3D Localization with Conical Vision

Cyril Cauchois, Eric Brassart, Laurent Delahoche, Arnaud Clerentin

CREA (Center of Robotic, Electrotechnic and Automatic)

IUT Département Informatique - Avenue des facultés – 80025 Amiens Cedex 1 – FRANCE

{Cyril.Cauchois, Eric.Brassart, Laurent.Delahoche, Arnaud.Clerentin}@u-picardie.fr

Abstract

This paper deals with an absolute mobile robot self-localization algorithm in an indoor environment. Until now, localization methods based on conical omnidirectional vision sensors uniquely used radial segments from vertical environment landmarks projection. The main motivation of this work is to demonstrate that the SYCLOP sensor can be used as a vision sensor rather than a goniometric one. We will show how the calibration allows us to know the omnidirectional image formation process to compute a synthetic image base. Then, we will present the spatial localization method using a base of synthetics images and one real omnidirectional image. Finally, some experimental results obtained with real noisy omnidirectional images are shown.

1. Introduction

To operate successfully, autonomous mobile robots must know where they are. "Mobile robot self-localization" has been, and still is, a considerable subject of research. The problem is that the actual position and orientation always differs from the pose that it is commanded to hold (wheel slippage, ground defaults, …). Proprioceptive sensors are generally not sufficient to locate a mobile robot, thus exteroceptive techniques have been elaborated in order to estimate accurately the robot's configuration. The principal difficulty with an exteroceptive sensor, is to solve the matching between the representations of the landmarks provided by the robot's sensors and the landmarks themselves. In a real situation, the absolute matching is quite difficult because the observations are not error-free. Moreover, the matching phase is, generally, very time consuming if one does not consider judicious strategies to reduce the combinatory aspect. In this work, we use only one exteroceptive sensor called SYCLOP that means Conical SYstem for LOcalization and Perception. Such omnidirectional vision sensor are called "catadioptric systems" ([1] [13] [14]). Our system, as Yagi's one [14], is constituted of an oriented vertical camera under a conical mirror. The majority of authors using this type of mirror uniquely process the radial features. It is due to the fact that straight lines other than verticals do not project themselves following a simple mathematic model. In fact, in all these

applications, the omni-sensor is used as a goniometric one.

This work is the meeting point of several previous works [3][4][5]. In this paper, we consider SYCLOP as a vision sensor. Section 2 will refer the reader, if necessary, to get information about the mathematical model created for SYCLOP and the protocol used to calibrate it. Then section 3 will present the SYCLOP simulator developed with the defined model. In section 4, we will explain the localization method based on a panoramic synthetic image base and an automatic points extraction. We will show in which manner we have reduced the computational time of matching by subdividing the robot's evolution field. Then, we will see how we can extract some features (points) for the spatial localization. Finally, experimental results with real acquisitions matched with synthetic pictures from the image base are given. We conclude our subject by a discussion on perspectives offered by this method and how this work can be extended.

2. The SYCLOP model and calibration

The reader will be able to get details about the SYCLOP model and its calibration by referring to [3].

3. The simulator

With the mathematical model of SYCLOP and its calibration, we have implemented a SYCLOP simulator. With the help of a 3D environment map, the simulator is able to compute synthetic images close to real ones :
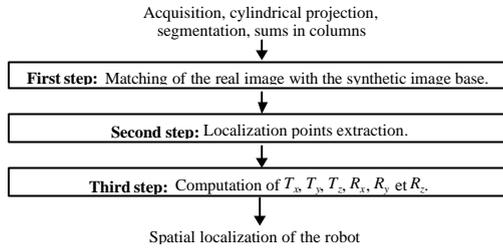
Table 1: Comparison between real and synthetic images.

Real image	Simulated image	Superimposition
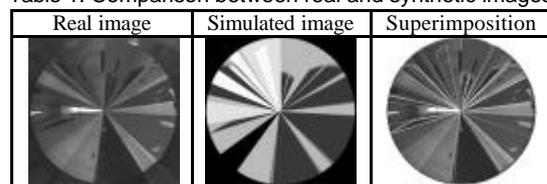
4. The localization

This localization method will be split in three different steps. After an acquisition, we compute a cylindrical projection in order to get a panoramic image rather than an omnidirectional. Then, the first step consists in estimating the robot's pose with the help of a set of synthetics images. The second step is the extraction of the points in order to estimate the spatial position of the

sensor. In theory, only 3 points are necessary to estimate the six degrees of freedom. To limit estimations errors, we have chosen to compute the spatial localization with 12 points. Finally, the third step is the spatial localization. This diagram summarizes the method:

Acquisition, cylindrical projection, segmentation, sums in columns

First step: Matching of the real image with the synthetic image base.

