Research Article

Using Omnidirectional Vision to Create a Model of the Environment: A Comparative Evaluation of Global-Appearance Descriptors

L. Payá, O. Reinoso, Y. Berenguer, and D. Úbeda

Department of Systems Engineering and Automation, Miguel Hernández University, Avenida de la Universidad s/n, Elche, 03202 Alicante, Spain

Correspondence should be addressed to L. Payá; lpaya@umh.es

Received 23 October 2015; Accepted 11 February 2016

Academic Editor: Yassine Ruichek

Copyright © 2016 L. Payá et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, the design of fully autonomous mobile robots is a key discipline. Building a robust model of the unknown environment is an important ability the robot must develop. Using this model, this robot must be able to estimate its current position and to navigate to the target points. The use of omnidirectional vision sensors is usual to solve these tasks. When using this source of information, the robot must extract relevant information from the scenes both to build the model and to estimate its position. The possible frameworks include the classical approach of extracting and describing local features or working with the global appearance of the scenes, which has emerged as a conceptually simple and robust solution. While feature-based techniques have been extensively studied in the literature, appearance-based ones require a full comparative evaluation to reveal the performance of the existing methods and to tune correctly their parameters. This work carries out a comparative evaluation of four global-appearance techniques in map building tasks, using omnidirectional visual information as the only source of data from the environment.

1. Introduction

During the last years, the presence of mobile robots in both industrial and household environments has increased substantially since they are able to solve many different tasks. The expansion of robots into such environments and applications has been eased thanks to the development of their abilities in perception, computation, autonomy, and adaptability to different circumstances. As far as perception is concerned, the robots must be equipped with sensors that allow them to extract the necessary information from the environment to be able to carry out autonomously their tasks. Vision sensors have gained popularity because they present some interesting advantages such as providing a big quantity of information with a relatively low cost and low power consumption (comparing to other sensors, such as laser rangefinders) and stable data both outdoors and indoors (unlike GPS, whose signal is prone to degradation indoors). They also permit carrying out additional high level tasks, such as people detection and recognition. Among vision sensors,

catadioptric systems have extended in recent years as they are able to capture images with a field of view of 360 deg. around the robot [1]. In our approach, the mobile robot is equipped with a catadioptric system on it, which captures images from the environment. Using this information, the objective is building a robust model of the environment. In general, these models can be represented as a metric, a topological, or a hybrid map [2]. First, metric maps define the position of some relevant features of the environment with respect to a coordinate system and permit robot localization with geometric accuracy (except for a relative error) [3]. Second, topological maps often represent the environment as a graph where nodes are distinctive localizations (e.g., rooms) of the environment and links are the connectivity relationships between localizations. Usually, such maps do not permit fine localization but they are enough to estimate the position of the robot and to navigate to the desired localizations [4]. At last, hybrid maps are hierarchical models where information is arranged in multiple levels. Usually, there is a high level of topological information that allows an

approximate localization (in an area of the environment) and several low levels of metric information that permit refining the localization (in the previously detected area) [5].

In all cases, to build a functional map, it is necessary to extract relevant information from the scenes. Traditionally, researchers have focused on methods that extract some outstanding landmarks or interest points from the scenes and describe them with any robust description method. These methods have become popular in map building and mobile robots localization. For example, Angeli et al. [6] make use of SIFT features [7] to solve the mapping and global localization problems simultaneously (SLAM) and Valgren and Lilienthal [8] and Murillo et al. [9] make use of SURF features [10] to solve the localization problem in a previously created model. Using feature-based approaches in combination with probabilistic techniques, it is possible to build metric maps [3]. However, these methods present some drawbacks; for example, it is necessary that the environment be rich in prominent details (otherwise, artificial landmarks can be inserted in the environment, but this is not always possible); also, the detection of such points is sometimes not robust against changes in the environment and their description is not always invariant to changes in robot position and orientation. Besides, camera calibration is crucial in order to incorporate new measures in the model correctly. This way, small deviations in either the intrinsic or the extrinsic parameters add some error to the measures. At last, extracting, describing, and comparing landmarks are computationally complex processes that often make building the model in real time unfeasible, as the robot explores the environment.

