Localization through omnivision for a tour-guide robot

C. Gamallo, P. Quintía, C. V. Regueiro and M. Mucientes

***Abstract*—The localization of a mobile robot in a real environment is a complex task. In this paper, an algorithm that solves the global localization is presented. The proposal is based on a merit function that ranks different possible poses obtained from the acquired image, together with an iterative process for the minimization of that function using a particle filter. Landmark detection has been done with an omnidirectional camera pointing to the ceiling, combined with an infrared passband filter which extracts the lights. Several real experiments, both for global localization and the kidnapped robot problem, have been done in a museum. Results show a high accuracy, robustness, and real-time execution in this complex and crowded environment.**

***Index Terms*—Global localization, omnivision, tour-guide robot, kidnapping problem.**

I. Introduction

LOCALIZATION is one of the most important tasks in the field of autonomous mobile robotics. Determining the location of a mobile robot means finding the Cartesian coordinates and angular orientation relative to an external frame. A localization algorithm must be reliable, robust and executable in real time.

Different types of sensors have been used for localization: laser [16], [9], ultrasonic, or infrared sensors, and vision. Nowadays, cameras are widely used in robotics. The main advantages of these sensors are the quantity of information that can be extracted from one acquisition. This is particularly interesting for localization, as different types of landmarks can be detected using information of shape, color, etc. Also, cameras have a good performance independently of the materials objects are made off.

Popular vision approaches are feature-based [5], [13], which exploit typical properties of the environment or any distinctive and recognizable objects (landmarks), and pixel-based techniques [1], [12], that compute the correlation between images, in order to estimate the mobile robot pose.

In this paper we present an approach based in a map of landmarks and an artificial vision system to estimate the pose of the robot. We use the omnidirectional camera shown in Fig. 1. It provides a very wide field of vision (FOV of about 185^o) which covers half the space of the environment and, therefore, it can get a high amount of information in one acquisition. The camera is pointing to the ceiling and elevated 1.5 m over the robot (1.8 m over the ground) (Fig. 1). Thus its movements are restricted to the *x*-*y* plane and the noise or occlusion generated by moving people is minimized.

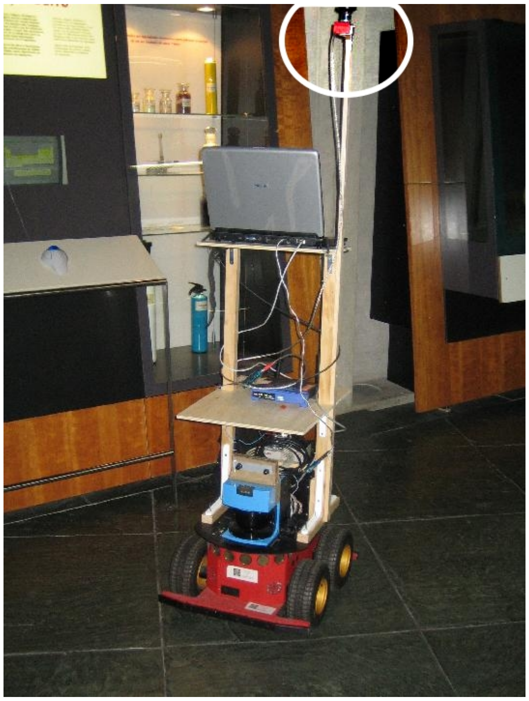


Fig. 1. The vision system in the tour-guide mobile robot, based on a Pioneer 2-AT, placed at the Domus Museum in A Coruña (Spain). The omnivision camera is marked with a circle.

The landmarks that have been used are the lights placed on the ceiling of the environment (Fig. 3). These are easy to detect, repetitive and usually visible for long trajectories. On the other hand, all the buildings have these kind of landmarks, so there is no need for prior adaptation of the environment in order to use the proposed localization method. The main problem is their individual identification, because they usually are identical, enhancing the difficulties for data association.

The key point of our solution is a function that evaluates the similarity among the positions of the landmarks detected in the image acquired with the camera and the theoretical position of the landmarks projected according to the camera model. We present experiments in a dynamic museum environment (Fig 2) which show that our algorithm can estimate the global robot pose even in situations where odometry suffers from serious noise and, also, when the robot is kidnapped.

The paper is organized as follows. Section II contains an

C. Gamallo and M. Mucientes are with the Intelligent Systems Group, Department of Electronics and Computer Science, University of Santiago de Compostela (Spain).
E-mail: cristina.gamallo@usc.es, manuel.mucientes@usc.es

C. V. Regueiro and P. Quintía are with the Computer Architecture Group, Department of Electronic and Systems, University of A Coruña (Spain).
E-mail: cvazquez@udc.es, pquintia@udc.es

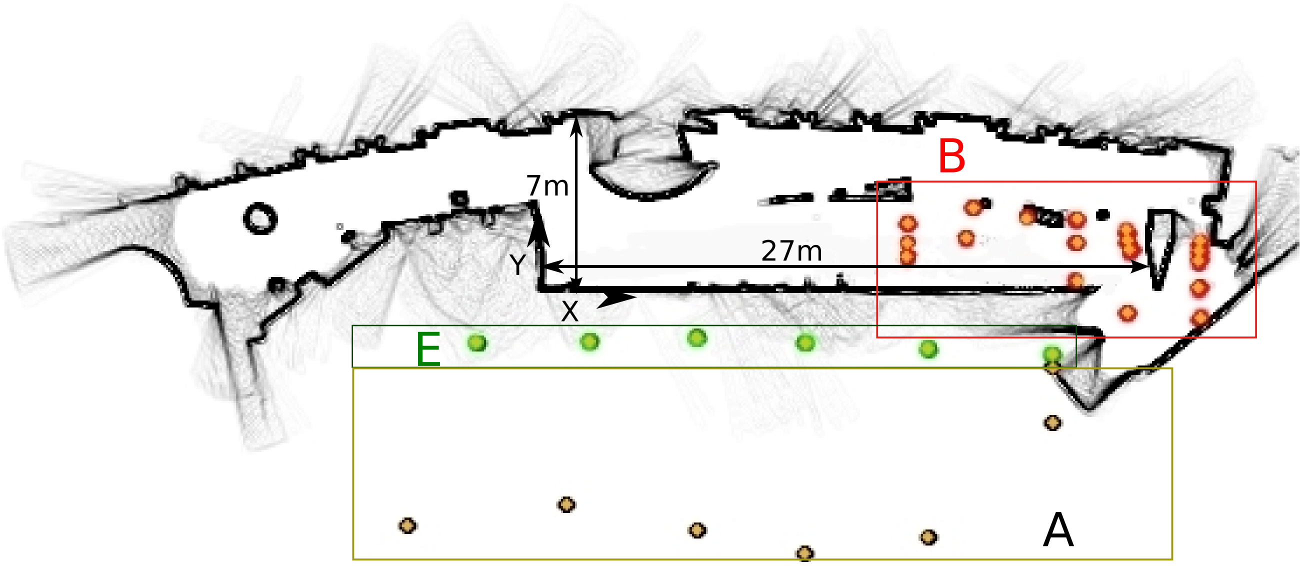
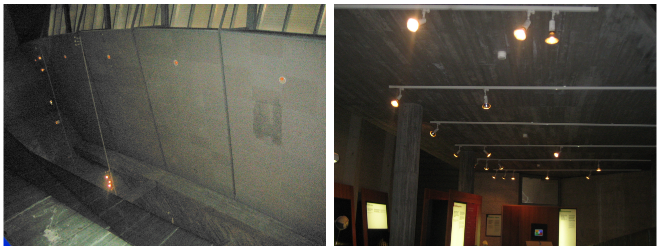


Fig. 2. 2D environment grid-map with landmarks positions (circles). The position of the landmarks is shown in table I



(a) (b)

Fig. 3. Some of the lights that have been used as landmarks: (a) Type of lights placed in areas E and A of the map (Fig. 2); (b) Type of lights placed in area B of the map lights of the map (Fig. 2).

