Visual Odometry using the Global-appearance of Omnidirectional Images

F. Amorós, L. Payá, D. Valiente, A. Gil and O. Reinoso
System Engineering and Automation Dep., Miguel Hernández University,
Avda. de la Universidad, s/n. 03202, Elche (Alicante), Spain

Keywords: Multi-scale Analysis, Global-appearance Descriptors, Visual Odometry, Robot Navigation.

Abstract: This work presents a purely visual topologic odometry system for robot navigation. Our system is based on a Multi-Scale analysis that allows us to estimate the relative displacement between consecutive omnidirectional images. This analysis uses global appearance techniques to describe the scenes. The visual odometry system also makes use of global appearance descriptors of panoramic images to estimate the phase lag between consecutive images and to detect loop closures. When a previous mapped area is recognized during the navigation, the system re-estimates the pose of the scenes included in the map, reducing the error of the path. The algorithm is validated using our own database captured in an indoor environment under real dynamic conditions. The results demonstrate that our system permits estimating the path followed by the robot with accuracy comparing to the real route.

1 INTRODUCTION

In order to navigate autonomously, a mobile robot should be equipped with one or several sensors that gather the information of its surrounding environment. Additionally, it must interpret the data that those sensors provide.

Normally, this information is used to build an internal representation of the navigation area in a map, and it allows the robot to localize itself.

In the literature, we can find a wide variety of sensors that provide the robot with information of the surroundings. Among all the possibilities, vision systems have become popular sensors for robot control due to the richness of the information they provide, their relative low weight and cost, and the variety of possible configurations. This work is focused on omnidirectional scenes obtained with a catadioptric system, composed of a hyperbolic mirror and a CCD camera.

A key aspect in visual navigation is the description of the scenes. In this regard, two main categories can be found: feature based and global-appearance descriptors. The first approach is based on the extraction and description of significant points or local regions from the scene. It is possible to find many examples of use of these descriptors applied to localization and mapping tasks, such as features (Lowe, 2004; Lingua et al., 2009), SURF (Murillo et al., 2007; Bay et al., 2008), or Harris edge and corner detector (Gartshore et al., 2002). On the other hand, global-appearance descriptors try to describe the scene as a whole, without the extraction of local features or regions. These techniques have a special interest in unstructured and changing environments where it may be difficult to find patterns to recognize the scene. For example, (Krose et al., 2007) demonstrate the robustness of PCA (Principal Component Analysis) applied to image processing; (Menegatti et al., 2004) take advantage of the properties of the Discrete Fourier Transform (DFT) applied to panoramic images in order to build descriptors of the scene and, at last, (Kunttu et al., 2004) describe the behaviour of a descriptor based on Fourier transform and Wavelet filter in image retrieving tasks.

Regarding the representation of the map, three main approaches stood out: metric, topological and hybrid techniques. Metric maps (Moravec and Elfes, 1985) include information of distances with respect to a predefined coordinate system. In these maps, we can know the position of the robot except for an uncertainty associated with the sensor error. However, they usually have a high computational cost.

In contrast, topological techniques use graph-based representations of the environment. In those maps, there are not absolute distances. Despite this

Amorós F., Payá L., Valiente D., Gil A. and Reinoso O..
Visual Odometry using the Global-appearance of Omnidirectional Images.
DOI: 10.5220/0005062604830490
In *Proceedings of the 11th International Conference on Informatics in Control, Automation and Robotics* (ICINCO-2014), pages 483-490
ISBN: 978-989-758-040-6

fact, they usually contain enough information for a robot to autonomously navigate within the environment. As an example, FAB-MAP (Cummins and Newman, 2011) is a well-known topological SLAM approach, based on SURF features extraction to describe the appearance of the image.

Hybrid techniques try to take advantage of both topological and metric proposals. Normally, hybrid maps use metric in order to build sub-maps of separated areas, whereas topological relations are used in order to connect the different areas of the map. It is also possible to introduce the topological relations to carry out loop closures in metric maps. An example of hybrid SLAM algorithm is RatSLAM (Milford et al., 2004).

In this work we propose a framework for the estimation of the path of a robot using only visual information. Comparing to other works, we make use only of omnidirectional views and global appearance descriptors to estimate the movement of the robot. We describe our localization as topological, since no metric information is provided. However, as shown in this paper, our algorithm is able to describe the route path of the robot with a layout similar to the real metric distribution.

The paper is structured as follows: Next section describes the use of the Multi-Scale analysis to omnidirectional images in order to obtain a topological measure of displacement between scenes. Section 3 presents an algorithm to estimate the relative pose of two omnidirectional images. In section 4 we explain the application of this algorithm to navigation tasks, including loop closures that improve the estimation of the path. Section 5 includes the experimental database and results. Finally, in Section 6 we summarize the main ideas obtained in this work.