Second step: Localization points extraction.

Third step: Computation of T_x, T_y, T_z, R_x, R_y et R_z.

Spatial localization of the robot

4.1. The environment

Our simulator is able to compute synthetic images in a "flat" model (no texture possibility). Furthermore, we wanted an environment without any vertical beacons in order to not be tempted to use SYCLOP as a goniometric sensor. Consequently, we chose to create an environment comprising a lot of polygons (which act as texture) with no long vertical straight lines. The environment of work (Figure 1) has the following measurements: 2m by 3.5m. It is composed of 5 blocks of 1.25m height and different widths. As you can note in Figure 1, we have covered each block with different patterns. These motives are black on a white background in such a way that contour extraction can be easily done.
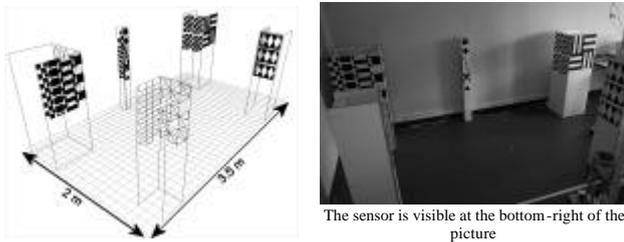

The sensor is visible at the bottom-right of the picture

Figure 1: the environment.

4.2. Step 1: The robot's pose estimation

The first step of this method is the robot's pose estimation. We will use an image base to estimate the configuration of the robot in the environment.

4.2.1. The image base

The first solution could consist in displacing the sensor everywhere in its environment in order to construct the image base. This solution has already been developed ([8] [9]) but it is not very convenient, especially if we want a thin grid. To displace the robot anywhere in the environment and to measure with accuracy the different positions is a relatively long and trying task. The second solution will consist in using a base of synthetic pictures. With our simulator, we are able to calculate synthetic pictures close to real ones. The disadvantage is that a sufficiently accurate 3D map of the environment is

necessary, which is not the case for a real picture base. On the other hand, there is no need to displace the robot anywhere in the environment. We only have to calculate all synthetic pictures to the different desired places.

The panoramic picture interest is double. The first advantage resides in the size of pictures. With resolutions that we fixed (1440x100), the storage of a picture requires 142 Ko. That reduced the necessary disk space. The second advantage is the calculation of rotations. On a panoramic picture, the horizontal shift corresponds to the rotation of the picture. Calculations are simplified and accelerated.

Finally, we have subdivided the robot's evolution environment with a step of 5 centimeters. Thus, the image base is composed of 1906 synthetic panoramic images. Once we have the 3D map, the computation of the 1906 synthetics panoramic images requires about 1 hour with a Pentium® III 800 MHz.

4.2.2. The matching

An image base is useful, however the major problem remains the matching. Contrary to applications that already exist [8] [9], we are going to try to match a real picture with a synthetic one. Nevertheless, our synthetic panoramic pictures are deprived of textures, and information that we possess is not on 360° but solely on five blocks distributed in the environment. Therefore, we can only match zones of pictures that correspond to projections of blocks onto the image plan.

Our first idea was to uniquely calculate 2D correlations between the real panoramic picture and synthetic ones from the base. Our panoramic pictures have a horizontal definition of 1440 pixels which permits us to calculate rotations of 0.25 degrees. So, we have 1440 2D correlations to compute for each position. This method is too gluttonous in calculation time especially as we have 1906 different positions to test.

