Global localization based on omnivision sensor for a guide mobile robot

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

***Abstract*—This paper describes the solution adopted to localize a guide robot at the Domus Museum based on omnidirectional vision and a known map of the beacons in the environment. We propose a merit function that ranked different proposed position for each acquired image and an iterative process based on particle filter for minimizing this function. Finally, experiments in the Domus Museum are shown which demonstrate that our system localizes a mobile robot in a very complex and crowed environment with accuracy and robustness and it can be executed in real time.**

***Index Terms*—Global localization, omnivision, robot guide.**

I. Introduction

ROBOT localization is one of the most important problems in autonomous mobile robotics. Determining the location of a mobile robot is finding the Cartesian coordinates and angular orientation relative to an external frame. It requires to be reliable, robust and executable in real time.

There are several possibilities to find out a solution using different types of sensors: laser, ultrasonic, or infrared sensors and vision. Nowadays vision sensors are preferred to the other ones, because for a low cost they can reflect accurately more details of the environment and they can run and be processed in real time due to the improvements made in computers in the last years.

Our work consists of a positioning system that uses artificial vision to estimate the position of the camera (i.e. robot Fig. 1) from a map of beacons. The camera is pointing to the ceiling and elevated 1.5 m above the robot (1.8 m above the ground), so that its movements are restricted to the xy plane and the noise or occlusion generated by moving people is minimized.

We use the omnidirectional camera shown in Fig. 2. It provides a very wide field of vision (FOV of about 185^o) which covers half the space of the environment and so it can obtain a lot of information about it in one acquisition.

A beacon can be any distinctive and recognizable object on the environment. This work uses the own environment lights. These are easily detectable, repetitive and usually visible along large trajectories. On the other hand, any building has lights (Fig. 3), so there is no need for prior preparation of the environment in order to use our localization method. The main problem is their individual identification, because they usually are identical.

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

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

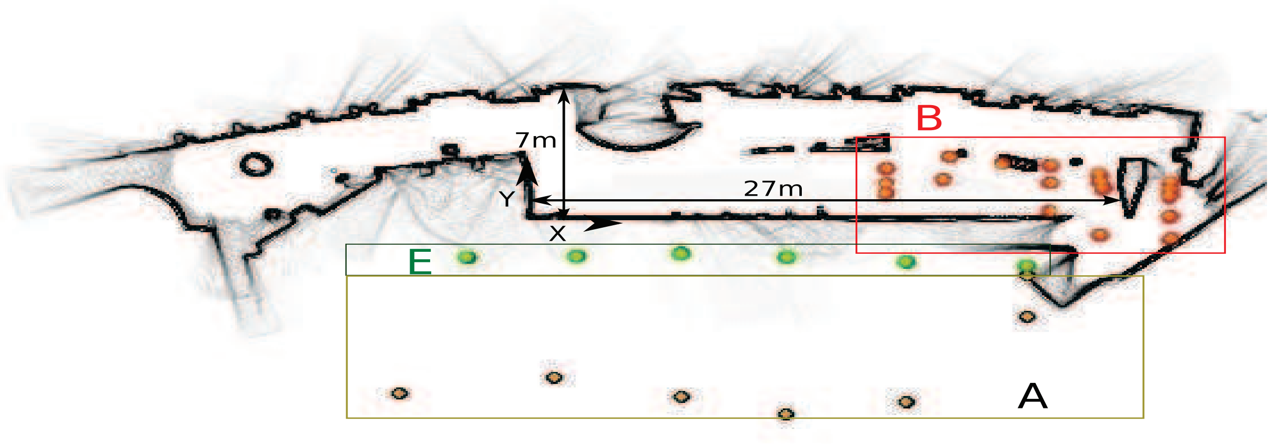
Fig. 1. The vision system in the Guide Mobile Robot, based on Pioneer 2AT, used at the Domus Museum in A Coruña (Spain). The omnivision camera is marked with a circle.

(a)

(b)

Fig. 2. (a) Omnidirectional lens (185^o FOV). (b) Omnidirectional image acquired with our omnidirectional camera.

Section II expounds an overview of related work. The next two sections describe deeply our omnivision and localization system. Section V presents the experimental results in a real environment. And finally, last section is devoted to conclusions and future work.



(a)

(b)

(c)

Fig. 3. Beacons used to localize the guide robot in the Domus Museum: (a) 2D environment map with glares (circles), the position of each glare can be seen in table I; (b) Some glares labeled with E and A in the map; (c) Glares labeled with A in the map. The position of each beacon are shown in Table I

II. RELATED WORK

The first work that considered omnidirectional vision to locate a mobile robot was published in 1986 by Cao et at [3]. But 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 possible omnidirectional vision configurations: *catadioptric* as [2], [5], [10], [9] (where the camera images are obtained through a conic mirror) and *dioptric* (where images are captured through a lens).