In contrast, global-appearance techniques have gained relevance in more recent works [11–13]. These techniques are useful when the robot moves within unstructured environments where extracting and describing robust points is difficult. These approaches lead to conceptually simpler algorithms since each scene is described by means of a unique descriptor. Map creation and localization can be achieved just storing and comparing pairwise these descriptors. As a drawback, extracting metric relationships from this information is difficult; thus, this family of techniques is usually employed to build topological maps (unless the visual information is combined with other sensory data, such as odometry). Despite their simplicity, several difficulties must be faced when using these techniques. Since no local information is extracted from the scenes, it is necessary to use any compression and description method that make the process computationally feasible. These descriptors do not present invariance to changes neither in the robot orientation nor in the lighting conditions or other changes in the environment (position of objects, doors, etc.). They will also suffer problems in environments where *visual aliasing* is present, which is a common phenomenon in indoor environments with repetitive visual structures.

Many algorithms can be found in the literature working both with local features and with global appearance of images. All these algorithms imply many parameters that have to be correctly tuned so that the mapping and localization processes are correct. Feature-based approaches have reached a relative maturity and some comparative evaluations have

been carried out, such as [14]. These evaluations are useful to choose the most suitable extractor and descriptor to a specific application. However, global-appearance-based approaches are still a field that is worth deeper exploration. We have not found any work that makes a comparative evaluation of the performance of such descriptors in mapping tasks. This is the main objective we propose in this paper. We have selected four accepted global-appearance description methods, adapted them to be used with omnidirectional visual information, and studied their properties. Then, we have developed the necessary algorithms to create a model of the environment, tested their performance, and studied the influence of the most relevant parameters.

The remainder of the paper is structured as follows. Section 2 presents briefly the description approaches that will be evaluated along the paper. After that, Section 3 describes the kind of models of the environment we will build to test the performance of the approaches. Then, Section 4 details the set of experiments designed and the results obtained. To finish, a final discussion is carried out in Section 5.

2. Global-Appearance Descriptors: State of the Art

This section outlines some methods to describe the global appearance of images. Four families of methods are proposed to be analysed: methods based on the discrete Fourier transform (Section 2.1), on principal components analysis (Section 2.2), on orientation gradients (Section 2.3), and on the essence of the scenes (Section 2.4). These are the description methods whose performance will be evaluated along the paper.

2.1. Methods Based on the Discrete Fourier Transform (DFT).

The Discrete Fourier Transform (DFT) is a classical method to describe scenes that presents some interesting features. When the two-dimensional DFT of a scene $im(x, y)$ is calculated, the result is a complex function $IM(u, v)$ in the frequency domain (u and v are the frequency variables) that can be decomposed into magnitude and argument matrices. The first matrix (also known as amplitude spectrum) represents the distribution of spatial frequencies within the image (i.e., it contains information on the overall structure of the image: edges orientation, smoothness, width, etc.). On the other hand, the argument matrix contains information about the local properties of the scene (shape and position of the objects). Taking these facts into account, the amplitude spectrum can be used as a global descriptor of the scene, since it contains information about the dominant structural patterns and it is invariant to the distribution of the objects. This information has proved to be relevant to solve simple classification tasks [15]. However, this kind of descriptors has no information about the spatial relationships between the structures in the image. To have a complete description of the scene, such information must be included.

Considering it, we have opted for a formulation of the DFT which contains complete information. This formulation is the Fourier Signature (FS), described first in [12]. It is defined as the matrix composed of the 1D DFT of each row

in the original image. When applied to panoramic scenes, it offers rotational invariance. When we calculate the FS of a panoramic image $\text{im}(x, y) \in \mathbb{R}^{N_x \times N_y}$, we arrive to a new matrix $\text{IM}(u, y) \in \mathbb{C}^{N_x \times N_y}$, where the main information is concentrated in the low frequency components of each row (so we can retain only the k first columns, having a compression effect). This new matrix with N_x rows and k columns can be decomposed in a magnitude matrix $A_j(u, y) = |d_j(u, y)|$ with N_x rows and k_1 columns and an argument matrix $\Phi_j(u, y)$, with N_x rows and k_2 columns.

Based on the shift property of the DFT, when two panoramic images have been captured from the same position but have the robot different orientations, both images have the same magnitude matrix and the arguments matrices permit obtaining the relative robot orientation. This property allows us to use the magnitude matrix to estimate the position of the robot (as it presents rotational invariance) and, then, the arguments matrix to estimate the relative orientation of the robot.

2.2. Methods Based on Principal Components Analysis (PCA). Panoramic images are data that fall in a space with a very high number of dimensions. However, the image pixels tend to be very correlated data, since they have been captured from a 3DOF process (robot pose on the ground plane). Taking this fact into account, a natural way to compress the information is principal components analysis (PCA), as shown in [16]. This kind of descriptors has evolved from the original formulation to adapt them to be used in mapping and localization tasks. The works of Leonardis and Bischof [17] show some examples of how this analysis can be used to mobile robots localization in a robust way.