To limit calculations, and to decrease the time of localization, we looked for a means to pass 2D correlation to a correlation of 1D signals. Elements of the scene are composed of black motives on a white background. If pictures are similar enough, the contours extraction of the real picture should be close to the contours extraction of the synthetic one. Therefore, we chose to calculate the sums in columns of binary image from contour extraction. In this way, we will match two 1D signals while calculating their correlation. The Figure 3 shows a real picture and a synthetic picture at a near position with their segmentation and their respective sums in columns. As we can see, the two signals (e) and (f) are very similar.
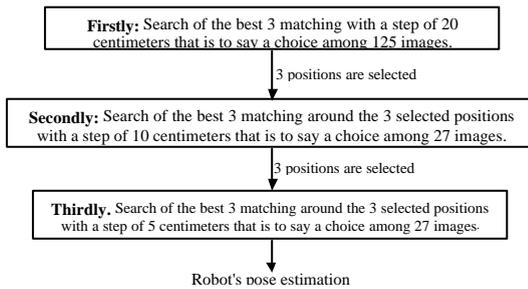
4.2.3. The matching algorithm

Firstly: By experimenting, we noted that the image base had to correspond to a grid of the environment with a step

of 20 centimeters. Beyond this distance, pictures are not similar enough to ensure a correct matching. So, in our environment the initial base will be composed of 125 sums in columns. During the first stage, we select the three best correspondences according to the criteria of 1D correlation between the sum in columns of the segmentation of the real image and sums in columns of segmentation of synthetic images from the base. In this stage, solely sums in columns are used.

Secondly: Then, around the three retained positions, we calculate correlations with a step of 10 centimeters. It gives us 27 positions to test. For each position, we determine the best orientation with the computation of the 1D correlation with sums in columns that we weight with the 2D correlation between the real picture and synthetics ones. The same weight is given to the two correlations. The three best correspondences are selected for the continuation of the process. In this stage, sums in columns and images are used with the same weight.

Thirdly: As we did previously, we are going to calculate correlations around the three retained positions, but this time with a step of 5 centimeters. Again, we have 27 positions to test. As before, we weight the correlation between sums in columns and the pictures. However, the 2D correlation between real and synthetic images has a weight of 2 against 1 for the 1D correlation between sums in columns. In this stage, correlations between pictures are major in relation to correlations between sums in columns.

Finally, for one estimation, we have a maximum of 179 tests to compute in a base that actually contains 1906 panoramic pictures. The Figure 2a shows an example of a robot's pose estimation with all tested positions. In this example, 172 positions have been tested. So, the following diagram summarizes the three stages of the robot's pose estimation algorithm :

Firstly: Search of the best 3 matching with a step of 20 centimeters that is to say a choice among 125 images.

3 positions are selected

Secondly: Search of the best 3 matching around the 3 selected positions with a step of 10 centimeters that is to say a choice among 27 images.

3 positions are selected

Thirdly. Search of the best 3 matching around the 3 selected positions with a step of 5 centimeters that is to say a choice among 27 images.

Robot's pose estimation

4.2.4. Experimental results

The goal of this experiment is to validate the matching method previously mentioned. Three trajectories have been implemented: two straight ones and one ovoid.

The first trajectory is straight and is 2.65 meters long. During the displacement, an image is grabbed every 5 centimeters. For this trajectory we obtain 54 pictures to process. Figure 2b presents results obtained during the 54 estimations. No bad matching is noted. The maximal gap

between a supposed position, and a calculated position is 43.51 millimeters with an average gap of 22.48 millimeters. In the ideal case, all gaps should have been lower than half of the diagonal (35.35 millimeters), but the results are very satisfactory. The second trajectory is also a straight line. It measures 2.35 meters which gives us 48 acquirements to process. Figure 2c shows the results of matching. Once again, there were no major mistakes since the noted maximum gap is 44.96 millimeters, and the average gap is 21.66 millimeters. Finally, the last trajectory is an ovoid. It measures 5.6 meters and includes 114 pictures. Figure 2d presents results of estimations. This time, the maximum gap is more important since it is 56.44 millimeters and the average gap is 22.69 millimeters. But it always remained very satisfactory.
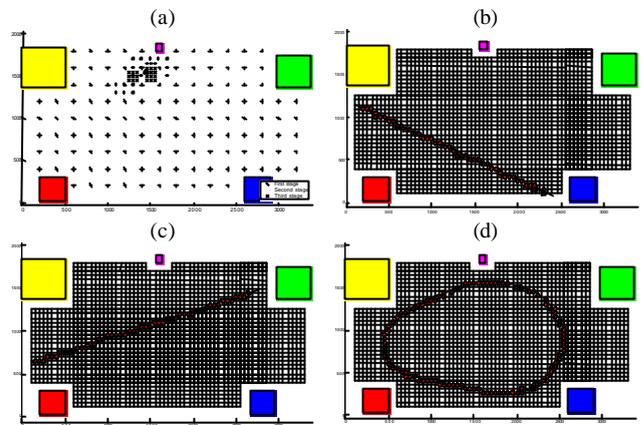

