety distance from obstacles). As a consequence, visual odometry estimates and range information are expected to be independent, and the prediction and update phases of the particle filter are not biased by the use of a single sensor. Similar considerations are reported in [13], where a stereo camera is employed for both prediction and update in EKF-SLAM. In Figure 5, for each time instant, we report the mean distance of the detected Shi-Tomasi features from the camera center and the average distance measured using the omnidirectional camera as range device.
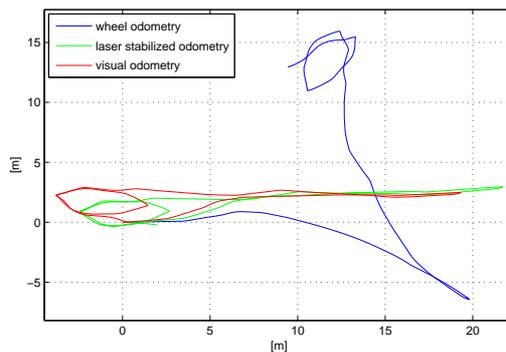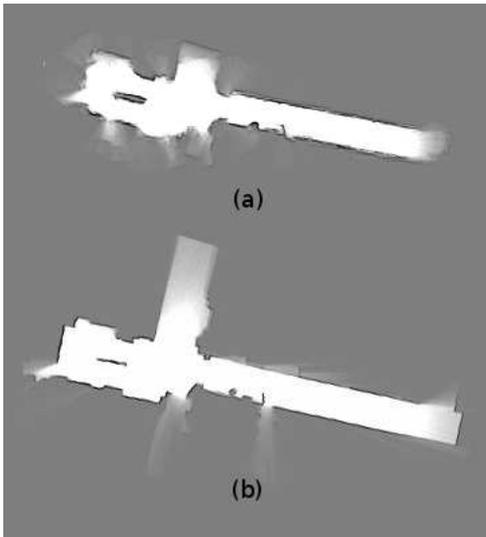
Fig. 7. (a) Map estimated through RBPF-SLAM with omnidirectional camera. (b) Map estimated through RBPF-SLAM with wheel odometry and laser range finder.

length of the corridor and the side hall in Figure 7(a)). On the other hand the occupancy grid map obtained from omnidirectional vision gives a consistent representation of the scenario and enables robot navigation while preserving the inexpensiveness of the application. Although in grid-based SLAM visual inspection is a common approach for evaluating map quality, we preferred to give quantitative evidence of the estimated map quality by comparing it with the corresponding laser-based map. For this purpose we used the metric proposed by Carpin in [2], called *acceptance index*, which gives information on map similarity, once a suitable roto-translation is applied. The acceptance index ranges from 0 to 1 and, applied to the maps of Figure 7, returned a value of 0.98, hence confirming map similarity.

IV. CONCLUSION

This work presents an application of an omnidirectional camera to Simultaneous Localization and Mapping with particle filters. The sensor is used for visual odometry and as range device at a time, providing information for both filter prediction and weights update. The proposed approach allows the robot to estimate a grid map representation of the environment by using a single vision sensor. The grid map, contrarily to landmark-based representations, can be safely used for robot navigation and it is an intuitive representation for human operators who monitor the robotic system via HMI. Moreover the application of RBPF-SLAM provides a natural way of managing loop closing episodes, thus avoiding the error accumulation problem, which is typical of visual odometry, and without need to solve complex *data association* problems, that can threaten filter consistency. The approach was validated through simulations and real tests and was shown to produce reasonable map representations, assuring remarkable advantages in terms of cost and power consumption, when compared with laser-based SLAM ap-

proaches. Current research effort is devoted to the test of the proposed technique in different indoor scenarios and to improve its robustness to environment characteristics.

REFERENCES

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filter for on-line nonlinear/non-gaussian bayesian tracking. *IEEE Trans. On Signal Processing*, 2(50):174–188, 2002.

