Rover localization in natural environments by indexing panoramic images

José-Joel Gonzalez-Barbosa and Simon Lacroix
LAAS-CNRS
7 Av. du Colonel Roche
31077 Toulouse Cedex 4 - France
{Jose-Joel.Gonzalez-Barbosa,Simon.Lacroix}@laas.fr

Abstract

In this paper, we present an approach to qualitative rover localization with panoramic images. The approach relies on the possibility to efficiently and robustly compute the resemblance between panoramic images, indexing them by histograms of local appearances. A database of image indexes is dynamically built during rover motions: when the rover re-perceives an already crossed area, it matches the current image with the stored ones (place recognition), and thus gets a qualitative estimate of its position. Experimental results on a 400 images database illustrates the algorithms throughout the paper.

1 Introduction

Among the various functionalities required for a rover to autonomously navigate in natural environments, localization is one of the most important. Indeed, a position estimate is required to build coherent environment models, to ensure that the given mission is successfully being achieved, or to control motions along a defined trajectory.

In natural environments, landmark-based position estimation techniques are quite challenging. The difficulty essentially lies in the conception of algorithms that extract and model the landmarks from the perceived data (either video or 3D): the environment being non structured, it is not easy to find salient, discriminative objects that can easily be modeled and matched from one perception to the other. Moreover, such techniques calls for *recognition abilities*: indeed, if landmarks can easily be matched between successive positions on the sole basis of their position and the rover motion estimate, it is necessary to *recognize* them when they are re-perceived after a while, when the initial rover position estimate is extremely unprecise, such as when performing a long loop trajectory for instance.

In this paper, we propose a qualitative *position refinement* technique that localize a rover when it comes back in a previously perceived area, using an image indexing technique on panoramic views of the environment to *recognize places*. The principle of our approach is the following: as it navigates, the rover continuously collects panoramic views of its environment, along with the current estimate of its position, and builds a database of *image indexes*, *i.e.* a set of characteristics computed on the images (learning phase). After a long traverse, when the rover arrives nearby an already crossed area, its position estimate is very likely to have drifted significantly: a newly perceived image is matched with the stored ones, and the best match gives a qualitative indication of the rover position (recognition phase). This position estimate can then subsequently enable landmark association or terrain model matching, and therefore be refined, which is out of the scope of this paper. The main advantage of our approach to place recognition is that it does not call for any landmark extraction procedure, thus avoiding the tricky data segmentation and landmark modeling steps. Moreover, it can be applied in *any kind* of environments (except some degenerate cases, *i.e.* purely texture free environments). Finally, our approach requires very few storage space and computation time.

The paper is organized as follows: the next section is a brief overview of image indexing techniques and of the use of panoramic images for localization purposes in robotics. Section 3 describes our approach to compute the resemblance between images. In section 4, we present how this approach is applied to panoramic images, and introduce a way to reduce both the database storage volume and the recognition computation time. Experimental results are provided along the paper.

2 Related work

2.1 Image indexing

Recent work in the vision community show that image indexing is a promising approach to object recognition. We can classify the various contributions related to image indexing in three great classes: the methods based on attributes computed on the overall images, changes of image space, and the methods that use local attributes.

Global attributes. Swain and Ballard [12] proposed to represent an object by a color histogram. Objects are identified by matching a color histogram from an image region with a color histogram from a sample of the object. The robustness to scale and orientation in their technique is mainly due to the use of color, while the robustness to changes in viewing angle and to partial occlusion are due to the use of histogram matching. However, the major drawback of the method is its sensitivity to lighting conditions. More recently, Schiele and Crowley proposed a technique to determine the identity of objects in a scene using matching of multi-dimensional receptive field histogram [10]. This technique can be used to determine the most probable objects in a scene, independently of the object position, image-plane orientation and scale.