Figure 2: (a) Representation of all tests done during a robot's pose estimation, - (b, c, d) Results of matching between the synthetic images base and, respectively, the first, the second and the third trajectory.
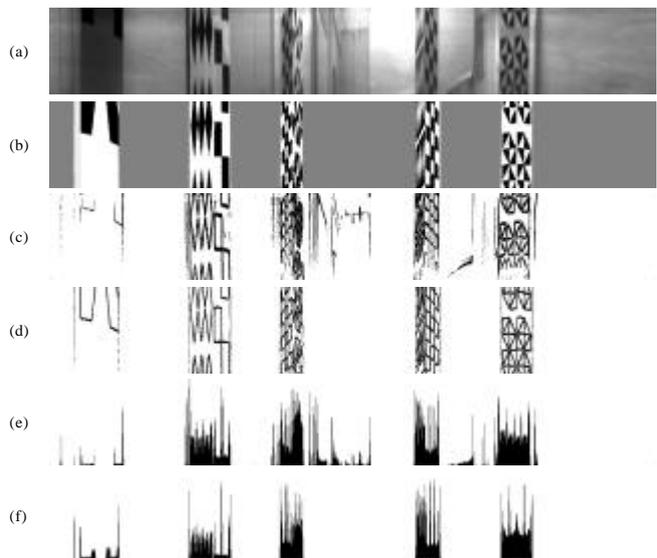

Figure 3: Results for acquisition n°28 of the third trajectory.
(a) Panoramic image of the 28th acquisition – (b) Synthetic panoramic image associated with (a)
(c) Segmentation of the panoramic image (a) – (d) Segmentation of the synthetic panoramic (b).
(e) Sums in columns of the segmentation of (c) – (f) Sums in columns of the segmentation of (d)

Figure 3 shows an example of matching between a real image (acquisition n°28 of the third trajectory) and the synthetic panoramic image base. In this example, the gap between the supposed robot's location and the position of the associated synthetic image is around 35 millimeters. On images Figure 3a and Figure 3b, we can see this difference, especially on the left of images. This is due to lighting conditions of the real environment.

As always with SYCLOP, the orientation of the robot is estimated with a high precision. During the first trajectory, the supposed orientation of the robot was 106.54°. The average orientation calculated is 106.32°. In the second trajectory the supposed orientation was 64.86°, and the average of evaluations is 64.63°.

4.3. Step 2: Localization points extraction

At this moment, we have an estimation of the robot's pose. In order to achieve the extraction of localization points, wherever the sensor is located in the environment, we defined a graph of view associated with the image base. In other words, to each position listed in the image base we associated twelve 3D points of the environment map. To choose the twelve points, we fixed ourselves these rules: the points must be visible at surroundings positions, and their projection must not be close to the top or bottom of the panoramic picture. To comment these choices, let's take as example acquirement n°28 of the third trajectory. The gap between the supposed position and the position of the synthetic picture is around 35 millimeters. It is a relatively important gap. If you look at the corresponding real and synthetic pictures, you can note this difference, especially on the left part (Figure 3).

The view graph gives us twelve 3D points that we project onto the synthetic picture in order to extract sub-pictures that will help us to determine the position of these points in the real picture. Figure 4 shows the twelve sub-pictures extracted from the panoramic picture. They have for dimension 13 by 13 pixels and are centered on the point to extract.

On Figure 3, we see that the black parts of the left block were not meaningful during the correlation calculation. Therefore, if we wish to extract points associated with these motives (which is the case), it will be necessary to make a relatively large research in the real picture. That's why we chose points that are also visible from the neighboring positions. Thus, with an error of ±5 centimeters around the estimated position and ±1 degree in orientation, we get research area materialized by the white rectangles on Figure 5. As you can see, more the point to extract is close to the position of the sensor and more the area of research is large.

To improve the extraction, we enhance contrasts by applying a stretch of histogram on each research area.