2 MULTI-SCALE ANALYSIS IN OMNIDIRECTIONAL IMAGES

In this section, we describe a visual odometry system that extends the use of the Multi-Scale analysis to omnidirectional information. The Multi-Scale Analysis permits obtaining information about relative displacement between two scenes using artificial zooming of plane projection images. In a previous work, (Amorós et al., 2013) we developed the use of Multi-Scale analysis in topological map building and route path estimation of a robot equipped with a fish-eye camera. As stated in that work, the Multi-Scale analysis only works properly if the digital zooming is applied to a perspective image captured in a plane perpendicular to the robot movement. Figure 1 shows

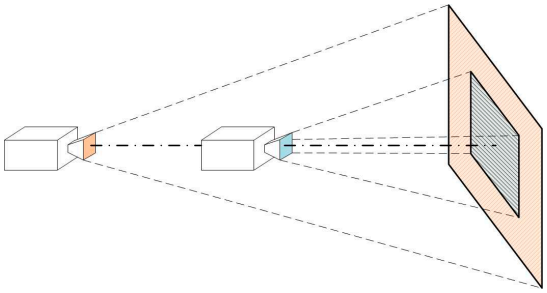


Figure 1: Visual field of two scenes captured by a visual system with a direction of travel perpendicular to the projection plane.

the field of view of a camera when it moves forward perpendicularly to its projection plane. It is possible to see that, the scene in the ahead position (represented in blue), correspondes to the central area of the camera field of view when it is in the back position (represented in orange). That way, if we select the central area of the orange image and re-scale it to the original image size (simulating a digital zoom), the appearance regarding the second image is more similar. The difference of scales between both images provides information about the direction and magnitude of the displacement between the points where the images were captured.

Our catadioptric system collects information of the entire field of view around the mirror axis. Taking advantage of the system calibration, it is possible to find out the direction of the rays that arrive to the mirror, and we can obtain projections using different geometries. For example, we can obtain the panoramic view of the scene. In the same way, we can obtain additional views by projecting the visual rays in some planes which are parallel to the mirror axis (Fig. 2). These perspective images are the projections that the Multi-Scale analysis uses.

In an omnidirectional image, we can extract two different plane projections which are perpendicular to the robot direction of travel: one in the forward direction, and the other in the reverse one. Figure 3 shows an example of the two plane projections we extract from the image. The blue arrow points forward, and the red one, backwards.

As the robot advances, it approaches the elements located in front of it, and moves away from the objects located behind it. This turns into a zoom-in of the front plane projection image, and a zoom-out of the backwards projection image. Therefore, in the Multi-Scale analysis, the scales of each orientation vary with different sign. The zooming in the projections is carried out by changing the focal distance of the projection plane. In Figure 4 we can see an example. The central focal length corresponds to $fc = 1.1$, and the

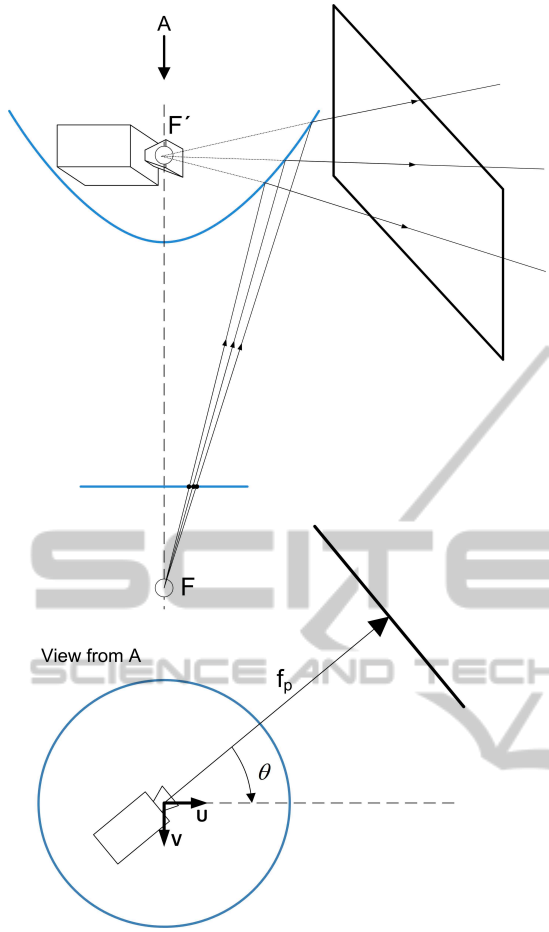


Figure 2: Representation of plane projection image from an catadrioptic visual system.

focal changes are added in the forward direction, and subtracted in reverse.