Space transformation. Turk and Pentland in [14] developed a method for faces detection and recognition. Their PCA

method (Principals Components Analysis) computes a subspace **o**f the space of all possible images called "face space". The faces can be described as linear combinations of a small number of characteristic face-like images. The main advantage of this approach is the representation of each image by a small number of coefficients, which can be stored and retrieved efficiently. Even though very successful, these approaches are not applicable in our context: on one hand, any change of individual pixel values, caused for example by scale changes, image plane rotation or illumination changes, changes the eigenvector representation of an image. On the other hand, the PCA can not be incrementally performed: all the learning images are required to compute the subspace.

Local attributes. [11] is an examplary recognition method based on local attributes. It is based on the local gray-value invariants computed at automatically detected interest points. It allows an effective search in a database of more than 1000 images, thanks to a probabilistic model for recognition based on local descriptors and spatial relations between these descriptors. Experimental results show a correct recognition of the objects which can appear in complex scenes, even if they are partially visible or observed from different points of view.

2.2 Localization with panoramic images

Appearance-based localization method with panoramic images are presented in [5, 1, 7]. The number of images required to represent the environment is very large, and one therefore needs to compress this information. In these contributions, a low dimensional eigenspace representation is built using PCA. The problem of estimating the robot position consists in determining the reference image that best matches the current images in the low dimensional eigenspace. Other basic functions, such as Fourier transforms, have also been used to approximate the learning sets [4]. The actual localization phase can be described as the search for the closest match between the current view and the views (or their interpolation) from the learning set. [13] describes a system which learns places by automatically selecting landmarks from panoramic images and uses them for localization task. An adaptation of the biologically inspired "Turn Back and Look" behavior is used to evaluate potential landmarks. Normalized correlation is used to overcome the effects of illumination changes in the environment. Localization is simply a matter of matching sets of landmarks, and their associated place, to the current visual scene.

3 Indexing images with histograms

Object recognition with histograms is an attractive method, because of its simplicity, speed and robustness. We have developped such a technique using the local descriptions proposed by Schiele and Crowley in [10], but we represent the images by a probability density function of the local appearances, a much smaller structure.

3.1 Local characteristics

Some local characteristics of images can be obtained by filtering with a derivative function [11, 6, 10]. The local characteristics we use for statistic representation rely upon Gaussian derivatives. The Gaussian derivatives provide a basis for signal decomposition consisting in successive derivatives of the input signal. They are often employed in computer vision, as they offer several interesting properties:

- Genericity: [8] shows that the eigen vectors of a great number of images can be approximated by Gaussian derivatives.
- Robustness relatively to scale changes: in [11], it is shown that the Gaussian derivative are robust to scale changes up to roughly $\pm 20\%$.
- Equivariance to the scale: in [6], a standardization of Gaussian derivative is proposed, thanks to which it is possible to obtain a characteristics space invariant to the scale.
- Finally, a recursive implementation of the Gaussian derivatives is possible [15].

Let I be an image, the local characteristics of I are defined by:

$$J[I(x,y)] = \{L^{\sigma}_{n,\phi}(x,y) \in I \times \Re^{+}\}$$

in which $L^{\sigma}_{n,\phi}$ is the convolution of I with Gaussian derivatives $G^{\sigma}_{n,\phi}$, n being is the order of the derivative, ϕ its orientation, and σ defines the Gaussian function, that determines the quantity of smoothing. We describe the local characteristics using the following equation:

$$Z^{\sigma}_{n,\phi}(x,y) = log(1 + ||L^{\sigma}_{n,\phi}(x,y)||)$$

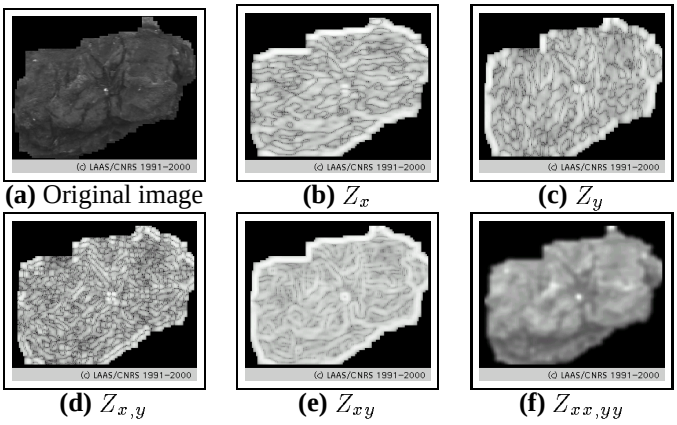
The interest of this equation is that it provides a measure of the magnitude of the difference between the neighboring pixels (the number 1 is used to eliminate abrupt variations when $0 < ||L_{i_1 \ldots i_n}|| < 1$).

Empirically, we determined that the following five equations represent enough information to compute image resemblance (figure 1):

$$\begin{aligned}
Z^{\sigma}_{x}(x,y) &= log(1 + ||L^{\sigma}_{x}(x,y)||) \\
Z^{\sigma}_{y}(x,y) &= log(1 + ||L^{\sigma}_{y}(x,y)||) \\
Z^{\sigma}_{x,y}(x,y) &= log(1 + \sqrt{L^{\sigma}_{x}(x,y)^2 + L^{\sigma}_{y}(x,y)^2}) \\
Z^{\sigma}_{xy}(x,y) &= log(1 + ||L^{\sigma}_{xy}(x,y)||) \\
Z^{\sigma}_{xx,yy}(x,y) &= log(1 + \sqrt{L^{\sigma}_{xx}(x,y)^2 + L^{\sigma}_{yy}(x,y)^2})
\end{aligned} \tag{1}$$

We represent the images by statistics on the local characteristics, parameterized by the robot position $\overrightarrow{p}$: $P(Z|\overrightarrow{p})$. These distributions being hardly modeled by parametric functions, we represent them by histograms:

$$H_{\overrightarrow{p}} = H(Z|\overrightarrow{p})$$



(a) Original image **(b)** Z_x **(c)** Z_y

(d) $Z_{x,y}$ **(e)** Z_{xy} **(f)** $Z_{xx,yy}$

Figure 1: *Original image* **(a)***, and local characteristics computed by equations 1.*

3.2 Comparing histograms

We define the distance between two images as the average of the distance between the 5 histograms that represent them:

$$D(I_1, I_2) = ||I_1 - I_2|| \equiv \frac{1}{5}\sum_{k=1}^{n} d(H_1^k, H_2^k)$$

There are several ways to compute such a distance:

Euclidean distance: it is defined as

$$||H_1 - H_2|| = \sqrt{\sum_i (\underline{H}_{1,i} - \underline{H}_{2,i})^2}$$

Representing histograms as 1-dimensional vectors, the notation $\underline{H}$ is used to denote the one-dimensional vector corresponding of H.

Histogram intersection: it has been studied in [12], and is defined by:

$$||H_1 - H_2|| = \sum_i min(\underline{H}_{1,i}, \underline{H}_{2,i})$$

where $\sum_i \underline{H}_{1,i} = \sum_i \underline{H}_{2,i} = 1$.

Quadratic distance: in [2], the authors claim that the inter-relationships between bins are also important in calculating histogram distances, which led them to the more general following distance measure:

$$||H_1 - H_2||^2 = (\underline{H}_1 - \underline{H}_2)^t A (\underline{H}_1 - \underline{H}_2)$$

where $A = [a_{ij}]$ is a $n \times n$ quadratic form. The matrix A is designed so that the equation calculates the local average of histogram bins, and therefore calculates a difference of 'smoothed' histograms. The matrix elements elements represent the similarity between the bins i and j, they can be normalized so that $0 \leq a_{ij} \leq 1$, with $a_{ii} = 1$.

Mahalanobis distance: it is the same definition as the quadratic distance, in which the matrix A corresponds to covariance matrix of the histograms learning set. Note the computation of this matrix required the knowledge of all the learning set: this distance is not applicable in our context.