Some works as [5], [11], [8] are limited to the routes that the agent has previously learnt. These approaches create a database with images of every route and their positions, and the robot can be localized by correlation between the captured images and the database images in real time. These systems have the drawback that they can not work in other routes on the environment.

Other implementations use landmarks (beacons) of the environment to get the position of the robot. For instance, in [10] the goals of a field of RoboCup are used as marks and [2], [12] is 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 beacons problem and our process to discover beacons is simple, fast and efficient. A similar approach to using an omnivision camera oriented to the ceiling is used on [9], but it is based on an information theory to get the global trajectory.

III. PROJECTIONS IN AN OMNIVISION SYSTEM

A. Camera Model

The camera model describes how a 3D scene is transformed into a 2D image (Fig. 5). The standard model is the *Pin-Hole*, which projects the scene on a flat retina (Fig. 4), but it is limited to cameras with $FOV << 180^0$.

The model that best fits our system was developed by Pajdla and Bakstein [1] based on a spherical retina (Fig. 4) where the image is formed on a curved surface. In our case the radial symmetric function is:

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

TABLE I
POSITION IN METRES OF THE BEACONS IN THE ENVIRONMENT LABELED IN FIG. 3(A).

LABEL	X	Y	Z
Glares 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
Glares 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
Glares 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

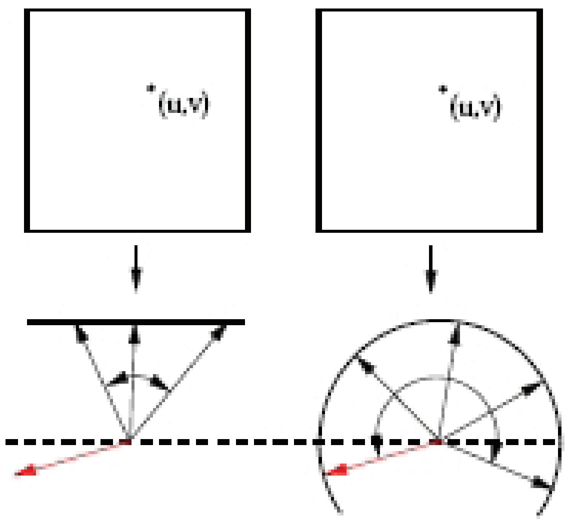


Fig. 4. The *Pin-Hole* camera model based on a flat retina (left) compared with the omnidirectional camera model based on spherical retina (right).

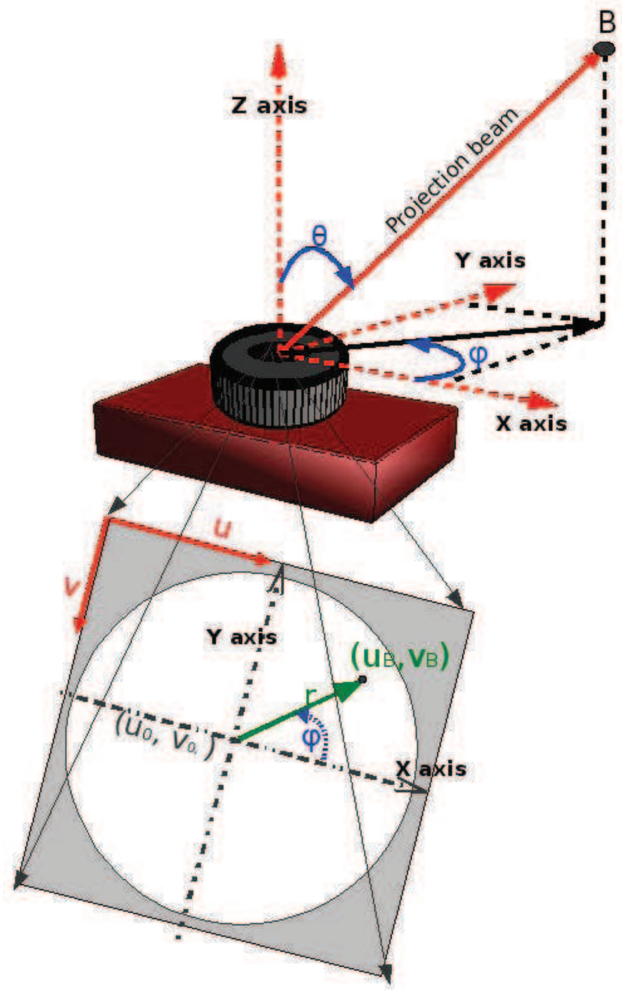


Fig. 5. Theoretic omnidirectional camera model and projection of a point B. Its projection ray is defined by the elevation (θ) and the azimuth (φ) with respect to the camera coordinate system. r y φ are the polar coordinates of the projected point (u_B, v_B). (u_0, v_0) are the coordinates of the image center.