TABLE I
POSITION (METRES) OF THE LANDMARKS IN FIG. 2.

LABEL	X	Y	Z
Lights Type E			
E1	-3.45	-2.26	11.39
E2	1.34	-2.15	11.24
E3	5.9	-2.04	11.29
E4	10.56	-2.25	11.43
E5	15.8	-2.50	11.48
E6	21.1	-2.80	11.60
Lights Type A			
A1	-6.45	-10.00	11.39
A2	0.34	-9.15	10.50
A3	5.9	-10.24	10.50
A4	10.56	-11.25	10.50
A5	15.8	-10.50	10.50
A6	21.1	-5.70	10.50
Lights Type B			
B11	14.98	2.86	3.25
B12	14.98	2.00	3.25
B13	14.99	1.69	3.29
B21	17.71	3.51	3.24
B22	17.5	2.28	3.24
B31	20.16	3.21	3.16
B41	22.13	3.08	3.16
B42	22.13	2.08	3.16
B43	22.13	0.3	3.16
B51	24.34	2.47	3.14
B52	24.34	2.07	3.14
B53	24.54	1.77	3.14
B54	24.34	-0.97	3.24
B61	27.50	1.97	3.24
B62	27.50	1.67	3.24
B63	27.50	1.37	3.24
B64	27.50	0.17	3.24
R65	27.50	-1.27	3.24

overview of related work. The next two sections describe in detail the omnivision and localization systems. Section V presents the experimental results in a real environment and, finally, the last section points out the conclusions and future work.

II. RELATED WORK

In the last years, several vision-based algorithms have been developed as solution to the localization problem. Techniques vary substantially, depending on the sensors, their geometric models, and the representation of the environment.

The probabilistic approach is the most widely used in recent publications. For example, in [16] a Monte Carlo localization algorithm is presented to solve the global localization problem using a camera. They used a visual map of the ceiling, obtained by mosaicing, and localized the robot using a simple scalar brightness measurement as the sensor input. Nervertheless, this system is sensitive to bumps and, as a result of the small FOV of the camera, at some time instants none or few lights can be seen. This provokes more uncertainty in the knowledge of the pose of the robot. A similar approach is presented in [12]. They use an omnivision camera oriented to the ceiling too, but it is based on information theory to get the global trajectory. The main problem of this work is the high computational cost.

In [1], [14], [11] a Monte Carlo localization is used, but

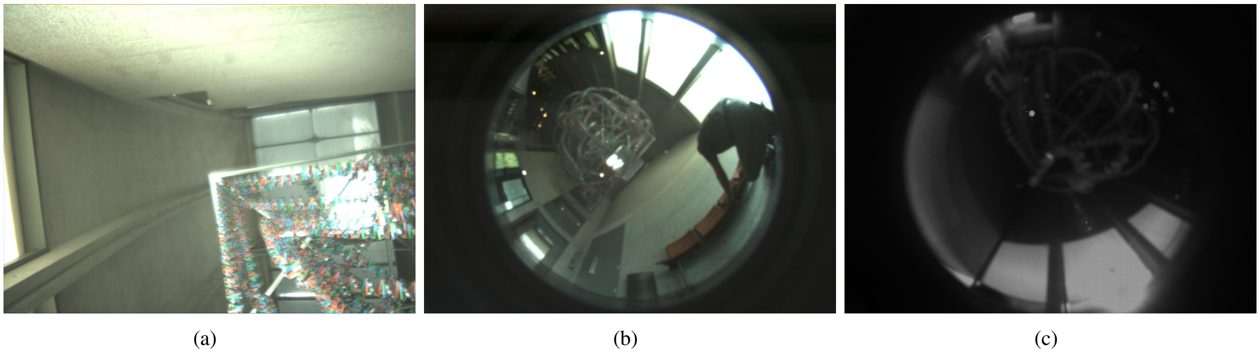


Fig. 4. (a) Original image, (b) omnidirectional image acquired without IRP filter and (c) omnidirectional image with IRP filter.

they create a database with images of every route and their poses. The robot can be localized by correlation between the captured images and the database images on real time. These systems have the drawback that they cannot work in other routes of the environment.

Menegatti et al. [10] have developed a system which uses a chromatic map of the floor to compute the robot pose. They obtained similar results to other proposed algorithms, but their system is limited to environments with natural color transitions and it is also light-sensitive.

The first work that used omnidirectional vision to localize a mobile robot was published in 1986 by Cao et al. [4]. Nevertheless, few related studies were published before the end of the nineties. Nowadays, such systems have become popular due to their low cost in addition to the benefit of having a very wide field of vision. There are two types of omnidirectional vision configurations: *catadioptric* (as in [3], [7], [13], [12]), where the camera images are obtained through a conic mirror and *dioptric*, where images are captured through a lens [11].

Other implementations use landmarks (beacons) of the environment to get the pose of the robot. For instance, in [13] the goals of a RoboCup field are used as marks and [3], [16] are based on features of the environment (corners, walls, lights ...) which were previously mapped.

Our model is similar to these ones but, in addition, we do not have the occluded landmarks problem and our process to discover landmarks is simple, fast and efficient. A similar approach (omnivision camera oriented to the ceiling) is used in [12], but it is based on information theory to get the global trajectory.

III. VISUAL SYSTEM

The vision system consists of a color digital camera MDCS2, with a *fish-eye* lens FE185CO46HA-1 and a passband infrared filter (IRP) type HOYA IR85.

The fish-eye lens has a high resolution and a wide angle of view (185^o degrees), so that the amount of information captured at a time instant is high. Due to this wide field of view, the number of landmarks in an environment can be reduced because each of them can be detected from a higher number of places (Fig. 4(b)).

The infrared passband filter *IRP* dims the light visible components and only allows to pass the components near to the IR range (Fig. 4(c)) which makes simpler the landmarks detection.