Haussler distance: It has been studied in [3], and is defined by:

$$||H_1 - H_2|| = \sum_i \frac{|\underline{H}_{1,i} - \underline{H}_{2,i}|}{1 + \underline{H}_{1,i} + \underline{H}_{2,i}}$$

where $\underline{H}_1$ and $\underline{H}_2$ are not normalized.

χ^2 statistics: The χ^2 test is the formal statistical method used to determine how much two distributions are different. In [10], the authors introduce a modified χ^2:

$$||H_1 - H_2||^2 = \sum_i \frac{(\underline{H}_{1,i} - \underline{H}_{2,i})^2}{\underline{H}_{1,i} + \underline{H}_{2,i}}$$

The Earth Mover's Distance (EMD) : This distance is based on the minimization of the necessary cost to transform a distribution in an another one [9]. The cost is defined as a flow: it integrates the value of the f_{ij} quantities moves of a i^{th} bin histogram to j^{th} bin histogram, multiplied by the d_{ij} distance between the two bins. Once the global flow $\sum_I \sum_j d_{ij} f_{ij}$minimized, the distance is defined as follows:

$$D(H_1, H_2) = \frac{\sum_i \sum_j d_{ij} f_{ij}}{\sum_i \sum_j f_{ij}}$$

The minimization of the global flow is an iterative algorithm, which convergence time grows exponentially with the number of histograms bins[1].

Comparisons To evaluate these various distances, we used as a learning base of 573 images of 8 different natural rocks, as shown in figure 1. The images have been taken with a classical camera, the rock being on a rotating platform: to each image is associated an object identity, and a rotating angle. 39 test images have been taken and compared to each of the 573 images of the database, with the 7 distances considered above. Table 1 shows that the χ^2 statistics is the distance measure that gives the best results. With histograms containing 100 bins, this distance computation takes less than 0.2 ms on a Sun UltraSparc 5 station.

4 Indexing panoramic images

To evaluate the efficiency of the place recognition functionality using panoramic images, we drove the robot Lama (figure 2) in our experimental test site. Panoramic images were continuously acquired and saved, their position being measured by a differential phase GPS. Figure 3 show the various rover positions during a trajectory consisting in three loops. The image resolution is 768×576, and the mean distance between two consecutive images is about 0.4 meters.

[1] In practice, this time can be reduced according to the "difference" between the considered histograms

Distance	Complexity	Recognition rate (%)	Second candidate
Intersection	$O(n)$	46.1	7.7
Euclidean	$O(n)$	35.9	15.4
Quadratic	$O(n^2)$	41.0	15.4
Mahalanobis	$O(n^2)$	53.8	12.8
Haussler	$O(n)$	43.6	15.4
χ^2 statistics	$O(n)$	69.2	15.4
EMD	–	66.6	12.8

Table 1: *Comparison between the various histogram distances.*

Figure 2: *The robot Lama used for the experiment (the panoramic camera can not be seen here - it was mounted on top of the mast, near the white GPS antenna). The numbers indicating the position of some images.*