where *a, b, c,* and *d* are the adjustment parameters of the model, r is the distance in the image between the projection point of B ((u_B, v_B)) and the image center ((u_0, v_0)), and θ is the elevation of B regard to the optical axis of camera (see Fig. 5).

This function makes it possible to calculate the coordinates of the image *(u,v)* depending on the azimuth (φ) and the elevation (θ) (Fig. 5):

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

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

B. A Beacon Projection

If we have the coordinates of beacon i ($\mathbf{B}_i^W$) and the coordinates of the camera $\mathbf{C}^W$, both in respect to the environment reference system (W), we can calculate the projection line of the beacon $\mathbf{B}_i^C$ (B in 5) relative to the camera (C).

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

Algorithm 1 Calculate $Map(P)$ for one position P.

for all Beacons ${B_i}^W$ on the map (in world cartesian reference system) **do**
  $\mathbf{B}_i^C = R_C * \mathbf{B}_m^W - \mathbf{C}_W$
  $Proj({B^P}_i) = (u_{{B^P}_i}, v_{{B^P}_i})$ applying 1 and 2
end for

where R_C is the rotation matrix of $\mathbf{C}_W$ relative to W.

The traditional Euclidean transformations apply to obtain the elevation (θ) and the azimuth (φ) angles from $\mathbf{B}_i^C$.

To get the projection of the beacons $\mathbf{B}_i^W$, $Proj({B^C}_i)$, as image coordinates ($u_B and v_B$ in Fig. 5), we apply the Eq. 1 and 2:

$$Proj({B^C}_i) = (u_{{B^C}_i}, v_{{B^C}_i}) \tag{4}$$

C. Ceiling Map Projection

We have named ceiling map projection, $Map(P)$, to the set of theoretical positions that each beacon in the environment (${B^W}_i$) would have in the image (u, v), the pixel where it would be (Sec. III-A), if the robot was at the position P. The algorithm used is detailed in Alg. 1. A graphical example is showed in Fig. 7.

IV. Global Localization

To localize a guide robot at the Domus Museum based on omnidirectional vision and a known map of the beacons in the environment (Fig. 3) we propose a merit function that evaluates each proposed position and an iterative process (based on a particle filter) for minimizing that function. Our process follows the same idea as particle filter but it does not use any motion model. For each position (particle) we build its ceiling map projection, $Map(P)$ using Alg. 1, and compare it with the detected beacons in the image. The general scheme of our method can be seen on Fig. 6. The image processing was explained in a previous work [7]. The other parts of the system will be described here, but first we are going to explain the merit function and then the minimization process.

A. Merit Function

The *Merit Function* is presented in Eq. 5. It estimates the similarity between the image acquired with the camera and one ceiling map projection $Map(P)$ (artificial image built for one position based on the theoretical model, see Sec. III-C).

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

where N_P, is the number of matchings between beacons detected in the image and $Map(P)$, and ε_P is the accumulated error for this associations, the sum of the errors between detected landmarks (beacons) on the image and $Map(P)$ at one position (see Fig. 7). This calculation is detailed in Alg. 2. A graphical example and the matching process is illustrated in Fig. 8

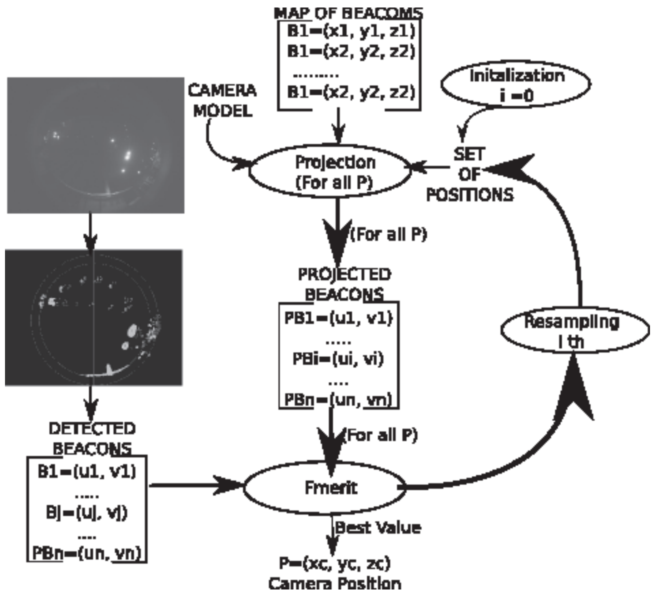


Fig. 6. General scheme of our global localization system from an omnivision image and the map of beacons.