For each Δfc, we compute the descriptor $\mathbf{z}_{proj} \in \Re^{1\times L}$. This vector is made up of the global-appearance descriptors of both plane projection images. Taking into account the focal changes, $\mathbf{z}_{proj}$ is defined as:

$$\mathbf{z}_{proj,\Delta fc} = [\ \mathbf{z}_{forward,fc_{central}+\Delta fc}\ ,\ \mathbf{z}_{backward,fc_{central}-\Delta fc}\ ] \quad (1)$$

In order to estimate the relative displacement between two positions, denoted by $i-1$ and i, the algorithm computes the first position descriptor $\mathbf{z1}^i_{proj,\Delta fc}$ using different Δfc. Then, we compare them with the descriptor of the second position without zooming ($\mathbf{z2}_{proj,0}$). The comparison is carried out using the Euclidean distance:

$$d^i_{proj} = \min\left(\sqrt{\sum_{l=1}^{L}\left((\mathbf{z1}^i_{proj,\Delta fc}(l))^2 - (\mathbf{z2}^{i-1}_{proj,0}(l))^2\right)}\right) \quad \forall\, \Delta fc. \quad (2)$$

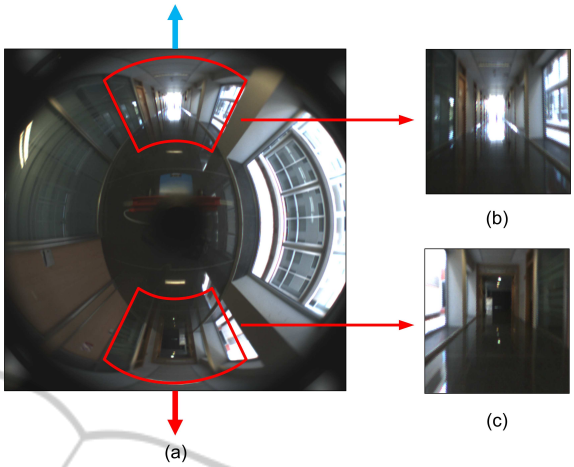


Figure 3: (a) Omnidirectional image, (b) forward and (c) plane projections in the direction of travel.

Once the algorithm has carried out all the comparisons, we select the association with lower image distance d_{proj}. The focal difference of this association (Δfc^i) denotes the relative displacement between scenes. Note that, since for the second image the algorithm considers $\Delta fc = 0$, the focal difference between images coincides with the Δfc of the first image.

The absolute value $|\Delta fc^i|$ is proportional to the displacement magnitude, whereas its sign informs about the direction of the movement (forward or reverse).

3 RELATIVE POSE ESTIMATION USING CONSECUTIVE SCENES

In the previous section, we describe the Multi-Scale analysis applied to omnidirectional images without considering changes in the robot direction. However, during a real navigation, the robot can change its direction, introducing a phase lag between consecutive images.

If we apply the Multi-Scale analysis to consecutive images that present a phase lag (Figure 5 (b)), the plane projections obtained in the direction of travel do not present the same orientation, and their visual information would not contain the same elements of the environment. Therefore, we must take into account the phase change $\Delta\theta$ to carry out a proper comparison.

Previous works (Murillo et al., 2013; Payá et al., 2014) demonstrate that global-appearance descriptors can cope with the problem of mobile location and orientation estimation using omnidirectional information. Those techniques usually make use of panoramic images. Moreover, independently of the

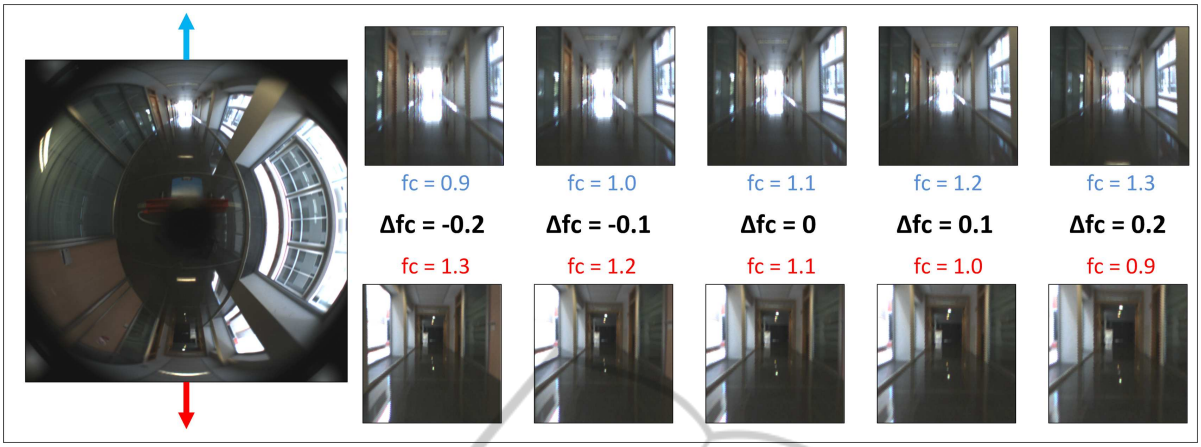


Figure 4: Change of focal length in the projective images in both forward and reverse direction in the Multi-Scale analysis.