[2] S. Carpin. Fast and accurate map merging for multi-robot systems. *Autonomous Robots*, 25:305–316, 2008.

[3] M. Cummins and P. Newman. Probabilistic appearance based navigation and loop closing. In *IEEE International Conference on Robotics and Automation (ICRA07)*, 2007.

[4] G. Dissanayake, P. Newman, H.F. Durrant-Whyte, S. Clark, and M. Csobra. A solution to the simultaneous localisation and mapping (SLAM) problem. *IEEE Trans. Robot. Automat.*, 17(3):229–241, 2001.

[5] A. Doucet, J. de Freitas, K. Murphy, and S. Russel. Rao-Blackwellized particle filtering for dynamic bayesian networks. In *Proceeding of the Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 176–183, 2000.

[6] A. Elfes. Occupancy grids: A probabilistic framework for robot perception and navigation. *PhD thesis, Department of Electrical and Computer Engineering, Carnegie Mellon University*, 1989.

[7] P.F. Felzenszwalb and D.P. Huttenloche. Efficient graph-based image segmentation. *International Journal of Computer Vision*, 59(2):167–181, 2004.

[8] A. Howard. Multi-robot simultaneous localization and mapping using particle filters. *Robotics: Science and Systems*, pages 201–208, 2006.

[9] A. Howard. Real-time stereo visual odometry for autonomous ground vehicles. In *IEEE-RSJ International Conference on Intelligent Robots and Systems (IROS'08)*, pages 3946–3952, 2008.

[10] M.Y.I. Idris, H. Arof, E.M. Tamil, N.M. Noor, and Z. Razak. Review of feature detection techniques for simultaneous localization and mapping and system on chip approach. *Information Technology Journal*, 8(3):250–262, 2009.

[11] F. Labrosse. The visual compass: Performance and limitations of an appearance-based method. *Journal of Field Robototics*, 23(10):913941, 2006.

[12] S.M. LaValle. Planning algorithms. *Cambridge University Press*, 2006.

[13] T. Lemaire, C. Berger, I.K. Jung, and S. Lacroix. Vision-based SLAM: stereo and monocular approaches. *International Journal on Computer Vision*, 74(3):343–364, 2007.

[14] A. Martinelli, D. Scaramuzza, and R. Siegwart. Automatic self-calibration of a vision system during robot motion. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA 2006)*, 2006.

[15] H.P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9(2):61–74, 1988.

[16] D. Scaramuzza, F. Fraundorfer, and M. Pollefeys. Closing the loop in appearance-guided omnidirectional visual odometry by using vocabulary trees. *Robotics and Autonomous System Journal (Elsevier)*, 58(6):TO APPEAR, 2010.

[17] D. Scaramuzza and R. Siegwart. Appearance guided monocular omnidirectional visual odometry for outdoor ground vehicles. *IEEE Transactions on Robotics*, 24(5), 2008.

[18] L.G. Shapiro and G.C. Stockman. Computer vision. *New Jersey, Prentice-Hall*, pages 279–325, 2001.

[19] C. Stachniss, G. Grisetti, and W. Burgard. Information gain-based exploration using Rao-Blackwellized particle filters. In *Proceedings of Robotics: Science and Systems*, 2005.

[20] C. Stachniss, G. Grisetti, and W. Burgard. Analyzing gaussian proposal distributions for mapping with rao-blackwellized particle filters. In *Proceedings of International Conference on Intelligent Robots and Systems (IROS'07)*, 2007.

[21] C. Stachniss, G. Grisetti, D. Hahnel, and W. Burgard. Improved Rao-Blackwellized mapping by adaptive sampling and active loop-closure. In *Proceedings of the Workshop on Self-Organization of AdaptiVE behavior (SOAVE)*, pages 1–15, 2004.

[22] S. Thrun. Robotic mapping: A survey. *Exploring Artificial Intelligence in the New Millenium*, 2002.

[23] S. Thrun, W. Burgard, and D. Fox. Probabilistic robotics. *MIT press*, 2005.