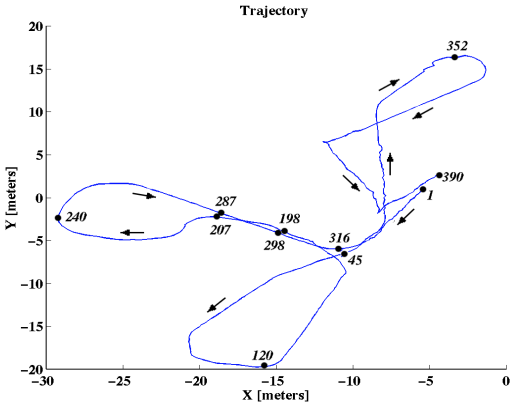


Figure 3: *The trajectory during which about 400 panoramic images were taken*

4.1 Image structuration

In order to determine both the location and orientation of a robot with omni-directional images, several authors proposed to sample the images: [1] samples the images according to a quadrangular grid, other approaches make a cylindrical transformation [5], followed by a discretization in columns [7]. With such samplings, the recognition algorithm is sensitive to the robot orientation. Instead, we propose to discretize the panoramic images in n rings (figure 4): the histogram representation of each ring is therefore invariant to the robot orientation. For each ring R_k, the five local characteristics defined in section 3 define a *histogram family* $F(I,k) = \{H(Z|R_k,\overrightarrow{p}), k = 1,\ldots,n\}$, where $H(Z|R_k,\overrightarrow{p}) = H^k_{\overrightarrow{p}}$) reflects the local characteristics properties of the ring R_k.

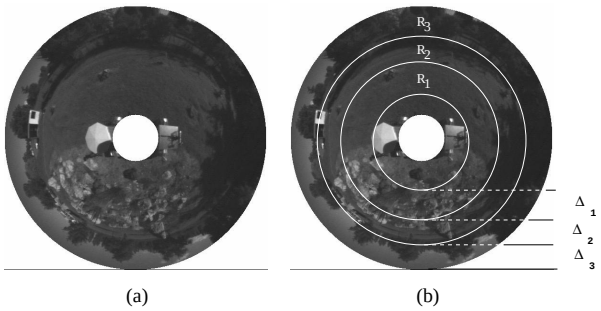


Figure 4: *A panoramic image* **(a)**, *and the corresponding three rings (b). The center ring, in which the rover wheels and GPS antenna is always present, is not considered.*

The advantage of discretizing the image in rings is to take into account the fact that the smallest ring, which corresponds to near areas, changes significantly with small robot motions, whereas the largest one, which corresponds to further areas, changes much less with the same motions. Figure 5 show the comparison between the distance computed when considering one histogram family defined over the whole image, with the distance resulting from the average of three rings: with three rings, the distance measure is much more discriminant.

4.2 Database building

During the learning phase, the rover continuously gathers panoramic images as it moves: it is worth to reduce the number of stored histograms, to reduce both the storage memory and the recognition computation time. A way to achieve this is to determine the less descriptive portion in the panoramic images [1], or to discard the images that can be approached by another: we chose this latter approach.

Figure 6 shows two examples of the evolution of the distance between the histograms family of one image with the following images during the motion (not surprisingly, the distance increases more rapidly for the smallest rings - the first one - than for the bigger rings). Given this behavior of the distance, the selection of the histograms to store (referred to as *key histograms*) is simply based on a threshold ξ. The procedure is as follows: all the histograms of the start image are selected as key histograms. During motion, the histograms of each new image are computed, and their distance to the last key histograms is computed: when this distance exceeds ξ, they are stored as key histograms. With this procedure, the 394 images corresponding to the trajectory of figure 3 generate 99 key histograms for the first ring, 84 for the second and 74 for the third, with an empirically chosen value $\xi = 0.6$. The database size is reduced to about a fourth, the smaller rings naturally generating more key histograms than the bigger.

4.3 Recognition

When the robot is asked to estimate its position after a long loop for instance, the recognition phase simply consists in