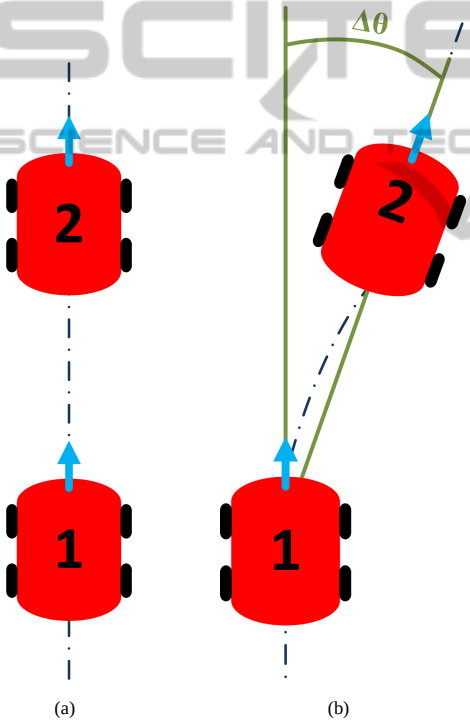


Figure 5: Example of different trajectories followed by the robot. (a) Rectilinear trajectory and (b) trajectory with a change of direction.

technique, we can obtain a measure of the certainty in the estimation, by means of the euclidean distance between phase descriptors (d_{phase}).

We propose to take advantage of the ability of global appearance descriptors to estimate the phase lag ($\Delta\theta$) between two scenes in order to determine the directions of the plane projections to be able to compare them using the Multi-Scale analysis.

Following the example in Figure 5 (b), the Multi-Scale analysis would change first the orientation of the plane projections of the first position, introducing a phase lag $\Delta\theta$, with respect to the direction of travel. That way, the plane projections of both positions will have the same direction.

We define the pose vector $[\, x^i\, y^i\, \theta^i\,]^T$ that includes the coordinates on the ground plane and the orientation of the robot. When a new image arrives, we compute the phase lag with the previous image, $\Delta\theta$, and apply the Multi-Scale analysis, obtaining Δfc^i. With that information, we update the position of the robot:

$$\begin{bmatrix} x^i \\ y^i \\ \theta^i \end{bmatrix} = \begin{bmatrix} x^{i-1} + \Delta x^i \\ y^{i-1} + \Delta y^i \\ \theta^{i-1} + \Delta\theta \end{bmatrix}, \; i = 1, \ldots, n \qquad (3)$$

with

$$\begin{bmatrix} \Delta x^i \\ \Delta y^i \end{bmatrix} = \begin{bmatrix} \Delta fc^i \cdot \sin\left(\theta^i\right) \\ \Delta fc^i \cdot \cos\left(\theta^i\right) \end{bmatrix} \qquad (4)$$

Therefore, we obtain a topological visual odometry system that allows the robot to estimate its current position regarding the previous one.

4 APPLICATION TO NAVIGATION TASKS

Used in a real navigation task, the visual odometry system exposed in the previous section will accumulate the error in the pose estimation as each new image arrives. To reduce this error, we include the detection of previously-visited positions of the route in order to carry out a loop closure that improves the estimation of the path. At every iteration, the robot creates the descriptor of the panoramic view of the current image, and saves it to a database. On the other hand, it estimates the pose of the current image n, $[\, x^n\, y^n\, \theta^n\,]^T$ regarding the previous route pose $n-1$ using Eq. 3.

Moreover, the algorithm compares the descriptor of the panoramic view of the current image and carries out a matching process with the previously mapped locations. To that end, it calculates the Euclidean distance of the current image descriptor and the descriptors in the database.

$\mathbf{z}^i_{pano} \in \Re^{1xM}$ is the vector that contains the descriptor of the $i-th$ panoramic image. The image distance between the current panoramic image (n) and all the images included in the database is defined as:

$$d^i_{pano} = \sqrt{\sum_{m=1}^{M} \left((\mathbf{z}^i_{pano}(m))^2 - (\mathbf{z}^n_{pano}(m))^2\right)} \quad i = 1,\ldots,n. \quad (5)$$

After that, the algorithm selects the association with the minimum distance, and checks whether the distance is lower than a fixed threshold, th_{pano}:

$$\min\left(d^i_{pano}, i = 1,\ldots,n-20\right) < th_{pano} \quad (6)$$

If the condition is not satisfied, the algorithm includes the new pose $[\, x^n\ y^n\ \theta^n\,]^T$ in the map, and it waits for a new image to arrive.