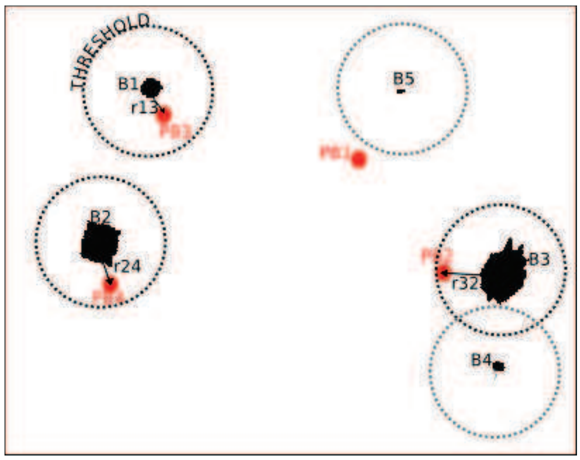


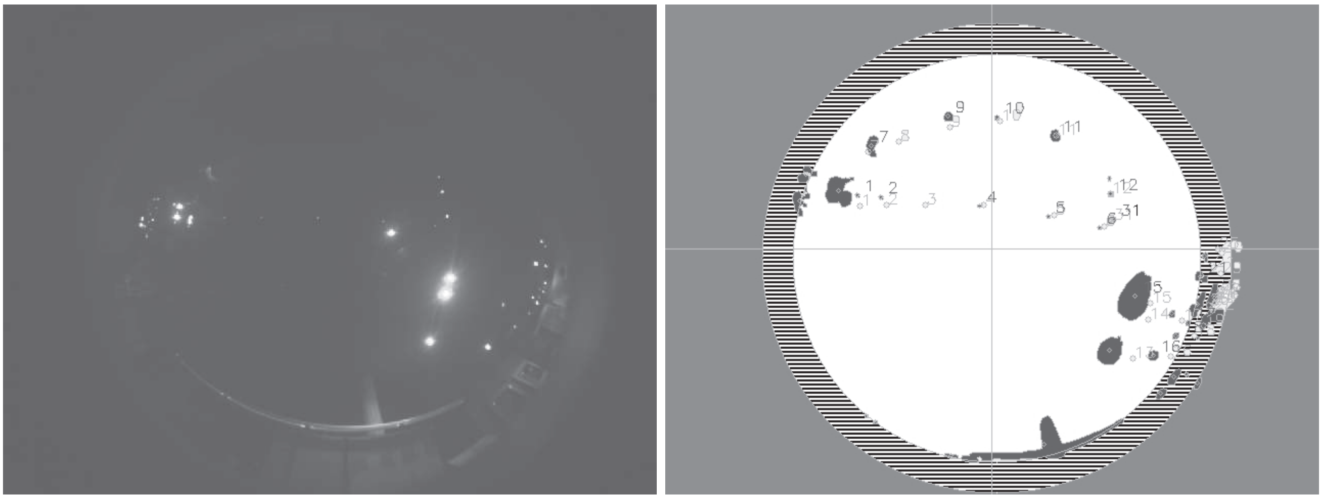
Fig. 8. Matching between detected landmarks (beacons) B_j on the image and the ceiling map projection $Proj({B^P}_i)$ (labeled with PBi). $N(P) = 3$ and $\varepsilon_P = r13 + r24 + r32 + THRESHOLD$.

B. Minimizing Process

The minimizing process consists of searching a position in the environment that has the minimum value of the merit function. To explore all possible positions we use a particle filter, defining a particle as a position in the environment. The filtering mechanism has two stages for each position (it is explained in detail in Alg. 3): initialized sample and resample.

1) Initialized sample: At the start we generate a set of particles ζ uniformly distributed around the whole environment. In our experiment represented in Fig. 9 they are distributed every 2 metres and 10 degrees.