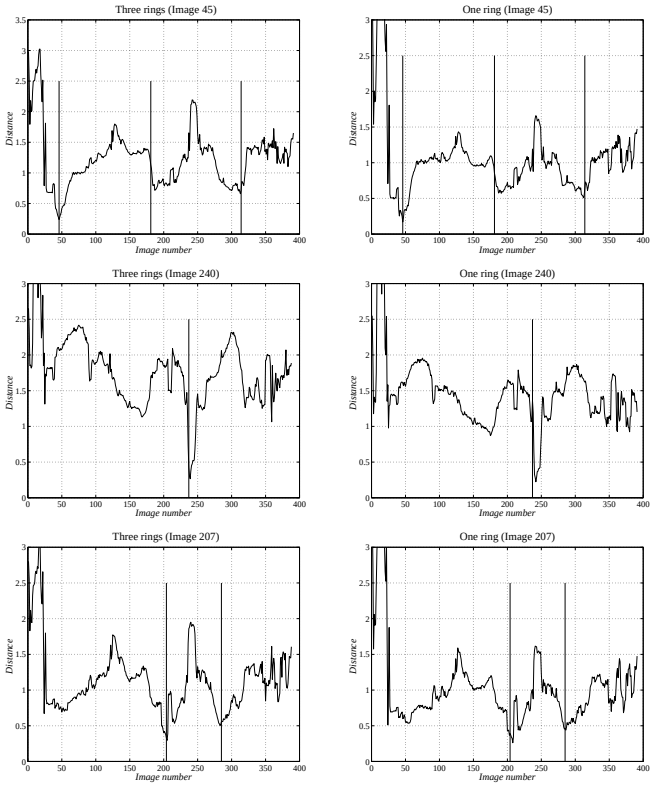


Figure 5: *Comparison between the similarities computed on the basis of three rings (left column), and on the basis of the whole image (right), for three sample images. The vertical lines show the closest images to the test image - refer to figure 3.*

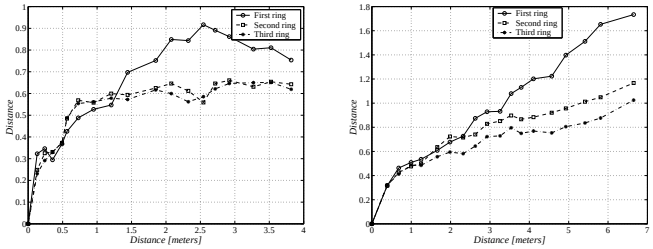


Figure 6: *Two examples of the evolution of the distance between the histograms family of an image with the following ones, as a function of the distances between the acquisition positions.*

comparing a newly perceived panoramic image with the histograms stored in the database. Figure 7 shows the distance the image 287 with *all* the stored key histograms (note that in a realistic context, this distance would have only been computed for a few key histograms around number 200, which is the area in which to rover is supposed to evolve).

At this stage, the rover knows the closest key histograms that match each ring of the current image. But the database reduction induces a higher discretization of the stored positions, and the closest key histograms do not necessarily correspond to the same position for the three rings. To have a more precise estimate of the rover position, during the learning phase,

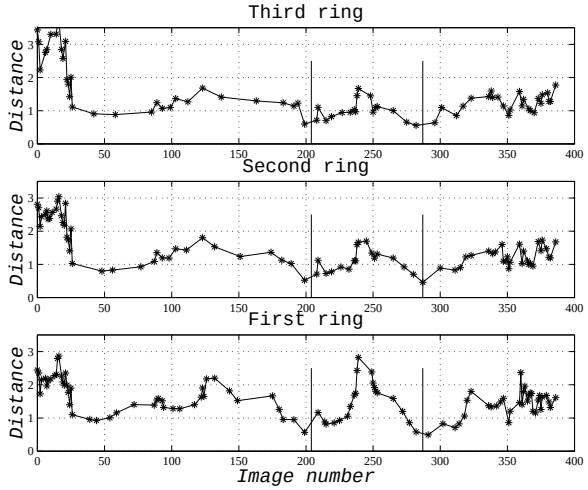


Figure 7: *Similarity between image 287 and all the key images. The interesting result for the localization purpose is here that this image matches an image closest to the image 200 (refer to figure 3). The computation time to compute all these distances is less than half a second.*

we associate to the position of each ring that is not selected as a key histogram the distances between the preceding and the following histograms. Let $\overrightarrow{p}$ the position of a ring that is not selected, and k^+ and k^- the two key histograms that surrounds it: the distances $d(\overrightarrow{p}, \overrightarrow{k^-})$ and $d(\overrightarrow{p}, \overrightarrow{k^+})$ (respectively corresponding to $||H_{\overrightarrow{p}} - H_{\overrightarrow{k^-}}||$ and $||H_{\overrightarrow{p}} - H_{\overrightarrow{k^+}}||$) are stored at the position $\overrightarrow{p}$.