A. Camera Model

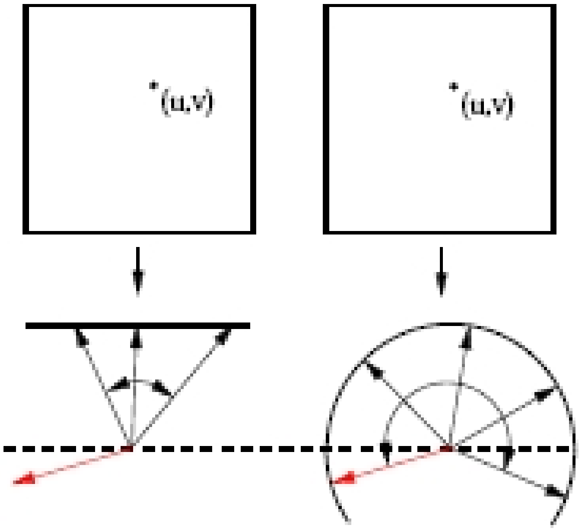


Fig. 5. The *Pin-Hole* camera model based on a flat retina (left) compared with the omnidirectional camera model based on spherical retina (right).

The camera model describes how a 3D scene is transformed into a 2D image (Fig. 6). The standard model is the *Pin-Hole*, which projects the scene on a flat retina (Fig. 5), but it is limited to cameras with $FOV << 180^0$. The other cameras require a model based on a spherical retina (Fig. 5). In our system, we have used a projection model developed by Pajdla and Bakstein [2] that describes the relation between the angle (θ) formed between the optical axis and the light ray, and the distance r from the image center (u_0, v_0) to the projection point of B (u_B, v_B) in the image (Fig. 6):

$$r = a * \tan\frac{\theta}{b} + c * \sin\frac{\theta}{d}, \tag{1}$$

where *a, b, c,* and *d* are parameters of the model. This function makes it possible to calculate the coordinates of the image *(u,v)* depending on the azimuth (φ) and the elevation (θ) (Fig. 6):

$$\left.\begin{array}{r} u = u_0 + r * cos\varphi \\ v = \beta * (v_0 + r * sin\varphi) \end{array}\right\} \qquad (2)$$

where β is the ratio between the width and the height of a pixel.

B. A Landmark Projection

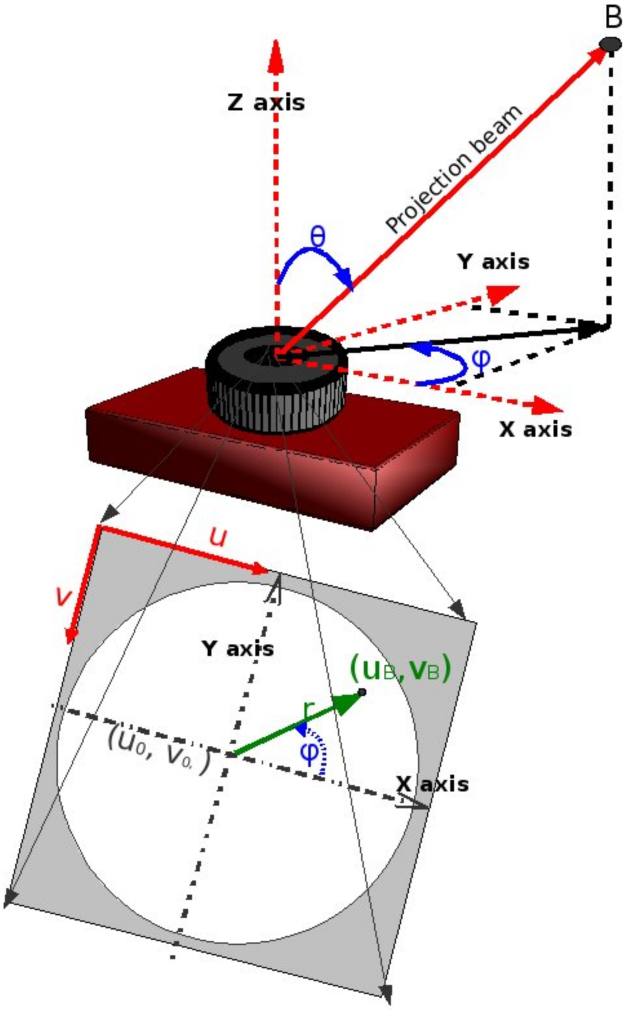


Fig. 6. Theoretical omnidirectional camera model and projection of a point B. Its projection ray is defined by the elevation (θ) and the azimuth (φ), both in the camera coordinate system. r and φ are the polar coordinates of the projected point (u_B, v_B). (u_0, v_0) are the coordinates of the image center.

A Landmark projection is the calculation of the image coordinates (u_B, v_B) for a landmark i (B in Fig. 6) in the world, given its coordinates in the world ($\mathbf{B}_i^W$) and the coordinates of the camera ($\mathbf{C}^W$).

First, we have to change the landmark coordinates ($\mathbf{B}_i^W$) to the camera reference system ($\mathbf{B}_i^C$) with the rotation matrix R_C between the camera and the world:

$$\mathbf{B}_i^C = R_C * \mathbf{B}_m^W - \mathbf{C}_W \qquad (3)$$

From $\mathbf{B}_i^C$, we obtain the elevation (θ) and the azimuth (φ) angles by applying the traditional Euclidean transformations (Fig. 6). Finally to get the landmark projection ($Proj(B_i^C)$) represented as (u_B, v_B) in Fig. 6, we apply Eqs. 1 and 2:

$$Proj(B_i^C) = (u_{B_i^C}, v_{B_i^C}) \qquad (4)$$

C. Ceiling Map Projection

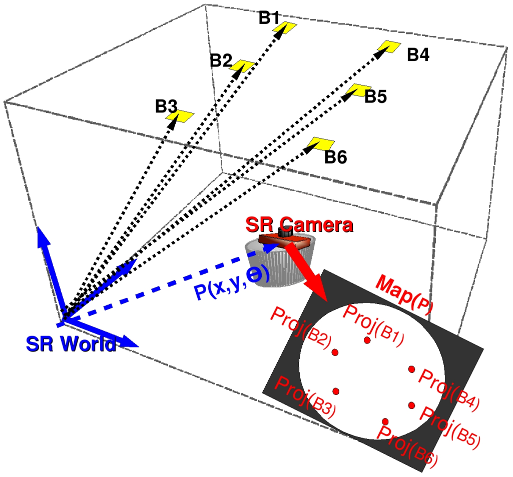


Fig. 7. A graphical example of Ceiling Map Projection for six landmarks.

Ceiling map projection, $Map(P)$, is an image that has been generated based on the theoretical camera model described in Sec. III-A. It is composed of a set of points $\{(u_{B_i}, v_{B_i}), \ldots\}$ that correspond with the coordinates of each landmark in the image reference system when the robot is placed at pose P. The process is detailed in Alg. 1, and a graphical example is shown in Fig. 7.

Algorithm 1 Calculate $Map(P)$.