Then, the research is achieved by doing a correlation computation. We put the synthetic sub-picture onto the area of research, and we calculate all correlations by displacing it all over the research area. We preserve the position with the strongest coefficient. Figure 6 shows positions of the twelve sub-pictures onto their respective research area.
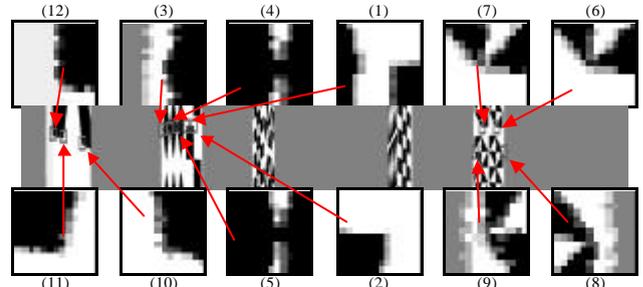

Figure 4: Sub-picture extraction from an image of the base.

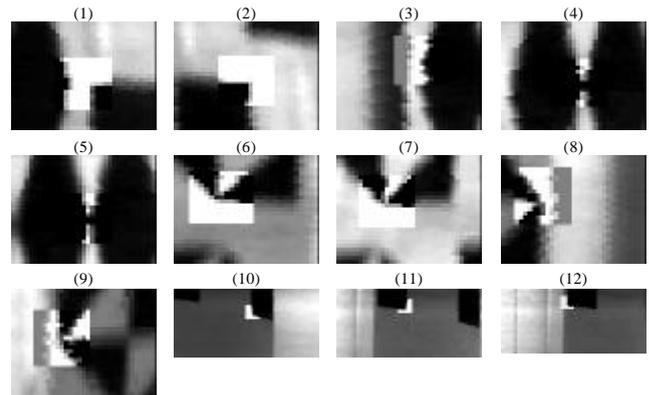Figure 5: Real picture with the 12 research areas.

Figure 6: Matching between synthetic sub-picture (Figure 4) and research area from the real picture (Figure 5).

Except for picture n°3, the matching is satisfactory. As the point to extract is situated to the center of the synthetic sub-picture, we get point coordinates in the panoramic picture. Of course, since the synthetic panoramic picture associated with the real picture during the previous step (step 1) didn't correspond exactly, patterns numbers 10, 11 and 12 are going to generate an erroneous extraction, as well as the point number 3.

With coordinates of localization points in the panoramic picture and the mathematical model we can project these points onto the real omnidirectional picture. In this way we get points with sub-pixel coordinates. Figure 7 presents the twelve points extracted from acquirement n°28 of the third trajectory.
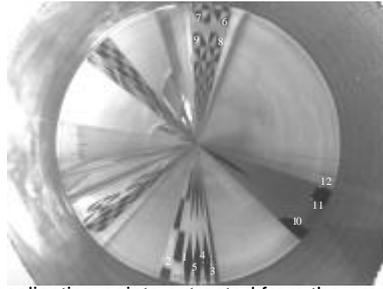
Figure 7: Localization points extracted from the omnidirectional picture n°28 of the third trajectory.

4.4. Step 3: Spatial Localization

At this step, we have an estimation of the robot's pose and we have a set of twelve localization points (their 3D coordinates and their 2D coordinates extracted from the omnidirectional image). The aim of this last step is to estimate the 3D position of the sensor and so, the spatial localization of the robot. To achieve this localization, we have split this step in two stages.

The first stage will consist in refining the robot's pose in the plan (3 degrees of freedom) and the second stage will be the spatial localization (6 degrees of freedom).

4.4.1. The refinement

First, we define a window of 10cm by 10cm centered on the estimated position. Then, we sample this window with a step of 1cm. It gives us 100 positions. For each position, we compute the projection of the twelve 3D points onto the image plan.

Next, we search in these 100 positions the best matching. This research is effected by using the Hausdorff distance as criteria to minimize [6]. The Hausdorff distance measures the extent to which each point of a "model" set lies near some point of an "image" set and *vice versa*. Thus this distance can be used to determine the degree of resemblance between two objects that are superimposed on one another. Given two finite sets $A=\{a_1, …, a_p\}$ and $B=\{b_1, …, b_q\}$, the Hausdorff distance is defined as