On the contrary, if the condition is satisfied, the robot supposes that the current position has been visited previously and carries out a loop closure correction, following these 3 steps:

1. Current Pose Estimation

   When the algorithm detects a loop closure, the current pose is re-estimated using the image matched in the database (Eq. 6). i_{loop} denotes the coefficient of the matched map image. The current pose $[x^{loop},\ y^{loop},\ \theta^{loop}]^T$ is now estimated regarding i_{loop}. For that purpose, we make use again of Eq. 3, considering the images i_{loop} and n. It is important to highlight that the pose of i_{loop} has lower uncertainty than the pose of $n-1$ since it has estimated in a previous iteration, and consequently, its pose error is lower.

2. Phase Error Correction

   We define the phase error as:

   $$e_{phase} = \theta_{loop} - \theta^n \quad (7)$$

   The algorithm propagates the correction of the error to the positions considered in the loop. This correction is weighted with the uncertainty associated in the phase estimation (d_{phase}). The phases of the poses included in the loop are re-estimated as:

   $$\overline{\Delta\theta^i} = \Delta\theta^i + e_{phase} \cdot \frac{d^i_{phase}}{\sum_{j=i_{loop}}^{n} d^j_{phase}},\ i = i_{loop},\ldots,n. \quad (8)$$

   That way, the orientation of the last route pose is the same that the estimated in the loop closure, i.e., $\overline{\Delta\theta^n} = \theta^{loop}$.

   The correction of the phase implies a change in the path coordinates:

   $$\begin{bmatrix} \overline{x^i} \\ \overline{y^i}\ \overline{\theta^i} \end{bmatrix} = \begin{bmatrix} x^{i-1} + \overline{\Delta x^i} \\ y^{i-1} + \overline{\Delta y^i} \\ \theta^{i-1} + \overline{\Delta\theta} \end{bmatrix},\ i = i_{loop},\ldots,n. \quad (9)$$

   with

   $$\begin{bmatrix} \overline{\Delta x^i} \\ \overline{\Delta y^i} \end{bmatrix} = \begin{bmatrix} \Delta fc^i \cdot \sin(\overline{\theta^i}) \\ \Delta fc^i \cdot \cos(\overline{\theta^i}) \end{bmatrix},\ i = i_{loop},\ldots,n. \quad (10)$$

3. XY Positions Correction

   Once we have re-estimated the poses of the route included in the loop by correcting their phase, the algorithm carries out a new error propagation using the XY coordinates of the pose estimated in the loop closure $[x_{loop},\ y_{loop}]$. The initial information includes the position of the route with the re-estimated phase $[\overline{x^i}, \overline{y^i}]$.

   The XY position error is defined as:

   $$\begin{bmatrix} e_x \\ e_y \end{bmatrix} = \begin{bmatrix} x^{loop} - \overline{x^n} \\ y^{loop} - \overline{y^n} \end{bmatrix} \quad (11)$$

   The propagation of the XY correction is weighted by the uncertainty in the relative displacement estimation in the Multi-Scale analysis d^i_{proj} (Eq. 2).

   Finally, the new coordinates of the route $\left[\,\overline{\overline{x}},\ \overline{\overline{y}}\,\right]^T$ are estimated as:

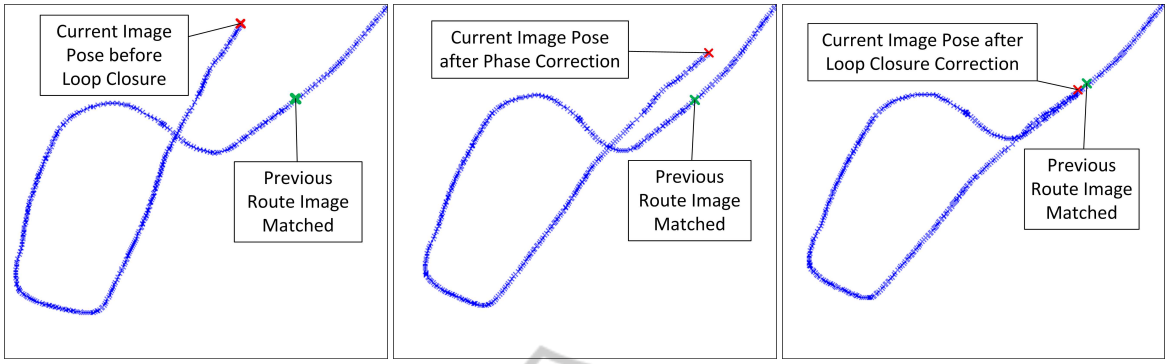
   $$\begin{bmatrix} \overline{\overline{x^i}} \\ \overline{\overline{y^i}} \end{bmatrix} = \begin{bmatrix} \overline{x^{i-1}} + \overline{\overline{\Delta x^i}} \\ \overline{y^{i-1}} + \overline{\overline{\Delta y^i}} \end{bmatrix},\ i = i_{loop},\ldots,n. \quad (12)$$