When we have a set of n images $\text{im}_j(x, y) \in \mathbb{R}^{N_x \times N_y}$, $j = 1, \dots, n$, each image can be considered a point in a space with $N_x \cdot N_y$ dimensions, $\tilde{x}_j(i) \in \mathbb{R}^{N_x \cdot N_y \times 1}$, $j = 1, \dots, n$, ($n \ll N_x \cdot N_y$). Using the classical formulation of PCA, it is possible to transform each point $\tilde{x}_j(i)$, in a new data point, namely, *image projection* $\tilde{p}_j(i) \in \mathbb{R}^{k_3 \times 1}$ $j = 1, \dots, n$, where k_3 is the number of PCA features that contain the most relevant information, $k_3 \leq n$. Turk and Pentland [18] show how the necessary transformation matrix $\mathbf{V}$ can be obtained in an efficient way. They make use of the SVD of the data matrix covariance, retaining only the eigenvectors with higher eigenvalues. If the number of eigenvectors is equal to n , then there is no loss of information during the compression process [16]. Thus, after applying PCA techniques, images can be handled efficiently, with a low computational cost. However, depending on the images' size, the process to obtain $\mathbf{V}$ may be substantially slow.

The use of PCA in mapping and localization tasks is limited since the image projections depend on the robot orientation. Independently, on using omnidirectional scenes, the images projections contain only information of the position and orientation the robot had when capturing the images. This is the reason why Jogan and Leonardis developed the concept of *eigenspace of spinning images* [19]. This model uses specific properties of panoramic images to obtain, in an efficient way, an optimal subspace that takes into account the different orientations a robot may have

when capturing each image. The method takes profit of the symmetry properties the data matrix presents when we add the rotations information. This method has the advantage of permitting the estimation of the robot orientation, but the computational cost to obtain the transformation matrix $\mathbf{V}$ is extremely high. By this reason, it has been only used with small environments, with a limited number of images.

2.3. Methods Based on the Histogram of Oriented Gradients (HOG). HOG is a description method used traditionally in object detection. This technique considers the gradient orientation in localized parts of a scene. The method outstands by its simplicity, good computational cost, and relatively good results in object recognition tasks. It was initially described by Dalal and Triggs [20], who used it in people detection tasks. Later on, some researchers developed an improved version of the algorithm both in detection accuracy and in computational cost [21].

However, the experience with HOG descriptors in the mobile robotics field is limited to simple and small environments. Few previous works have made use of HOG in robot mapping and localization. Hofmeister et al. [22] use this descriptor in small robots localization tasks, with low resolution images and small environments not prone to *visual aliasing*. Under these limited conditions, the algorithm works well.

HOG is not defined as a global-appearance descriptor because the basic implementation consists in dividing the scene in a set of cells and obtaining a histogram of gradient orientation using the pixels information in each cell. The combination of all these histograms is the image descriptor. We have redefined the algorithm to obtain a unique descriptor per image that contains information of the global appearance of this image. The version of HOG we consider is described in [23], where a global version of HOG is used to carry out mapping and Monte-Carlo localization in large environments. Anyway, it is necessary to make an evaluation of the performance of this algorithm and systematize it in map creation tasks.

2.4. Methods Based on Gist and Prominence. The *gist* concept was first introduced by Oliva and Torralba [24], with the idea of creating a low-dimension scene descriptor, and avoiding segmentation and processing of points, objects, or individual regions. They inspired by some works that suggested that humans recognize scenes by codifying the global configuration and just ignoring most of the details and individual objects [25].

More recently, some works make use of the *prominence* concept together with *gist*. It refers to regions of pixels that stand out with respect to the neighbor regions, in contrast to *gist*, which implies the accumulation of statistical data from the whole image. Siagian and Itti [26] try to establish a synergy between the two concepts and they design a unique descriptor that takes both into account. This descriptor is built using the intensity, orientation and color information.

The experience with this kind of descriptors in mobile robots applications is limited. For example, Chang et al. [27] present a localization and navigation system based on *gist* and *prominence* and Murillo et al. [28] make use of *gist*

descriptors in a localization problem. However, they obtain these descriptors using specific regions in a set of panoramic images.

Like HOG, *gist* is not primarily defined as a global-appearance descriptor and we have redefined the algorithm to obtain a unique descriptor per image. The version of *gist* we consider in this evaluation is described in [23] and is built from orientation information, analysed in some resolution levels.