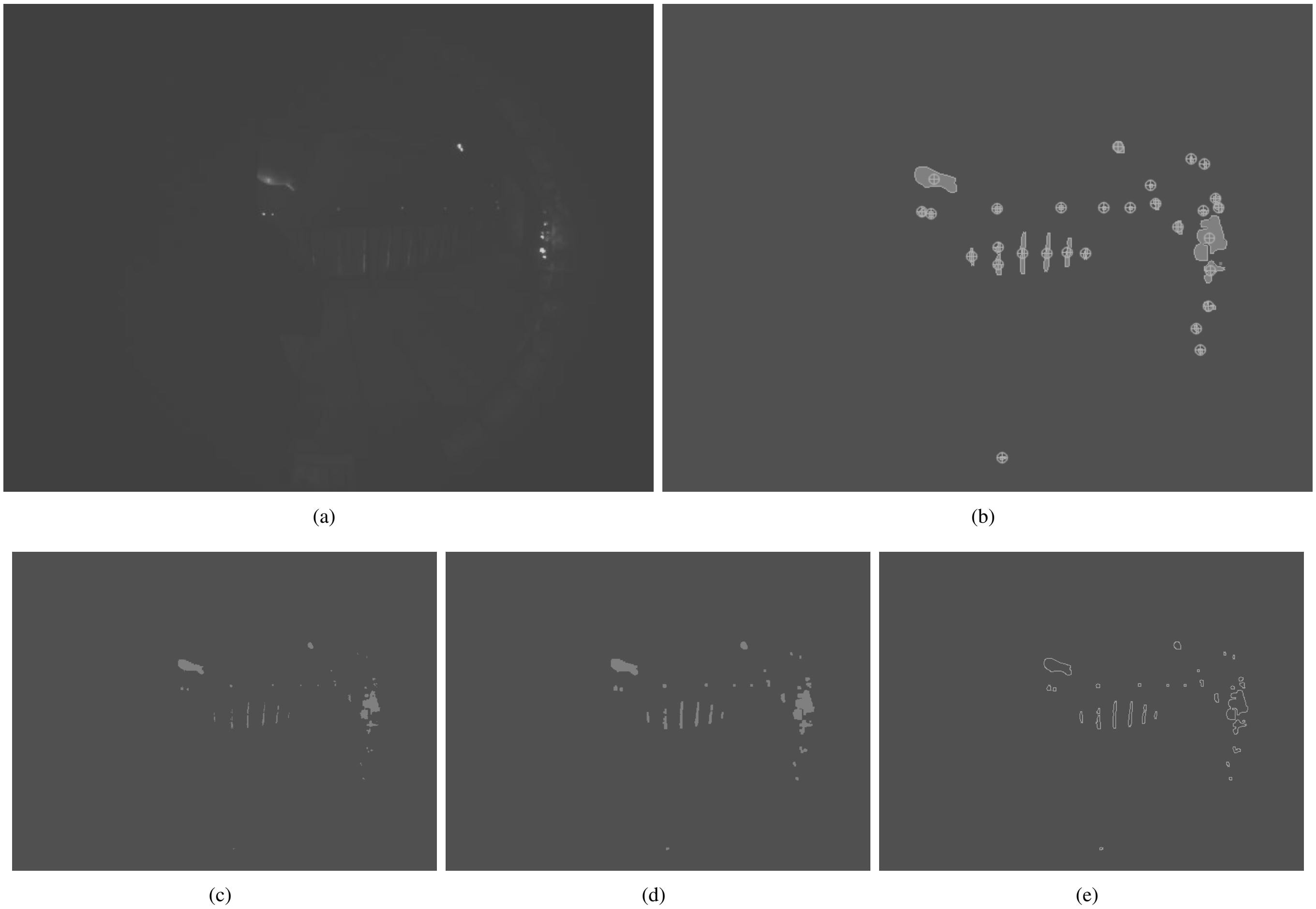
```
for all Landmarks $B_i{}^W$ in the map (in world Cartesian reference system) do
    $B_i^C = R_C * \mathbf{B}_m^W - C_W$
    $Proj(B^P{}_i) = (u_{B^P{}_i}, v_{B^P{}_i})$ applying Eqs. 1 and 2
end for
```

D. Landmark detection

The process for detecting landmarks consists of 5 phases: acquisition, preprocessing, segmentation, recognition and features extraction. The output of the system is an array of features for each landmark. In the preprocessing phase, the image (Fig. 8(a)) is transformed to facilitate the processing in the next stages. The techniques that have been used are binary thresholding (Fig. 8(c)) and morphological filtering (dilation) (Fig. 8(d)).

As segmentation techniques, the system uses a Canny filter and contour extraction (Fig. 8(e)). The next step is to extract the characteristics of each region:

- *Ratio*: number of pixels in the perimeter.
- *Centroid:* coordinates of the center of gravity.
- *Radio:* centroid distance to the center of the image.


(a)

(b)

(c)

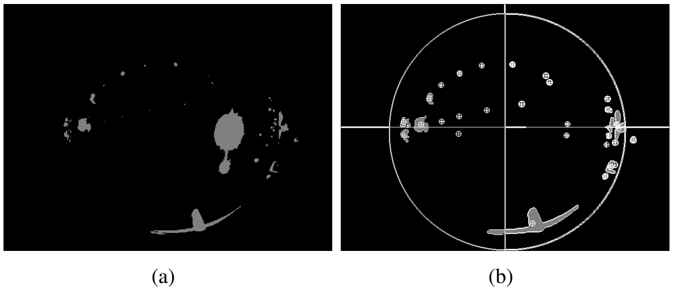
(d)

(e)

Fig. 8. Detecting landmarks in the omnidirectional filtered image: (a) original image, (b) binary thresholded, (c) dilated, (d) edges, (e) blobs and (f) saturated image.

- *Azimuth:* orientation of an object in the image with respect to the *x* axis (φ in Fig. 6).

If a light is pointing directly to the camera, then the acquired image will be saturated (Fig. 9(a)). In such cases, a big blob can be detected and the image has to be processed again using a higher threshold (Fig. 9(b)). This situation is very frequent in the region labelled as B (Fig. 2) because lights can be very close to the camera.


(a)

(b)

Fig. 9. Postprocessing phase: (a) saturated image and (b) postprocessed image.

IV. Global Localization

The localization algorithm for omnidirectional vision is based on a merit function that evaluates each proposed pose and an iterative process (based on a particle filter) for minimizing that function. The algorithm follows the same idea of a particle filter but it does not use any motion model. For each pose (particle) we build its ceiling map projection ($Map(P)$) using Alg. 1, and compare it with the detected landmarks in the image. The general scheme of the method is shown in Fig. 11.