$$H(A,B) = \max(h(A,B), h(B,A)) \qquad (1)$$

where

$$h(A,B) = \max_{a \in A} \min_{b \in B} \|a - b\| \qquad (2)$$

and $\|\cdot\|$ is some underlying norm of the points of A and B (e.g., the L_2 or Euclidean norm). The function $h(A,B)$ is called the *directed* Hausdorff distance from A to B. It identifies the point $a \in A$ that is the farthest from any point of B, and measures the distance from a to its neighbor in B. Intuitively, if $h(A,B) = d$, then each point of A must be within the distance d of some point of B, and there also is some point of A that is exactly distance d from the nearest point of B (the most mismatched point). The Hausdorff distance, $H(A,B)$, is the maximum of $h(A,B)$ and $h(B,A)$. Thus it measures the degree of

mismatch between two sets, by measuring the distance of the point A that is the farthest from any point of B and *vice versa*. Let's consider that A is the set of points computed with the SYCLOP model from the synthetic picture, and B the set of points extracted from the real picture.

The best correspondence gives the estimation of position (T_x, T_y) and orientation (R_z) of the sensor. Finally, we refine the estimation by minimizing the SYCLOP model by estimating the rotation around the cone axis.

4.4.2. The spatial localization

This last stage permits us to localize the SYCLOP sensor spatially in its environment. The final values of the previous stage are going to act as initialization for the minimization of the SYCLOP model. In this way, we are going to be able to estimate the rigid motion existing between the world (the environment) and the sensor (the robot). This minimization is calculated with the help of Levenberg-Marquardt algorithm [12].
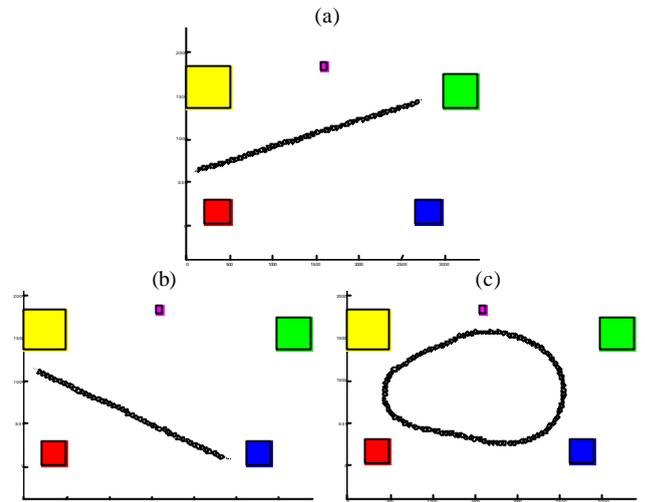
4.5. Experimental results

Figure 8: 2D projection of localization results for respectively, the first, the second and the third trajectory.
The stippled line represents the supposed trajectory.

For this experimentation, we take the three trajectories of section 4.2.4. As shown in Figure 8a, results obtained are very interesting. In fact, the average gap between supposed positions and estimated ones was about 22.5 millimeters and now it is about 12 millimeters. The maximum gap during the first step was about 43.5 millimeters and now it is about 21.7 millimeters. We improved the precision of localization nearly by a factor 2. Note that these gaps are computed in the plane and not in the space. In fact, we only have a 2D supposed position. With regard to the orientation, the average orientation in step 1 was 106.3° and now it is 106.2° where the supposed one is 106.5°.

Results obtained with the second trajectory are very accurate too. Once again, the result improvement is about a factor 2. The average gap was about 21.7 millimeters and now it is about 11.1 millimeters, and the maximum gap was 45 millimeters and now 19.4 millimeters. The average orientation was 64.6° and now 64.8° where the supposed one is 64.9°. Figure 8b shows a 2D projection of spatial localizations.

Results for the last trajectory are better (Figure 8c). The average gap between the supposed positions and the estimated ones was 22.7 millimeters for the first step and now it is about 9.7 millimeters, and the maximum gap was 56.4 and now it is 19.3 millimeters.

Note: videos of these demonstrations can be downloaded from: http://www.iut-amiens.fr/~cauchois/labo/Recherche/index_en.html

5. Conclusions and Future Works

The goal of this paper was to demonstrate that the SYCLOP sensor can be used as a vision sensor rather than a goniometric one. The presented method uses a synthetic picture base in order to achieve a spatial localization of a mobile robot equipped with an omnidirectional vision sensor. This method permits us to achieve an absolute localization of a mobile robot in a robust, accurate and reliable manner. Indeed, on the set of the three trajectories, that is to say 216 acquirements and so 216 absolute localizations, no bad matching was committed. The most important gap recorded is 12.03 millimeters in an environment of 7 m^2. Furthermore, the use of sums in columns and the division of the process in 3 steps permit us to make an absolute spatial localization in only 40 seconds on a Pentium$^{®}$ III 800 MHz.