   with

   $$\begin{bmatrix} \overline{\overline{\Delta x^i}} \\ \\ \overline{\overline{\Delta y^i}} \end{bmatrix} = \begin{bmatrix} \overline{\Delta x^i} + e_x \cdot \frac{d^i_{proj}}{\sum_{j=i_{loop}}^{n} d^j_{proj}} \\ \\ \overline{\Delta y^i} + e_y \cdot \frac{d^i_{proj}}{\sum_{j=i_{loop}}^{n} d^j_{proj}} \end{bmatrix} \quad (13)$$

   After re-estimating the XY coordinates, the phase values may change. For that reason, the algorithm re-calculates them as:

   $$\overline{\overline{\theta^i}} = \arctan\left(\frac{\overline{\overline{y^i}} - \overline{\overline{y^{i-1}}}}{\overline{\overline{x^i}} - \overline{\overline{x^{i-1}}}}\right) \quad (14)$$



(a) (b) (c)

Figure 6: Example of route path correction with the loop closure. (a) Detection of a previously navigated area, (b) correction of phase estimations and (c) correction of XY coordinates and final layout.

Figure 6 shows the three steps in the loop closure. In Figure 6 (a), the algorithm has detected a loop closure. Figure 6 (b), shows the route once the phase error has been taken into account, and Figure 6 (c) shows the final layout of the route after re-estimating the XY positions.

As we can see in Eq. 6, the last 20 images are not taken into account when considering the loop closure. Those scenes present a high similarity with the current position scene, and would carry out a loop closure when, in fact, there is not.

5 EXPERIMENTS AND RESULTS

The objective of the experiments is to show that our approach is able to estimate the path followed by a robot using only visual information. Our experimental database is composed of a set of images captured while the robot goes trough a route in a real indoor office environment. The number of images is 1211, captured with a frequency equal to 1Hz, with a approximate navigation speed equal to 0.1 m/s. The scenes have been captured using a catadioptric vision system made up of a CCD camera and a hyperbolic mirror. The resolution of the images is 1280×960 pixels. We use two different projections of each omnidirectional image. On the one hand, the panoramic projections allow us to estimate the phase lag between consecutive positions, and to detect loop closures. On the other, the Multi-Scale analysis uses plane projective images. The resolution of the panoramic views is 128×512 pixels. Regarding the plane projection images, its resolution is 256×256 pixels. The acquisition of the omnidirectional images has been carried out using a robot Pioneer P3-AT, equipped with a laser rangefinder.

Figure 7 shows the route followed by the robot. Specifically, the robot navigates along a laboratory, corridors and common areas of a building. The path estimated with the laser data is shown in blue, which is used as our ground truth. It should be pointed out that there exist numerous windows in the building that cause important changes in the lighting conditions. Moreover, the path includes a slope in the floor.

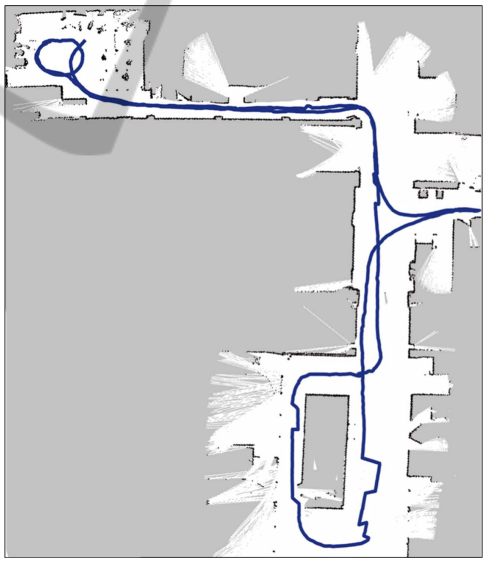


Figure 7: Route layout obtained from laser rangefinder (Ground Truth).