A. Merit Function

Algorithm 2 Calculate the *Merit Function* (N_P and ε_P)

```
$Map(P)$
for all Beacons j in the image do
    for all Beacons i in Map(P) do
        $\varepsilon(B^P_{ij}) = |Proj(B^P_i) - Detected(B_j)|$
        if $\varepsilon(B^P_{ij}) < THRESHOLD$ then
            $\varepsilon_P = \varepsilon_P + \varepsilon(B^P_{ij})$
            $N_P = N_P + 1$
        else
            $\varepsilon_P = \varepsilon_P + THRESHOLD$
        end if
    end for
end for
```

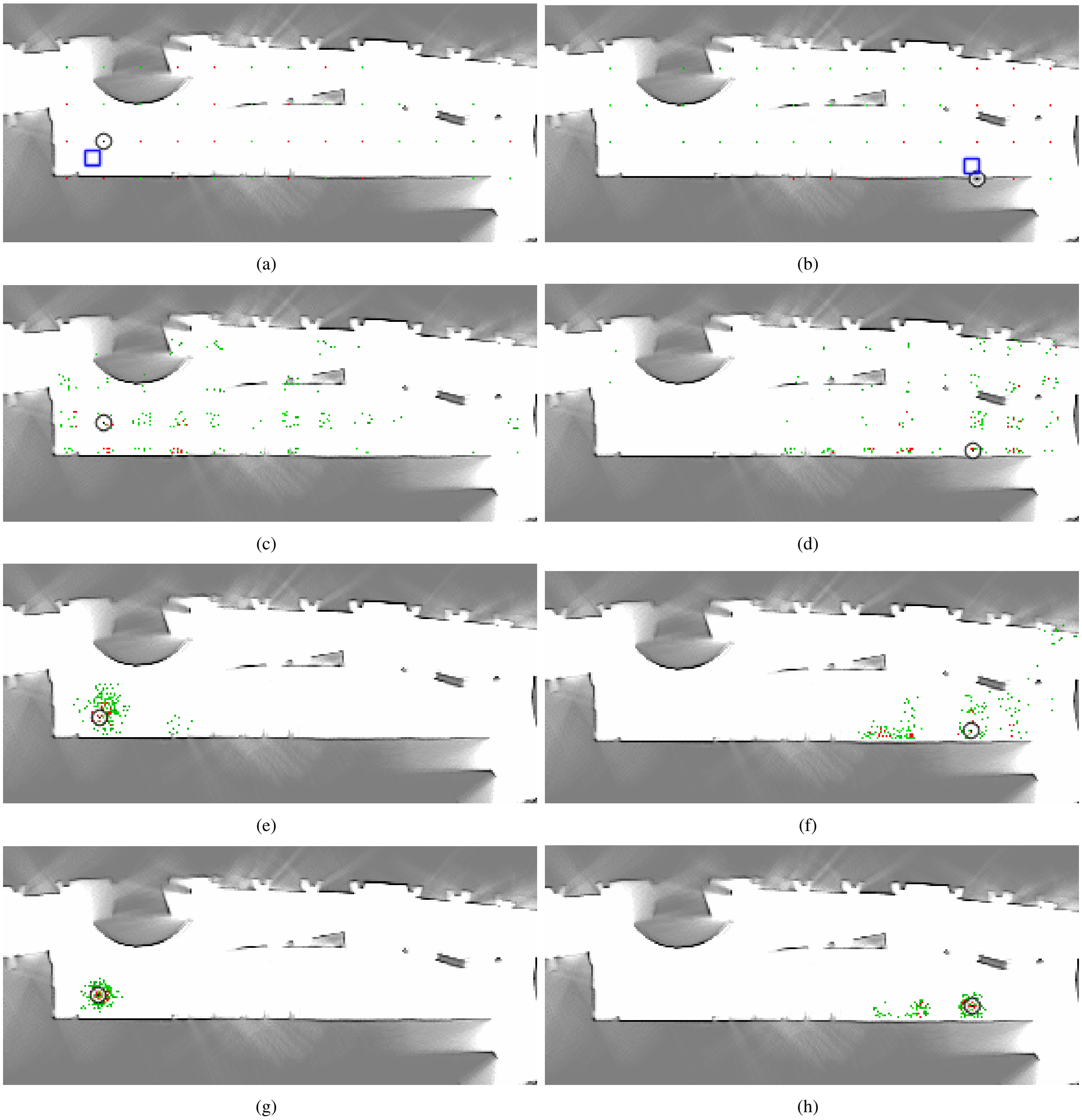


Fig. 10. Results for global localization in the Domus Museum for 2 different poses (left and right columns, respectively). (a,b) Top 200 poses in the initialization stage. Real poses are represented with a square and calculated poses (Eq. 6) with a circle. (c,d) First iteration of the resampling process. (e,f) Third Iteration. (g,h) Last iteration.

The *Merit Function* estimates the similarity between the image acquired with the camera and one ceiling map projection ($Map(P)$). It is defined as:

$$M(P) = \frac{1}{N_P} * \varepsilon_P \tag{5}$$

where N_P, is the number of landmarks that have been matched between the image and $Map(P)$, and ε_P is the accumulated error for these associations, i.e., the sum of the errors among each of the detected landmarks on the image and $Map(P)$ at one pose (see Fig. 12). Before the calculation of the *Merit Function*, a matching process has to be implemented. This matching process establishes the correspondence among the landmarks detected in the real image (B_j) and the theoretical pose of landmarks into the ceiling map projections ($Proj(B_i^P)$). It is based on minimum distance, and an example is show in Fig. 12. The conditions that have to be fullfilled for an association between B_j and $Proj(B_i^P)$, represented by B_{ij}, are:

1) $|B_j - Proj(B_i^P)| \leq THERSHOLD$
2) $|B_b - Proj(B_i^P)| \geq |B_j - Proj(B_i^P)|, \forall b \neq j$
3) $B_j \longleftrightarrow Proj(B_i^P)$ (uniqueness)

B. Minimization Process

The minimization process (Alg. 3) tries to find the pose that

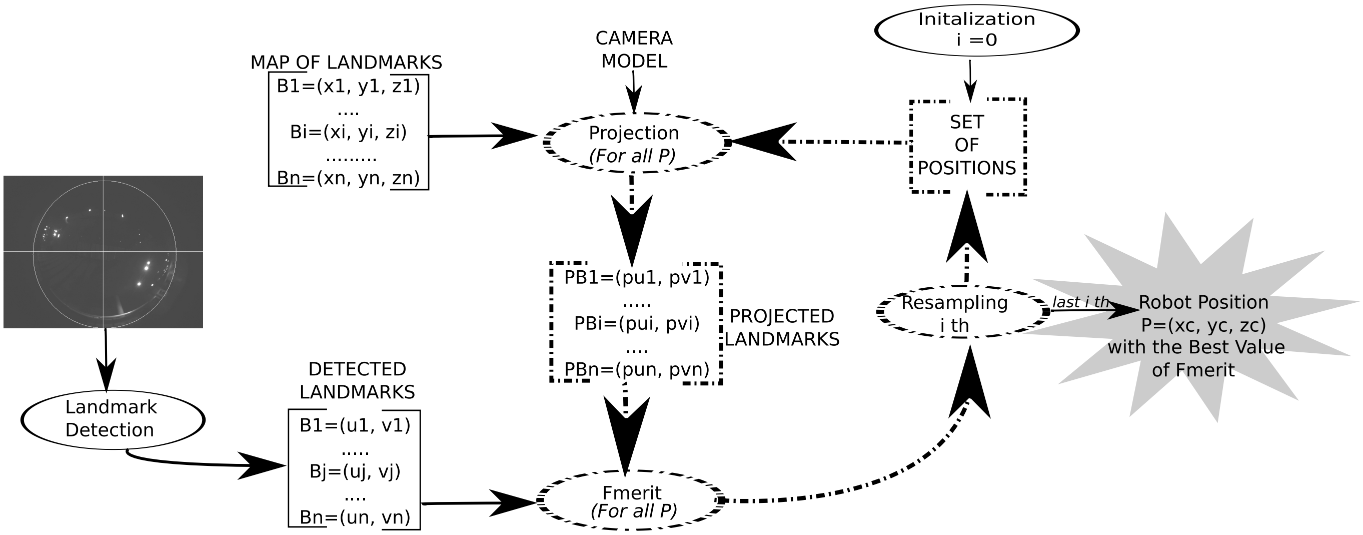


Fig. 11. General scheme of the global localization system for an omnivision image and a map of landmarks.