When perceiving an image at a position $\overrightarrow{n}$, we have (figure 8):

$$d(\overrightarrow{n}, \overrightarrow{p}) \geq max(d(\overrightarrow{n}, \overrightarrow{k^-}) - d(\overrightarrow{p}, \overrightarrow{k^-}), d(\overrightarrow{n}, \overrightarrow{k^+}) - d(\overrightarrow{p}, \overrightarrow{k^+}))$$

$$d(\overrightarrow{n}, \overrightarrow{p}) \leq min(d(\overrightarrow{n}, \overrightarrow{k^+}) + d(\overrightarrow{p}, \overrightarrow{k^-}), d(\overrightarrow{n}, \overrightarrow{k^+}) + d(\overrightarrow{p}, \overrightarrow{k^+}))$$

In these equations, $d(\overrightarrow{n}, \overrightarrow{k^{+,-}})$ are computed during the recognition step, and $d(\overrightarrow{p}, \overrightarrow{k^{+,-}})$ are stored in the database. The distance $d(\overrightarrow{n}, \overrightarrow{p})$ is simply computed as the mean of the values of the right member of these equations. As a result, we are able to associate the perceived image n to an image of the database that has not been selected as a key one, thus refining the rover position estimate.

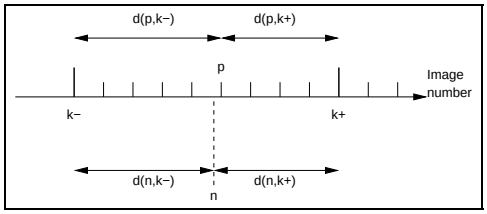


Figure 8: *Principle of the position refinement.*

Figure 9 compares the histogram distance between a test image and the *whole* database, and the histogram distance between the same image and the key histograms, using the ap-

proximation above. The histograms of errors due to the approximation shows that it is very precise.

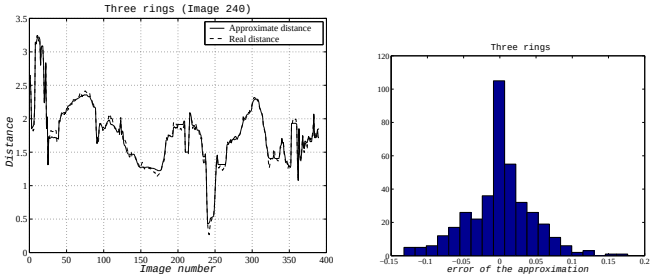


Figure 9: *Left: real and approximated distance between and image test (number 240 here). Right: histogram of the errors in the approximation.*

4.4 Finding the robot orientation

With the discretization in rings and the use of global attributes, the algorithm computes a resemblance between the images that does not depend on their relative orientation. Once the closest image is found in the learning base, a simple and efficient way to determine this rotation is to correlate a panoramic profile defined in the matched images. We define such a profile as a 360 dimensions vector, whose elements are given by the mean grey level value of the areas defined by the discretization shown in figure 10. The correlation score used is the zero-mean normalized cross-correlation score (ZNCC):

$$cor(\theta) = \frac{\sum_{i=1}^{360}(u_i - \overline{u})(v_j - \overline{v})}{\sqrt{\sum_{i=1}^{360}(u_i - \overline{u})^2}\sqrt{\sum_{i=1}^{360}(v_i - \overline{v})^2}} = \frac{\sigma_{u_i v_j}}{\sigma_u \sigma_v}$$

where $j = (i + \theta) mod(360)$. Figure 10 shows the correlation curves obtained with the image 293 and the three closest images found in the learning base with the histogram distance. The advantage of using a profile defined by the discretization (as opposed to a thinner 1-pixel width profile) is that the correct orientation can be retrieved, even if the images have not been precisely acquired at the same positon.