3. Creating a Visual Topological Map of the Environment

In this section we focus on the map creation problem. The robot, which is equipped with a catadioptric vision system on its top, explores the environment to map to cover it completely. During this process, the robot captures a set of omnidirectional scenes from several positions. Only this visual information will be used to build the map (neither odometry nor laser or other sensory data will be used). This way, the final model will be a topological map since it contains some localizations (represented as panoramic scenes) and connectivity relations, but no metric data. In Section 3.1, we describe how the nodes of the map are represented with each description method and, in Section 3.2, the process to add connections between the nodes is outlined.

3.1. Using Global-Appearance Descriptors to Create a Model of the Environment. Let us suppose that the mobile robot has gone across the environment to map (either in a teleoperated way or autonomously, following any exploration algorithm) and has captured a set of omnidirectional images $I = \{im_1, im_2, \dots, im_n\}$, where $im_j \in \mathbb{R}^{N_x \times N_y}$.

From this set of images, a set of descriptors, one per original scene, is calculated. As a result, the nodes of the map will be a set of descriptors $D = \{d_1, d_2, \dots, d_n\}$ where, in general, $d_j \in \mathbb{C}^{M_x \times M_y}$. With the objective that these nodes are functional, it is necessary that d_i contains information that permits estimating the position of the robot when capturing im_i (taking into account that the robot may have any orientation in this position). In the next subsections, we detail the kind of information each d_i should contain when using each description method.

3.1.1. DFT Descriptor. Each node d_i contains two matrices: the magnitudes one $A_i(u, y) \in \mathbb{R}^{N_x \times k_1}$ and the arguments matrix $\Phi_i(u, y) \in \mathbb{R}^{N_x \times k_2}$. k_1 is the number of columns we retain in the localization descriptor and k_2 is the number of columns retained in the orientation descriptor. The higher k_1 and k_2 , the more information the descriptor contains. However, we must take into account that the main information is concentrated in the low frequency columns, and if noise is present on the image, it will affect high frequency components mostly; thus, removing these components may imply an additional benefit. The effect of both parameters in a mapping process will be evaluated.

3.1.2. PCA Descriptor. The PCA descriptor we use is proposed in the works of Jogan and Leonardis [19]. This model uses

the specific properties of panoramic images to create a set of N_R spinning images from each of the n original panoramic images, so we get N_R data vectors per original image. To obtain the transformation matrix V , the similarities among the rotated versions of each image are taken into account. This permits decomposing the original problem (which is computationally very heavy) in a set of lower order problems.

As a result of the process, the map will be composed of (a) a set of descriptors $\tilde{p}_j(i) \in \mathbb{R}^{k_3 \times 1}$ $j = 1, \dots, n$, which are the projections of the original panoramic images and contain information on the robot position, (b) a set of phase vectors, $\tilde{\phi}_j \in \mathbb{R}^{k_3 \times 1}$, one per image, which contain information of the robot orientation, and (c) a unique transformation matrix $V \in \mathbb{C}^{k_3 \times N_x \cdot N_y}$. k_3 is the number of eigenvector chosen. The higher the k_3 , the more the information that the map contains. If $k_3 = n$, there is no loss of information.

3.1.3. HOG Descriptor. Each image will be described through two HOG descriptors. The first one, $\vec{h}_1$, is the position descriptor and is invariant against rotations of the robot. To obtain it, the panoramic image is divided into horizontal cells, whose width is equal to N_y (number of columns in the image) and whose height can be configured freely. The size of $\vec{h}_1$ is $1 \times k_4 \cdot b$, where k_4 is the number of horizontal cells and b is the number of bins in each orientation histogram. The second one, $\vec{h}_2$, is the orientation descriptor. To obtain it, the panoramic image is divided into vertical cells whose height is equal to N_x . Some overlap between these cells may exist. If the width of the cells is l_1 and the distance between consecutive cells d_1 , then the number of vertical cells is $k_5 = N_y/d_1$. The size of the orientation descriptor $\vec{h}_2$ is then $1 \times k_5 \cdot b$. In the experiments, the influence of k_4 , k_5 , and b will be evaluated.

Figure 1 shows, from a panoramic image whose gradient has been calculated, the process to obtain both descriptors: (a) $\vec{h}_1$ and (b) $\vec{h}_2$.

3.1.4. Gist and Prominence Descriptor. The information of the orientation of the edges in the image is used to build the descriptor. First, two versions of each image are considered: the original one and a new version after applying a Gaussian low-pass filter and subsampling to a new size $0.5N_x \times 0.5N_y$. Second, both images are filtered with a bank of m Gabor filters with orientations evenly distributed between 0 and 180 deg. Third, to reduce the amount of information, the pixels in each resulting image are grouped into blocks. The block division is carried out in a similar fashion as in HOG: a position descriptor $\vec{g}_1$ is obtained by defining k_6 horizontal blocks and an orientation descriptor $\vec{g}_2$ is calculated with k_7 vertical blocks (with overlapping). In the experiments, the influence of k_6 , k_7 , and m will be evaluated. Figure 2 shows, from a panoramic image, the process to obtain $\vec{g}_1$.