Regarding the global descriptors, we use the Fourier Signature (Menegatti et al., 2004) for both the panoramic view and the plane projections. The Fourier Signature is divided in two terms: the magnitudes of the transform, that permits associating scenes (both the panoramic images during the loop closure and the plane projections images in the Multi-Scale analysis), and the arguments, that contain information about the spatial distribution of the elements included in the scene. Applied to panoramic images, the mag-

nitudes are invariant to rotations, and the arguments allow us to estimate the phase lag between two scenes. N denotes the number of magnitude components selected in the Fourier Transform of each row, whereas N_{rot} is the number of arguments.

In the panoramic image, we set $N_{rot} = 32$, and N will be determined in a subsequent experiment. Regarding the plane projection image, the Multi-Scale analysis shows a correct performance with $N = 16$ elements in the image descriptor. The information of arguments is not used in this case. Regarding the scales, we use $fc \in [1.0 - 1.2]$, with $\Delta fc = 0.01$.

Since we use a topological approach, the estimation we obtain is expected to be similar to the actual trajectory except for a scale factor. Taking this into account, to know the accuracy of the results, we measure the error using the Procrustes analysis (Kendall, 1989; Dryden and Mardia, 1998). This analysis returns a standardized disparity measure $\mu \in [0,1]$ of the shape difference between the ground truth and the estimated path. $\mu = 0$ indicates that the reconstruction of the route has exactly the same shape that the real layout, i.e., there is no error.

Without considering loop closures, the algorithm accumulates the error associated to the pose estimation. The Procrustes analysis returns $\mu = 0.5596$. In Fig. 8 (a) two different paths are represented : in blue, the ground truth, and in green, the estimation of our algorithm. In this representation, the scale of our estimated path has been adapted to allow the comparison with the ground truth, since the ground truth is in meters, and our estimation is in Δfc.

Table 1 shows the results for different values of N (number of magnitude components per row selected in the Fourier Signature of the panoramic image), and the threshold th_{pano} (Eq. 6). We can check that, in all the configurations, the error is considerably lower than the results when the algorithm does not consider loop closures. We highlight the minimum error in bold, obtained when $N = 8$ and $th_{pano} = 3.3$. The estimation of the path using those parameters is shown in Fig. 8 (b). It is possible to check visually that our algorithm estimates the robot path with accuracy comparing to the real layout when using loop closures.

Table 1: Error in the route path estimation using Procrustes Analysis (μ) varying the loop closure threshold (th_{pano}) and the localization descriptor parameter (N) using the Fourier Signature.

th_{pano}	**N**	μ
2,30	8	0,0505
2,30	16	0,0735
2,30	32	0,0574
2,50	8	0,0565
2,50	16	0,0948
2,50	32	0,0578
2,90	8	0,0485
2,90	16	0,0449
2,90	32	0,0935
3,10	8	0,0447
3,10	16	0,0550
3,10	32	0,0949
3,30	**8**	**0,0383**
3,30	16	0,0532
3,30	32	0,0424
3,50	8	0,4656
3,50	16	0,0744
3,50	32	0,0539

6 CONCLUSIONS

In this work, we have presented a topological visual odometry system based on the global-appearance of omnidirectional information. This system adapts the Multi-Scale analysis, which provides a topological measure of displacement between plane projection images, to omnidirectional images. Moreover, a loop closure detection is introduced in the algorithm to reduce the accumulated error of the system. The loop closure also uses global appearance techniques to determine if the current image matches with a previously visited area.

The experimental results demonstrate that our visual odometry system is able to estimate the path followed by the robot with a similar layout to the real one, except for a scale factor. Regarding the loop closure detection, it reduces considerably the error in the path estimation. However, the results strongly depend on the association image threshold. If this threshold is too small, the image association is more reliable, but we reduce the number of loop closures. On the contrary, if the threshold is excessively high, the system might carry out false image association, and therefore, erroneous loop closures.

The results of this method encourage us to continue the research in topological visual odometry systems using the global-appearance of images. It might be interesting to extend this study to outdoor environments navigation. We can also consider the improvement of the loop closure estimation, re-estimating again the scene displacement with the Multi-Scale analysis after the phase correction, or comparing the results using different global-appearance descriptors.