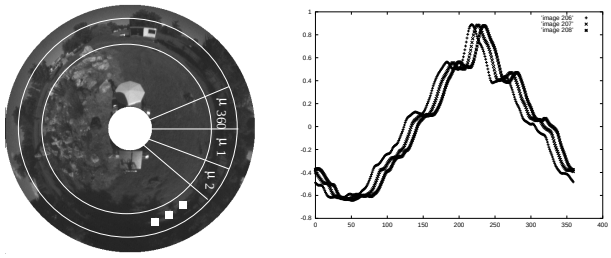


Figure 10: *Left: the image discretisation that defines a panoramic profile. Right: correlation curves between the image* 293 *and the images* 206 207 *and* 208.

5 Conclusion

We proposed an algorithm that determines a qualitative estimation of a rover position, by matching panoramic images. The image database is dynamically built while the robot moves: images are sampled in rings, local appearances characteristics are extracted and represented by histograms, and key images are selected. The reduction of the histogram database to about a fourth allows to save a lot of storage space and recognition time, while preserving the possibility to match images that have not been kept as key images.

The integration of the algorithm on board the robot Lama is currently under way. For that purpose, we will use higher resolution images (1280×1024), to define more than 3 rings. We plan to apply a discriminant analysis to both the 5 local apearances retained and the various rings, in order to have a more discriminant distance computation. Also, we will investigate the possibility to use color information in the histograms.

References

[1] J. Gaspar, E. Grossmann, and J. Santos-Victor. Information sampling for optimal image data selection. *9th International Symposium on Intelligent RoboticSystems, Toulouse (France)*, 2001.

[2] J. Hafner, H. Sawhney, W. Equitz amd M. Flickner, and W. Niblack. Efficient Color Histogram Indexing for Quadratic Distance Functions. *IEEE Trans. on Pattern Analysis and Machine Inteligence*, 17(7):729–736, 1995.

[3] Jing Huang, S Ravi Kumar, Mandar Mitra, Wei jing Zhu, and Ramin Zabih. Spatial Color Indexing and Aplications. *International Journal of Computer Vision*, 35(3):245–268, 1999.

[4] H. Ishiguro and S Tsuji. Image based memory of environment. *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 1996.

[5] M. Jogan and A.Leonardis. Robust localization using eigenspace of spinning-images. *5th Int. Conf. on Pattern Recognition*, 2000.

[6] Tony Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publisher, 1994.

[7] Lucas Paletta, Simone Frintrop, and Joachim Hertzberg. Robust localization using context in omnidirectional imaging. *International Conference on Robotics and Automation*, 2001.

[8] Rajesh P.N. Rao and Dana Ballard. Object Indexing using an Iconic Sparse Distributed Memory. In *International Conference on Computer Vision*, Jan 1995.

[9] Y. Rubner, C. Tomasi, and L. J. Guibas. A Metric for Distributions with Applications to Image Databases. *Proceedings of the 1998 IEEE International Conference on Computer Vision*, pages 56–66, 1998.

[10] Bernt Schiele and James L. Crowley. Recognition without Correspondance using Multidimensional receptive field histograms. *International Journal of Computer Vision*, 36(1):31–50, 2000.

[11] Cordelia Schmid. A Structured Probabilistic Model for Recognition. In *Computer Vision and Patern Recognition*, 1999.

[12] Michel J. Swain and Dana H. Ballard. Color Indexing. *International Journal of Computer Vision*, 7(1):11–32, 1991.

[13] Simon Thompson, Toshihiro Matsi, and Alexander Zelinsky. Localisation using automatically selected landmarks from panoramic images. *Proceedings of Australian Conference on Robotics and Automation, Melbourne Australia*, 2000.

[14] M. A. Turk and A. Pentland. Eigenfaces for Recognition. *Journal of Cognitive Neuroscience*, 3(1):59–70, 1991.

[15] Lucas J. van Vliet, Ian T. Young, and Piet W. Verbeek. Recursively Gaussian Derivative Filters. *Signal Procesing*, 44(2):139–151, 1995.