We didn't deal with the case of occlusion because knowledge about the environment does not cover the 360° of the field of view. In fact, in the synthetic panoramic images, a maximum of 50% of the image was known (5 blocks). Thus, we considered that the lack of knowledge between the blocks can be seen like occlusions.

A disadvantage of this method is that it uses a base of synthetic images created using a three-dimensional map of the environment. To ensure a very good correspondence between the real images and the synthetic ones requires a sufficiently precise map and a good calibration of the vision sensor. But this disadvantage is not inherent in this method but with all the methods which use a base of knowledge *a priori* (disappearance or appearance of an object). In fact, the major disadvantage of the solution presented is that the environment is artificial. It would have been interesting to test our method in a more "human" environment, such as offices or corridors. In this kind of environment the vertical landmarks presence would be supplementary information that we could exploit as in the previous applications. This way, we will be able to compare our method to existing ones [10]. In one hand,

to test our method in such environment, we are working on the improvement of the simulator in order to be able to calculate textured synthetic pictures. In this way, we should match the real image with the base by compute texture matching or with eigenspace [9]. On the other hand, the acquisition of the 3D model is an other important problem because its accuracy will have a great influence on localization results. We don't have resolve this problem yet, but we are working on it.

Acknowledgement

References

[1] S. Baker and S. K. Nayar "A Theory of Catadioptric Image Formation" *Proceedings of the 6th International Conference on Computer Vision*, Bombay, pp. 35-42, Juanuary 1998.

[2] R. Benosman, T. Maniere and J. Devars. "Multidirectional stereovision sensor, calibration and scenes reconstruction", In *ICPR*, Vienna, Austria, pp. 161-165, 1996.

[3] E. Brassart, L. Delahoche, C. Cauchois, C. Drocourt, C. Pegard, M. Mouaddib "Experimental results got with the omnidirectional vision sensor : SYCLOP", *Proc. IEEE Workshop OMNIVIS'00*, USA, p. 145-152, June 12, 2000.

[4] C. Cauchois, E. Brassart, B. Marhic, C. Drocourt – *" Absolute localization with the calibrated SYCLOP sensor "* - Proc. of the IEEE Workshop on "Omnidirectional Vision", Budapest, Hungary August 22, 2001, p. 65-72.

[5] C. Cauchois, E. Brassart, B. Marhic, C. Drocourt – *" An Absolute Localization Method using a Synthetic Panoramic Image Base "* - Proc. IEEE Workshop on Omnidirectional Vision, Copenhagen, Denmark, June 2, 2002, p. 128-135.

[6] D. P. Huttenlocher, G. A. Klanderman, W. J. Rucklidge "Comparing Images Using the Hausdorff Distance", *IEEE Transactions PAMI*, 15(9):850-863, 1993.

[7] H. Ishiguro, S. Tsuji "Applying Panoramic Sensing to Autonomous Map Making a Mobile Robot" *in Proc, Int. Conf. on Advanced Robotics*, pp127-132, november 1993.

[8] H. Ishiguro and S. Tsuji "Image-based memory of environment" *IEEE Int. Conf. on Intelligent Robotics and Systems*, pp 634-639, 1996.

[9] M. Jogan and A. Leonardis "Robust localization using the eigenspace of spinning-images" *Proc. IEEE Workshop on Omnidirectional Vision*, USA, pp 37-44, June 12, 2000.

[10] M. Mouaddib, B. Marhic "Geometrical Matching for Mobile Robot Localisation" *IEEE Transaction on Robotics and Automation*, Vol. 16, No 5, pp 542-552, Oct 2000.

[11] Shree K. Nayar "Omidirectional vision" *The eighth int. Symposium of Robotics Research*, October 3-7, 1997.

[12] "Numerical Recipes in C - Second Edition", *Cambridge University Press*, ISBN 0-521-43108-5.

[13] D.W. Rees "Panoramic television viewing system", United states Patent N°3, 505, 465 Apr. 1970.

[14] Y. Yagi., S. Kawato and S. Tsuji "Real-time omnidirectional image sensor (COPIS) for vision-guided navigation", *IEEE Transaction on Robotics and Automation*, Vol. 10, N°1, 1994.