Algorithm 3 Minimization of the *Merit Function*.

```
Initialize set of particles $\zeta$
repeat
  for all P in $\zeta$ do
    Calculate $Map(P)$ (Alg. 1)
    Calculate $N_P$ and $\varepsilon_P$ (Alg. 2)
    Calculate M(P) (Eq. 5)
    Resample $\zeta$
  end for
until $Niter < 0$
```

produces the minimum value of the merit function. To explore all possible poses we use a particle filter, defining a particle as a pose in the environment. The algorithm has two stages, initialization and resampling:

1) Initialization: A set of particles (ζ) is uniformly distributed in the environment. Each particle represents a pose of the robot.

2) Resampling: This stage is executed iteratively until the particle set is stabilized. For each pose P, its merit function value ($M(P)$) is calculated applying Alg. 2. Then, the best particles of the set ζ are selected to pass to the next stage and the others are discarded. The lower the value of the *Merit Function*, the better the particle.

With this reduced set of particles, a subset of them is selected, random Gaussian noise is added to each of them, and the resulting particles are added to the particle set. In our experiment we have generated 5 new poses, from each pose which belongs to the 15% best particles of set ζ. From each of the next 25% best particles, a new particle is added in the same way.

Figure 10 shows the evolution of the set of particles for two images captured from different poses at the Domus Museum. In the last row (Figs. 10(i) and 10(j)) the poses with the best values of the merit function are shown. In Fig. 10(j) particles are grouped in two sets which means that there is a symmetry in the poses of the landmarks.

The pose of the camera is the one that gets the minimum value of the *Merit Function*. This happens when the number of identified landmarks (N_P) is the largest and the estimated error (ε_P) is the smallest:

$$C^W = argmin_P(M(P)) \tag{6}$$

V. EXPERIMENTAL VALIDATION

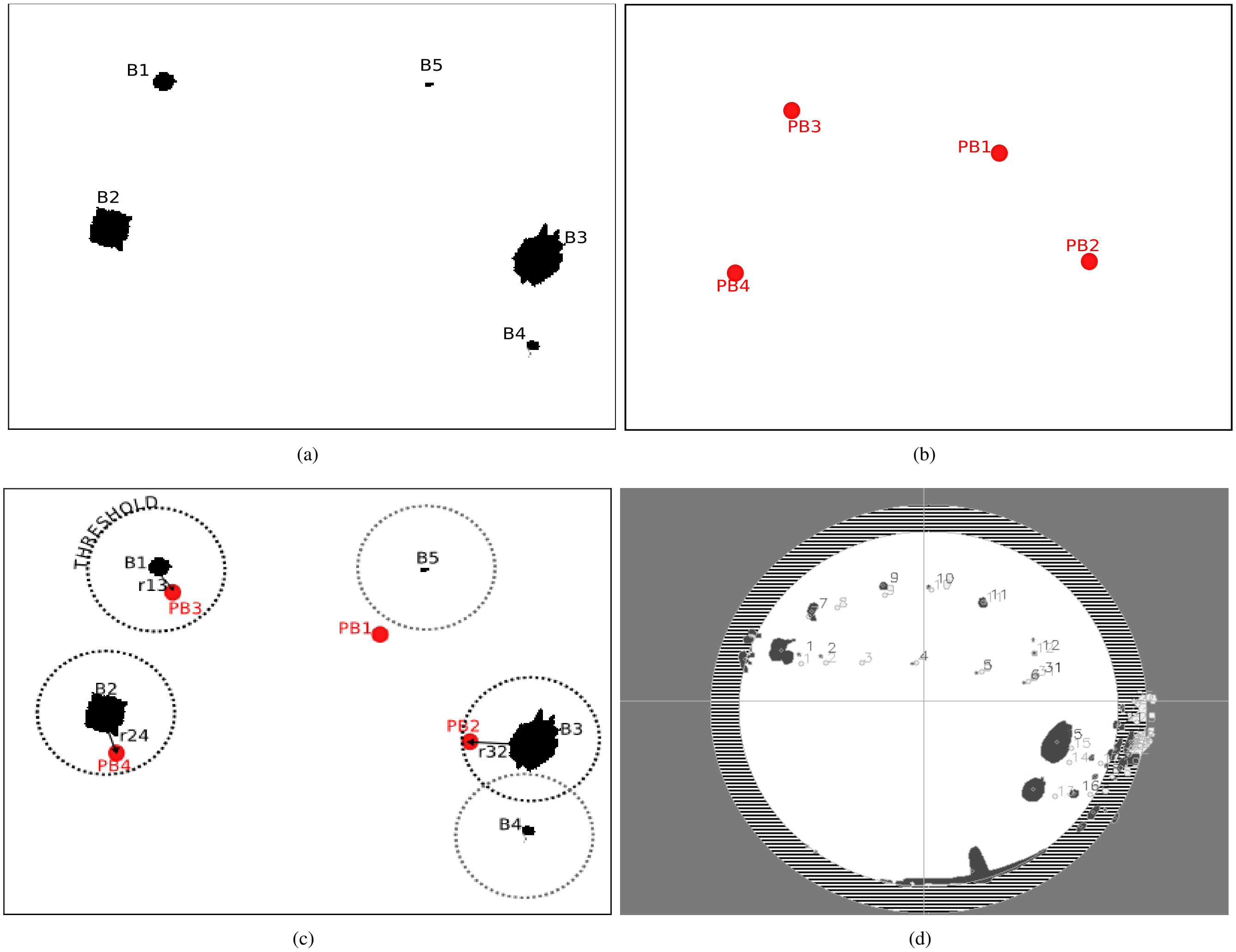
The localization system has been tested using data acquired from an omnidirectional camera mounted on a the Pioneer 2-AT robot in an exposition hall (Fig. 2) of the Domus Museum (A Coruña, Spain). The set of images were acquired each second and labeled with the corresponding pose determined with a localization system that uses laser range data. This position information has been used as the ground truth in order to measure the errors of the proposed localization algorithm. We are going to describe two experiments: the first one is a global localization task, and the second one is a kidnapping problem. The experiments were executed using 200 particles and *Niter* was set to 6.

A. Global localization

The experiment was built with a set of 60 images captured in a path of 24 meters long, and the robot moved along that path with a speed of up to 40 cm/s. Fig. 13(a) shows the estimated poses and the ground truth for the experiment.

The localization error (Fig. 14(a)) was calculated as the Euclidean distance between the pose of the robot given by the laser and the pose obtained by the proposed localization algorithm. The maximum error in the experiment was 2.42 m and the average error was 0.53 m.