For each position P, we estimate its merit function value $M(P)$ applying Alg. 2 to compute the number of 'matched' beacons (N_P) and its 'quality' (ε_P). The best particles of the set ζ are selected to pass to the next stage and the others are discarded. In our experiments (Fig. 9) we selected the particles



(a) (b)

Fig. 7. Comparison of the projection Map of Ceiling, $Map(P)$, with detected beacons: (a) Original image; (b) Graphical representation of the projection map (gray enumeration) in the processed image with beacons detected (black enumeration). All beacons detected and projected in shaded region are in the horizont and they will be discared to calculate the $M(P)$.

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

Algorithm 3 Minimization of the Merit Function.

Initialized set of particles ζ
repeat
  for all P in ζ **do**
    Calculate $Map(P)$ (Alg. 1)
    Calculate N_P and ε_P (Alg. 2)
    Calculate M(P) (Eq. 5)
    Resample ζ
  end for
until $Niter < 0$

that have the best value of the merit function.

2) Resample: The stage of resample is a routine that generate new particles adding noise to the positions in ζ according to a Gaussian distribution. This stage is executed iteratively until our set is stabilized.

In our experiment we generate 5 new positions, from each position which belongs to the 15% best of the set ζ adding noise generated randomly in the range $[-0.50m, 0.50m]$ and $[-5^o, 5^o]$. From the best 25% of the rest of positions another one position is generated. We have tested another resample models, but there was not significative differences in the final results.

Only the top positions of ζ according to $M(P)$ are selected to execute the next iteration.

Figure 9 shows the evolution of the set of particles for the images captured from two different positions at the Domus Museum. In the last row of images in Fig. 9(i) and 9(j) the positions with best values of the merit function can be seen for each position. Noted that in Fig.9(j) the particles are grouped in two sets which means that there is a symmetry in the positions of the beacons and then in the merit function $M(P)$. The second row shows the initialized stage of the filtering process, only the positions with the best values of $M(P)$ are selected.

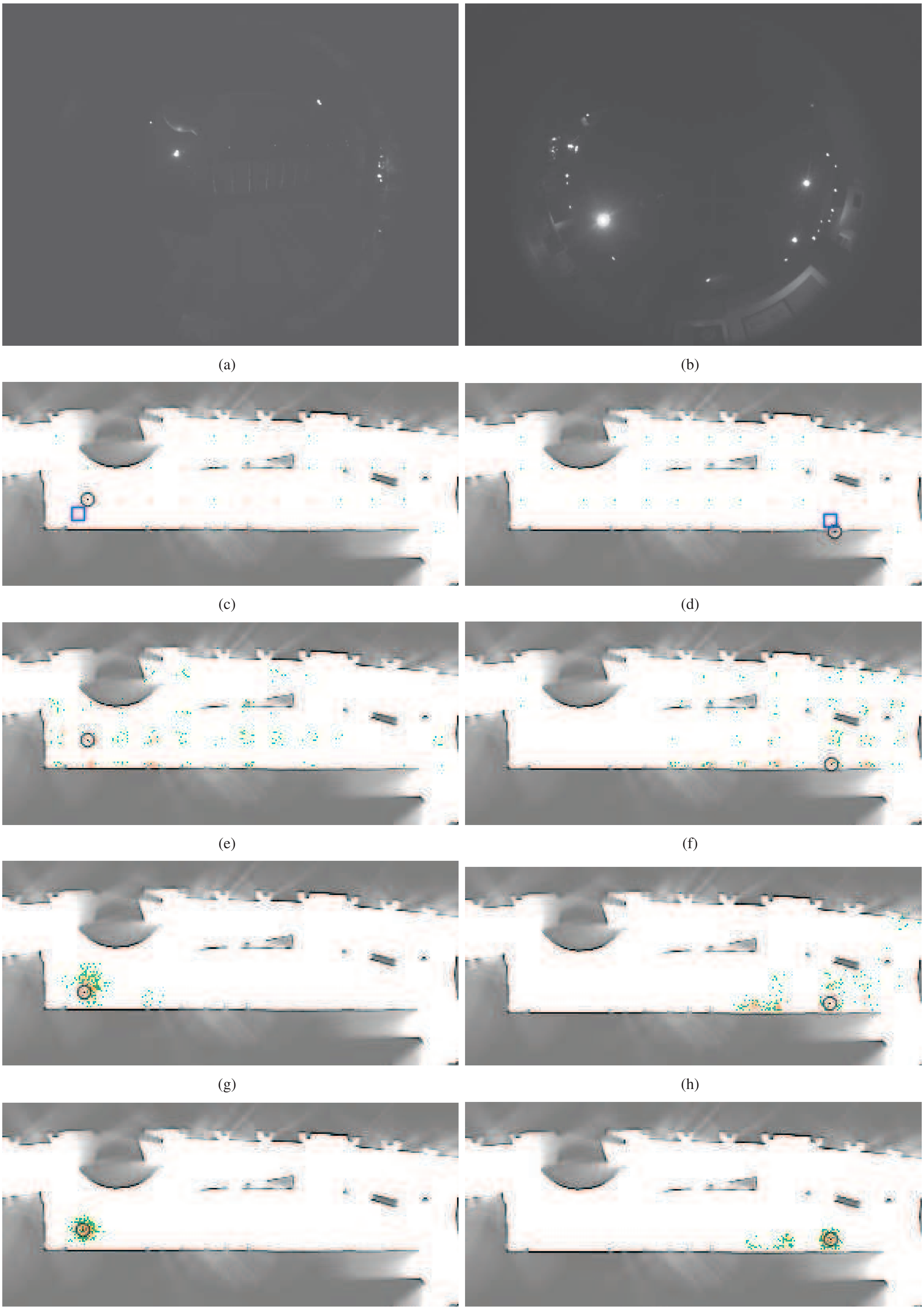
The 3D position of the camera is those that reach the minimum value according to our merit function. It happens when the number of beacons identified N_P is the largest and the error ε_P estimated is smallest:

$$C^W = \widehat{P} \setminus M(\widehat{P}) = \min_{P \in \zeta}(M(P)) \tag{6}$$

V. EXPERIMENTAL VALIDATION

The experiments were carried out at the Domus Museum in A Coruña (Spain) in a section of about 27m x 7m (Fig. 3(a)). The experiments were performed off-line on different sequences of images acquired in the museum. All images were labeled with the corresponding laser position, and 1 image per second was acquired on average.

The experiments have a length of 60 images, which corresponds to a path of about 24 meters long. Between all images the distance is about 0.40 metres on average. The trajectory travelled in the map taking into account the positions of laser sensor and the positions computed in our system is showed in Fig. 10(a).



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 9. Results for global localization in the Domus Museum for 2 different positions (left and rights columns, respectively), positions 8 and 50 in Fig 10. (a,b) Captured omnivision images. (c,d) Top 200 positions in the initialization stage. *Real* positions are marked with a square and calculated positions (Eq. 6) are marked with a circle. (e,f) First iteration of the resampling process. (g,h) Third Iteration. (i,j) Last iteration.

After several experiments, the best results (precision and processing time) were achieved using 6 iterations with 200 particles.

Figure 10(b) shows the localization error. It was calculated as the Euclidean distance between the position of the robot given by the laser and the position obtained through the minimization of our merit function. The maximum error is 2.42 metres and the average error is 0.53 metres.

The error in orientation (Fig. 10(c)) was estimated from the absolute error in degrees between the orientation of the laser and the orientation obtained by using our merit function. Although the maximum error achieved 15 degrees, on average the error is only 3 degrees.

The time required for the computation of the algorithm (Fig. 10(d)) depends on the total number of positions checked and the number of iterations. In our experiment, the total number of particles checked in our system is 702 in the initial phase and 200 for each of the other 6 iterations, which makes a total of 2502 particles checked. In an desktop computer Intel Pentium 4 CPU 3.06GHz the full process for this particles require an average of about 300 ms. Note that the time for processing each image (vision time) its insignificant in our system. Therefore, our localization system can be executed in real time.

VI. CONCLUSIONS AND FUTURE WORK

The solution adopted in this study to locate a guide robot at the Domus museum is based on searching the ceiling map projections from the images acquired each time that the robot needs to know its position. Our omnivision system is composed of an IR filter that reduces the image process to find the lights. The localization system is based on the search of the position that minimizes the merit function. The only previous information needed is a map of the glares in the environment.

It is noticeable that the Domus museum is a crowded environment, so that another distance sensor like laser or ultrasonic could not work. The management of the museum imposes the restriction of not to modify the environment because the image and design of the exhibition halls can not be broken. Indicate also that the environment has an irregular floor, it produces swinging in the camera support that may increase the error.

The most important problem of the system is the symmetry problem. It can produce error because there are some positions in the environment from which the views of the ceiling are similar, this can mislead the system.

We want to highlight that our system was designed to cope with occlusions, since our beacons are located in the ceiling and only other objects in the environment can caused the problem. Moreover the very wide visual field guarantees that enough beacons will be viewed from our sensor to estimate a good localization. In our experiments we considered only 31 glares and we can localize the robot in a space of $216m^2$.