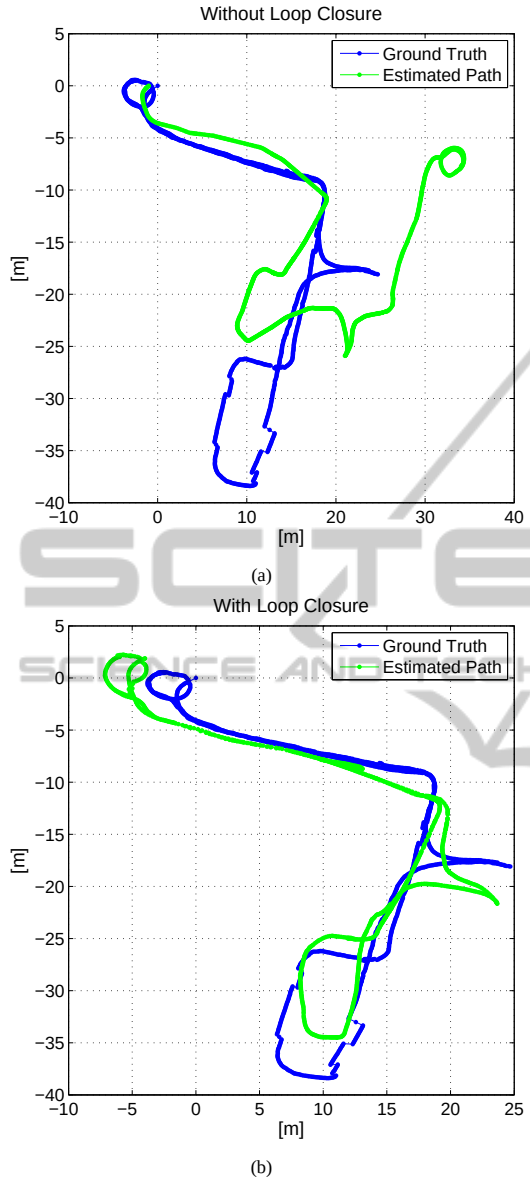


Figure 8: Graphical representation of the ground truth route and our visual odometry system path estimation (a) without loop closure corrections and (b) with loop closure corrections.

ACKNOWLEDGEMENTS

This work has been supported by the Spanish government through the project DPI2010-15308.

REFERENCES

Amorós, F., Payá, L., Reinoso, O., Mayol-Cuevas, W., and Calway, A. (2013). Topological map building and path estimation using global-appearance image descriptors. In *10th Int. Conf. on Informatics in Control, Automation and Robotics (ICINCO 2013)*, pages 385–392, Reykjavik, Iceland. SciTePress.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. (2008). Speeded-up robust features (surf). *Comput. Vis. Image Underst.*, 110(3):346–359.

Cummins, M. and Newman, P. (2011). Appearance-only slam at large scale with fab-map 2.0. *Int. J. Rob. Res.*, 30(9):1100–1123.

Dryden, I. and Mardia, K. (1998). *Statistical shape analysis*. Wiley series in probability and statistics. Wiley.

Gartshore, R., Aguado, A., and Galambos, C. (2002). Incremental map building using an occupancy grid for an autonomous monocular robot. In *7th Int. Conf. on Control, Automation, Robotics and Vision, 2002. ICARCV 2002.*, volume 2, pages 613 – 618 vol.2.

Kendall, D. (1989). A survey of the statistical theory of shape. volume 4, pages 87–99.

Krose, B., Bunschoten, R., Hagen, S., Terwijn, B., and Vlassis, N. (2007). Visual homing in enviroments with anisotropic landmark distrubution. In *Autonomous Robots, 23(3), 2007, pp. 231-245*.

Kunttu, I., Lepisto, L., Rauhamaa, J., and Visa, A. (2004). Multiscale fourier descriptor for shape-based image retrieval. In *Proceedings of the 17th International Conference on Pattern Recognition, ICPR 2004*, volume 2, pages 765 – 768 Vol.2.

Lingua, A., Marenchino, D., and Nex, F. (2009). Performance analysis of the sift operator for automatic feature extraction and matching in photogrammetric applications. *Sensors*, 9(5):3745–3766.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110.

Menegatti, E., Maeda, T., and Ishiguro, H. (2004). Image-based memory for robot navigation using properties of omnidirectional images. *Robotics and Autonomous Systems*, 47(4):251 – 267.

Milford, M., Wyeth, G., and Prasser, D. (2004). Simultaneous localisation and mapping from natural landmarks using ratslam. In *Australasian Conference on Robotics and Automation*, Canberra. Australian Robotics and Automation Association Inc.

Moravec, H. and Elfes, A. (1985). High resolution maps from wide angle sonar. In *Proc. IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 116 – 121.

Murillo, A., Guerrero, J., and Sagues, C. (2007). Surf features for efficient robot localization with omnidirectional images. In *IEEE Int. Conf. on Robotics and Automation*, pages 3901 –3907.

Murillo, A., Singh, G., Kosecka, J., and Guerrero, J. (2013). Localization in urban environments using a panoramic gist descriptor. *IEEE Transactions on Robotics*, 29(1):146–160.

Payá, L., Amorós, F., Fernández, L., and Reinoso, O. (2014). Performance of global-appearance descriptors in map building and localization using omnidirectional vision. *Sensors*, 14(2):3033–3064.