The orientation error (Fig. 14(b)) was estimated from the absolute error in degrees between the orientation of the laser and the orientation obtained from the localization algorithm. Although the maximum achieved error was of 15 degrees, on average the error was of only 3 degrees.



(a) (b)

(c) (d)

Fig. 12. Matching between a real image and a map ($Map(P)$): (a) Real image. (b) Map. (c) Matching between detected landmarks (beacons) B_j on the image and the map projection $Proj(B^P{}_i)$ (labeled with PBi). (d) Projected map (detected landmarks are shown in black)

The time required for the computation of the algorithm for each image (Fig. 14(c)) depends on the total number of poses checked and the number of iterations. In the experiment, the total number of particles was 702 in the initial phase and 200 for the next iterations. This means 2502 calculations of the *Merit Function* for each image, requiring 300 ms on average on a Pentium 4 3.06 GHz.

Fig. 14(d) shows the number of matched landmarks (N_p) and the value of the Merit function ($M(P)$) for each pose. We have to highlight that the localization system only needs a reduced number of landmarks for an accurate pose estimation. Moreover, the error in localization does not depend on the number of matched landmarks.

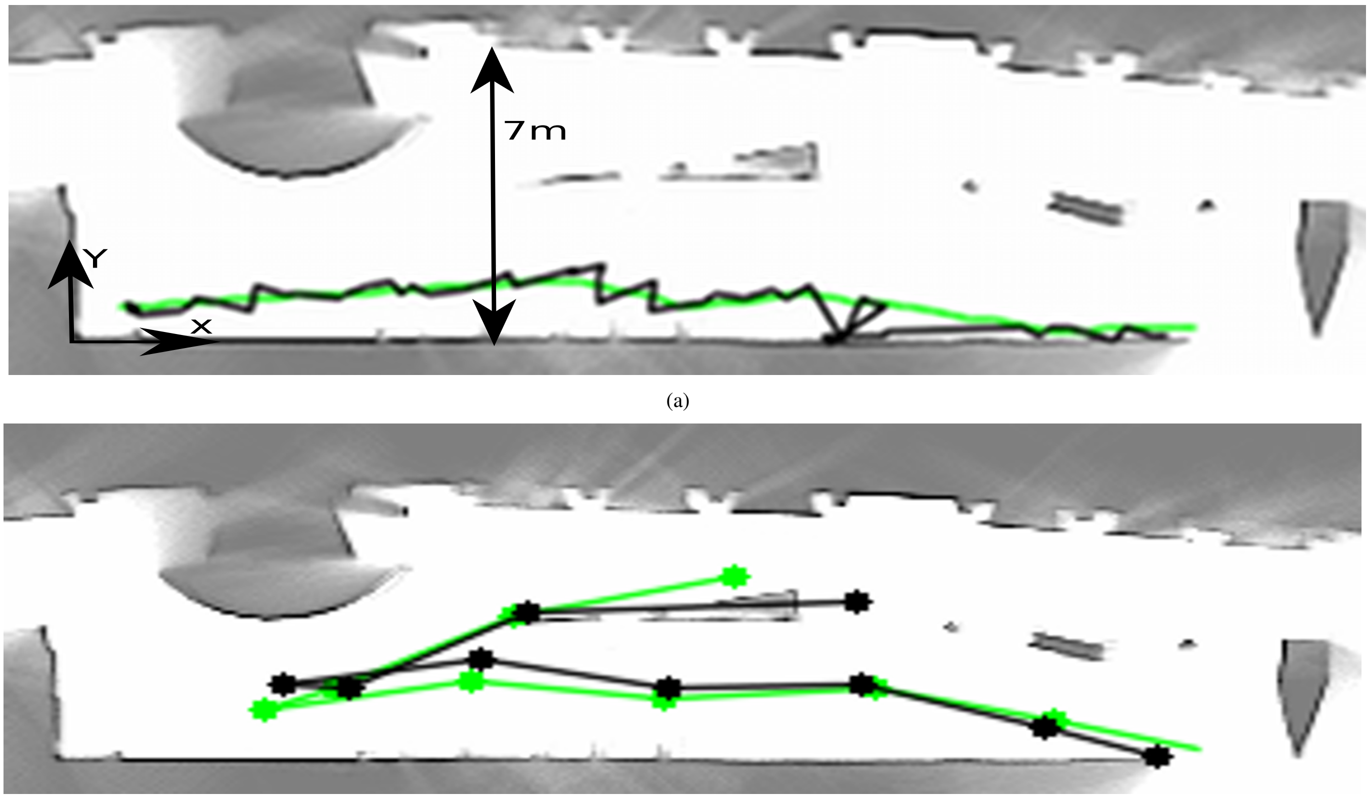
B. Kidnanapped robot

The second experiment was carried out to demonstrate that our system is be able to overcome kidnapping situations. We have used a sequence of images that represent jumps of about 400 cm between them in a path of more than 40 m. The estimated trajectories are shown in Fig. 13(b). In this experiments the maximum pose error was 2.59 m and 16 degrees (Fig 15).

VI. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a localization algorithm using omnivision. The algorithm is based on the minimization of a *Merit Function* through a particle filter. As the environment is too much crowded in normal conditions, the use of laser or sonar for localizing the robot is not possible. The use of an onmidirectional camera pointing to the ceiling solves this problems. However, irregularities on the floor produce swing movements in the camera, enhancing the difficulties. Nevertheless, and taking into account the experimental results, we can conclude that our system can solve the global localization problem and the kidnapping problem with an average pose error of 0.6 m and 3 degrees.

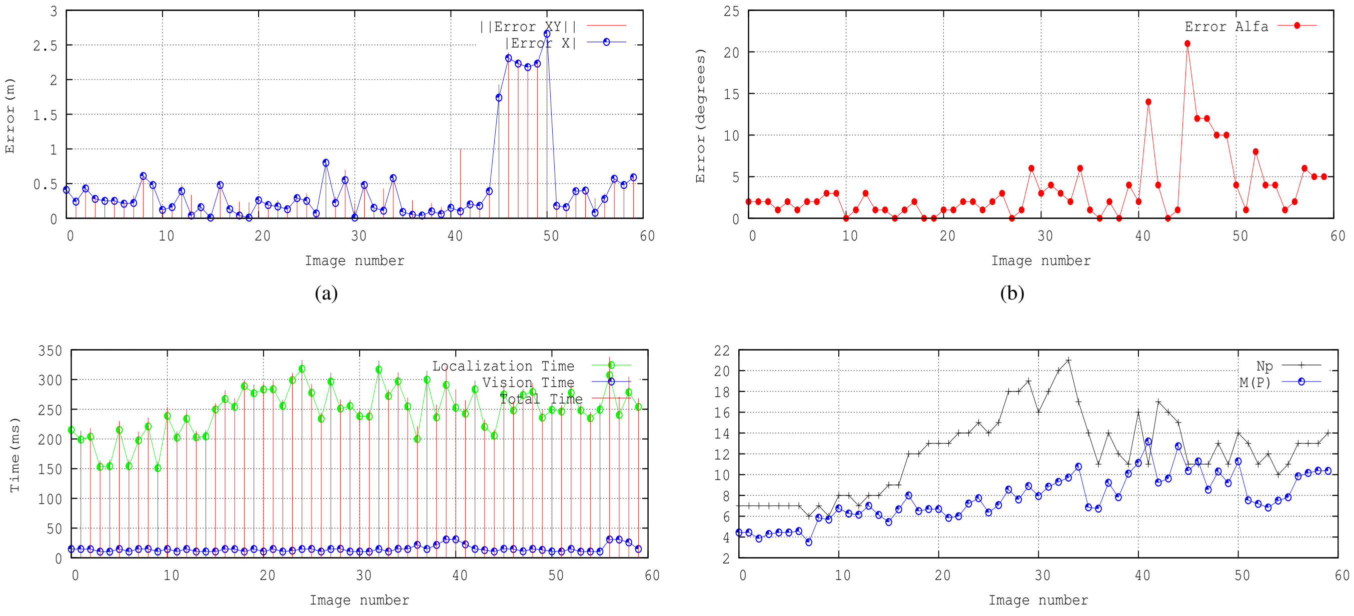
The most important problem of the system is the symmetry problem. It can produce errors because there are some poses in the environment from which the view of the landmarks configurations is similar, misleading the system. Moreover, the processing time for each image is quite low and, therefore, the system can be executed on real time. In summary, the system can localize the robot in a robust and accurate manner.