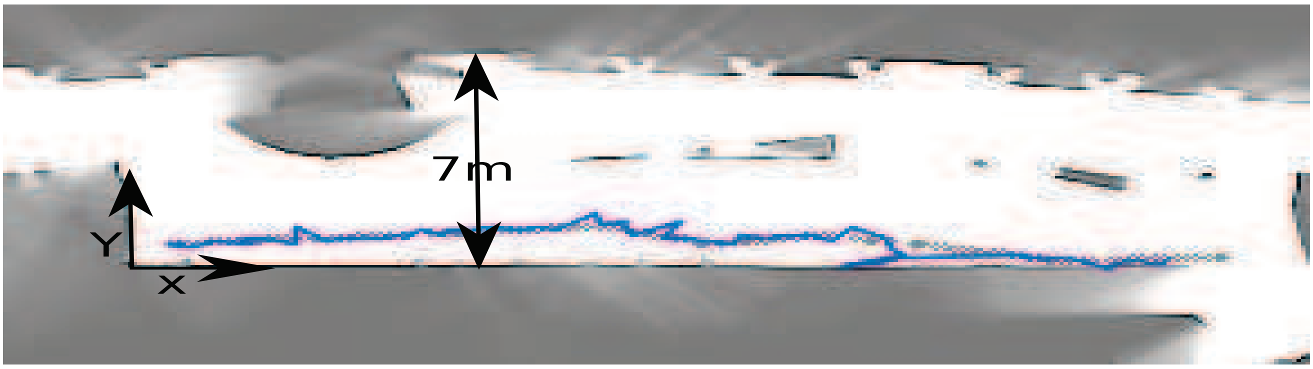
In spite of the mean of around 0.6 metres we conclud from the results of our experiments that the system can localize the robot in a robust and accurate manner. Besides, this error could be reduced adding the robot action model used in the probabilistic algorithms ([13], [4]) the Extended Kalman Filter, Monte Carlo methods or Bayesian filtering and the fusion of information from other sensors like laser or odometry. The implementation of these models in our guide robot will be made in future work. The final objective of our work is to produce a SLAM ([6]) system for omnivision that allows the localization in any environment without the restriction of having a previous map.

ACKNOWLEDGMENTS

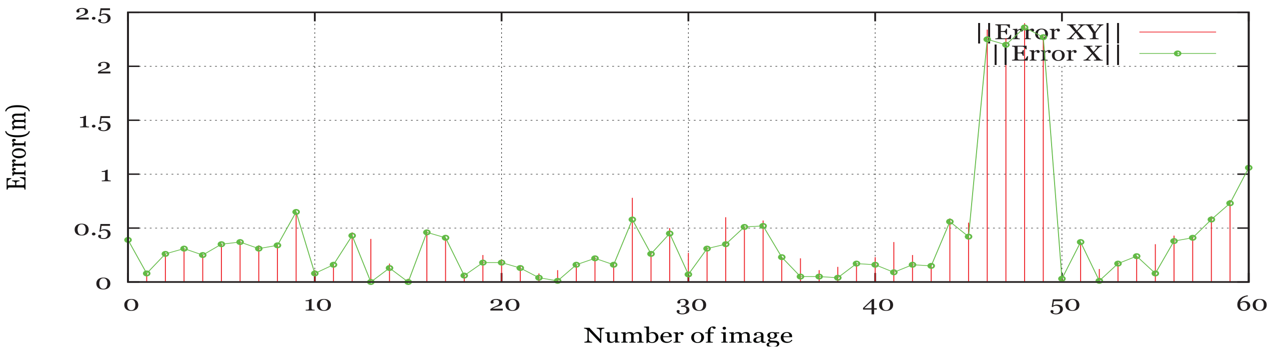
We want to give special thanks for the collaboration of Tomas Pajdla and Hynek Bakstein in the calibration of the camera. This work has been financed by the Galician regional government and the Spanish Government through grants PGIDIT-TIC206011PR and TIN2005-03844.

REFERENCES

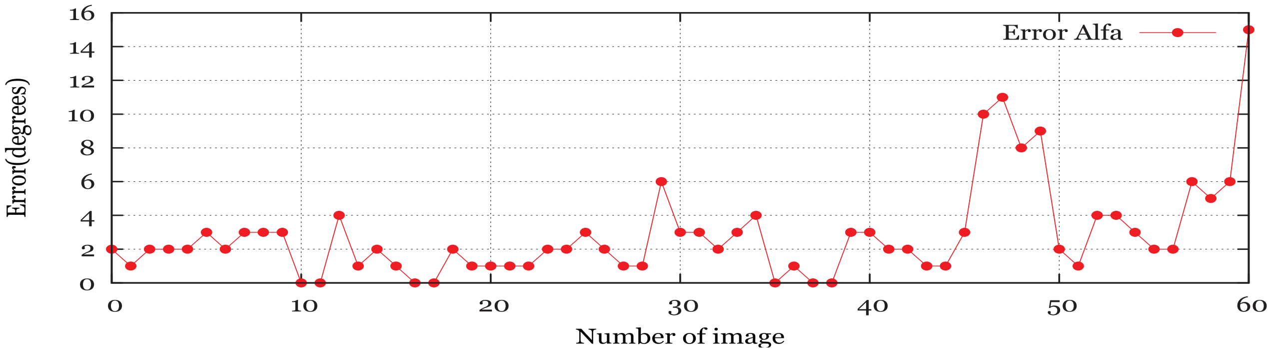
[1] H. Bakstein and T. Pajdla. Panoramic mosaicing with a 180/spl deg/ field of view lens. In *Omnidirectional Vision, 2002. Proceedings. Third Workshop on*, pages 60–67, 2002.
[2] E. Brassart, L. Delahoche, D. Cauchois, C. Drocourt, Claude Pegard, and E. M. Mouaddib. Experimental results got with the omnidirectional vision sensor:syclop. *omnivis*, 00:145, 2000.
[3] Zuo L. Cao, Sung J. Oh, and E. L. Hall. Dynamic omnidirectional vision for mobile robots. In *Journal of Robotic Systems*, 1986.
[4] H. Choset, K. M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun. *Principles of Robot Motion: Theory, Algorithms, and Implementation*. MIT Press, 2005.
[5] A. Clerentin, L. Delahoche, E. Brassart, and C. Drocourt. Localisations absolues en milieu intérieur basées sur la vision omnidirectionnelle. *TS. Traitement du signal*, 20:183–212, 2003. French.
[6] A. J. Davison, Y. Gonzlez, and N. Kita. Real-time 3d slam with wide-angle vision. In *IAV 2004*, July 2004.
[7] C. Gamallo, J. E. Domenech, P. Quintía, C. V. Regueiro, and J. Correa. Localización de un robot móvil en interiores mediante visión omnidireccional y balizas naturales. In *Workshop de Agentes Fsicos*, pages 119–126, Setiembre 2007. In Spanish.
[8] 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.
[9] J. P. Roda, J. M. Saez, and F. Escolano. Ceiling mosaics through information-based slam. *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 3898–3904, Oct. 29 2007-Nov. 2 2007.
[10] 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.
[11] H. Tamini, H. Andreasson, A. Treptow, Duckett. A., 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.
[12] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Haehnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Probabilistic algorithms and the interactive museum tour-guide robot Minerva. volume 19, 2000.
[13] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics.* MIT Press, 2003.



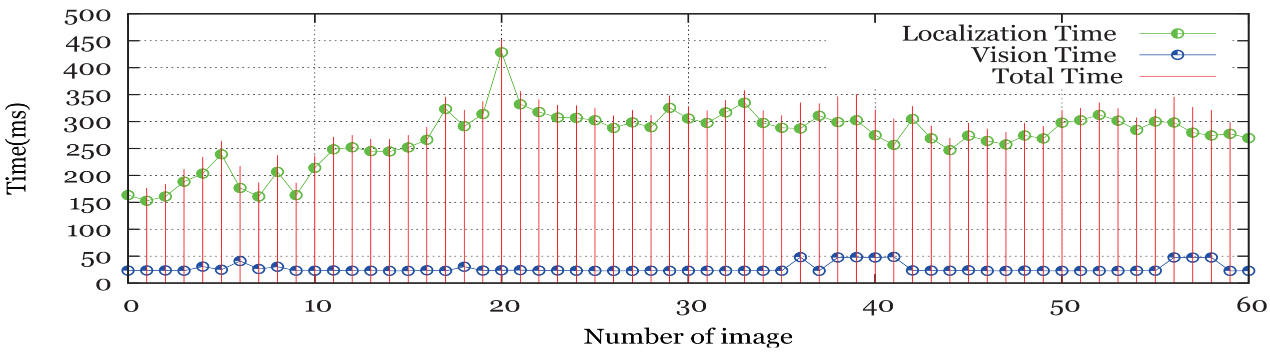
(a)



(b)



(c)



(d)

Fig. 10. Experiments on the Domus Museum: (a) Omnivision localization (gray) plotted on the grid map created with laser data (real trajectory is marked with dark circles). (b) Position error between laser pose estimation and the omnivision localization.(c) Orientation error between laser pose estimation and omnivision localization. (d) Time for procedessing each images.