(a)

(b)

Fig. 13. Path of the robot plotted on the grid map created with laser data. Omnivision localization is shown in black and the ground truth (laser) in green. (a) Global localization experiment. (b) Kidnapping experiment.



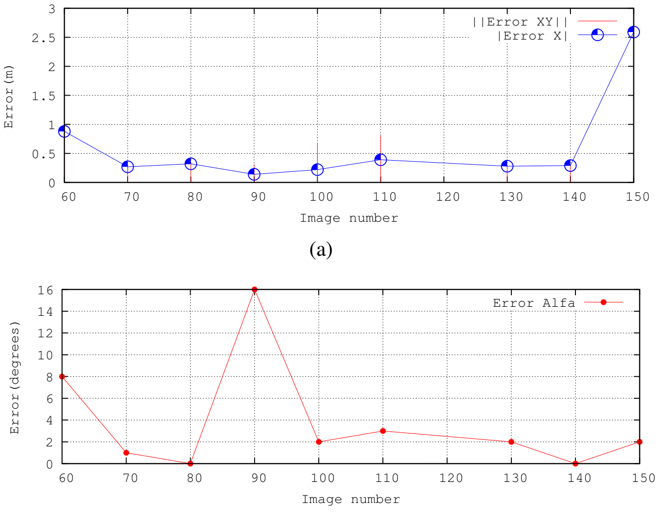
(a)

(b)

(c)

(d)

Fig. 14. Errors during global localization: (a) Pose error. (c) Orientation error. (d) Processing time. (e) Number of landmarks and *Merit Function* value.



(a)

(b)

Fig. 15. Errors during kidnapping experiments: (a) Pose error. (b) Orientation error.

As future work, we are planning to reduce the localization error adding the robot action model used in the probabilistic algorithms ([15], [6]). Also, the fusion of information from other sensors like laser or odometry can help in this error reduction. The final objective is to produce a SLAM [8] system for omnivision that allows the localization in any environment without the restriction of having a previous map.

ACKNOWLEDGMENTS

Authors want to thank Tomas Pajdla and Hynek Bakstein for their help in the calibration of the camera. This work has been partially supported by the Galician Regional Government and the Spanish Government through grants PGIDIT-TIC206011PR, TIN2005-03844, INCITE08PXIB262202PR and TIN2008-04008/TSI.

REFERENCES

[1] H. Andreasson, A. Treptow, and T. Duckett. Localization for mobile robots using panoramic vision, local features and particle filter. pages 3348–3353, 18-22 April 2005.

[2] H. Bakstein and T. Pajdla. Panoramic mosaicing with a 180/spl deg/ field of view lens. In *Omnidirectional Vision, 2002. Proceedings. Third Workshop on*, pages 60–67, 2002.

[3] Eric Brassart, Laurent Delahoche, Cyril Cauchois, Cyril Drocourt, Claude Pegard, and El Mustapha Mouaddib. Experimental results got with the omnidirectional vision sensor: Syclop. *omnivis*, 00:145, 2000.

[4] Zuo L. Cao, Sung J. Oh, and E. L. Hall. Dynamic omnidirectional vision for mobile robots. In *Journal of Robotic Systems*, 1986.

[5] D. Chekhlov, M. Pupilli, W. Mayol, and A. Calway. Robust real-time visual slam using scale prediction and exemplar based feature description. *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–7, 17-22 June 2007.

[6] H. Choset, Lynch, M. Kevin, Seth Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementation*. MIT Press, 2005.

[7] A. Clerentin, L. Delahoche, E. Brassart, and C. Drocourt. Localisations absolues en milieu interieur bases sur la vision omnidirectionnelle. *TS. Traitement du signal*, 20:183–212, 2003. French.

[8] A. J. Davison, Y. Gonzalez, and N. Kita. Real-time 3d slam with wide-angle vision. In *IAV 2004*, July 2004.

[9] D. Fox, S. Thrun, W. Burgard, F. Dellaert, and C. Fernańdez. Particle filters for mobile robot localization. In *Sequential Monte Carlo Methods in Practice*, New York, 2001. Springer.

[10] E. Menegatti, A. Pretto, A. Scarpa, and E. Pagello. Omnidirectional vision scan matching for robot localization in dynamic environments. In *IEEE transactions on robotics*, volume 22, June 2006.

[11] A. C. Murillo, C. Sagüés, J. J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool. Hierarchical localization by matching vertical lines in omnidirectional images. In *From Sensors to Human Spatial Concepts workshop, held together with IROS 06*, pages 13–19, 2006.

[12] Jose Pascual Roda, Juan Manuel Saez, and Francisco Escolano. Ceiling mosaics through information-based slam. *Intelligent Robots and Systems, IEEE/RSJ International Conference on*, pages 3898–3904, Oct. 29 2007-Nov. 2 2007.

[13] Y. Sun, Q. Cao, and W. Chen. An object tracking and global localization method using omnidirectional vision system. In *5th World Congress on Intelligent Control and Automation*, Hangzhou. P.R. China, 2004.

[14] H. Tamini, H. Andreasson, A. Treptow, T. Duckett, and A. Zell. Localization of mobile robots with omnidirectional vision using particle filter and iterative sift. In *European Conference on Mobile Robots (ECMR05)*, Ancona, Italy, September 2005.

[15] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2003.

[16] Sebastian Thrun, Michael Beetz, Maren Bennewitz, Armin Cremers, Frank Dellaert, Dieter Fox, Dirk Hähnel, Charles Rosenberg, Nicholas Roy, Jamieson Schulte, and Dirk Schulz. Probabilistic algorithms and the interactive museum tour-guide robot Minerva. *International Journal of Robotics Research*, 2000.