To sum up, Table 1 shows the parameters to be tuned in each description method included in the evaluation. On the other hand, Table 2 gives details of the contents of the map when we consider each description method.

FIGURE 1: Process to obtain (a) the HOG position descriptor $\vec{h}_1$ and (b) the HOG orientation descriptor $\vec{h}_2$.

3.2. Adding Topological Relations. Our starting point is a set of images captured from unknown positions. The objective of this section consists in designing an algorithm that allows us to establish adjacency relations among them, with the goal of creating a topological map. Apart from this, we expect the distribution of the nodes in this map to be similar to the distribution of the points where the images were captured. It goes beyond the classical concept of topological map since besides adjacency it also introduces the concepts of closeness and farness. Thanks to this kind of maps, the robot will be able to plan its trajectory more accurately.

To create such a map, a method based on a mechanical system of forces is used. This kind of methods has been used often to simulate the movement of flexible bodies, as in [29],

where the body is discretized into a set of particles, and the interaction among them is modelled with a set of springs. Our framework also includes a set of dampers in parallel with the springs, since the dampers can help to achieve an overdamped behaviour that facilitates reaching the steady state.

The idea we develop consists in considering each image a particle which is linked to the rest of images (particles) through a pair spring-damper, where the natural length of each spring is equal to the distance between the descriptors of the two images linked by this spring. The particles start their evolution from random positions. If we let the forces produced by springs and dampers move freely in the system until it tends to a minimum energy position, we expect the distribution of particles to be similar to the distribution

FIGURE 2: Process to build the gist position descriptor $\vec{g}_1$ ($m = 4$ orientations, $k_6 = 8$ blocks).

of capture points. The algorithm we use is inspired by the algorithm presented by Menegatti et al. [12], who used it in small environments.

3.2.1. Mass-Spring-Damper Method. Each image is considered a particle P_j , $j = 1, \dots, n$, with mass m_j , where n is the number of images to include in the map. No information about the coordinates of the capture points is available.

Each pair of particles P_k and P_l is linked with a spring S_{kl} with elastic constant k_{kl} and a damper with damping constant κ_{kl} . The natural length of each spring l_{kl}^0 is equal to the distance between the descriptors of the images associated with the particles P_k and P_l .

The initial positions of the particles are randomly initialised. After that, the system is allowed to evolve freely until it reaches a steady state. At this state, the distribution of the particles is expected to be similar to the distribution of capture points (except for a scale factor and a rotation). This way, the result is a scaled model of the real distribution. We consider the value of the elastic constants to be proportional to the distance between images and, from a threshold distance, the images are not linked by any spring.

Under these circumstances, the spring and damper linking each pair of particles P_i and P_j make on these particles the force:

$$\begin{aligned} \vec{f}_{ij} = & - \left[-k_{ij} \cdot (|\vec{p}_i - \vec{p}_j| - l_{ij}^0) + \kappa_{ij} \right. \\ & \cdot \left(\frac{(\vec{v}_i - \vec{v}_j) \cdot (\vec{p}_i - \vec{p}_j)}{|\vec{p}_i - \vec{p}_j|} \right) \left. \right] \cdot \frac{(\vec{p}_i - \vec{p}_j)}{|\vec{p}_i - \vec{p}_j|}, \quad (1) \\ \vec{f}_{ji} = & -\vec{f}_{ij}, \end{aligned}$$

where $\vec{p}_i, \vec{v}_i$ are the position and speed of the i th particle, respectively. Then, the resulting force on each particle is obtained:

$$\vec{F}_i = \sum_{j=\{1, \dots, n\}, j \neq i} \vec{f}_{ij}. \quad (2)$$

TABLE 1: Parameters to be tuned in each description method.

Descriptor	Parameters
FS	$k_1 \Rightarrow$ components per row, localization descriptor $\mathbf{A}_j$
	$k_2 \Rightarrow$ componentes per row, orientation descriptor Φ_j
r-PCA	$k_3 \Rightarrow$ number of eigenvectors
	$N_R \Rightarrow$ number of rotations per panoramic image
HOG	$b \Rightarrow$ number of bins per histogram
	$k_4 \Rightarrow$ number of horizontal cells, localization descriptor $\tilde{h}_1$
	$l_1 \Rightarrow$ width of vertical cells, orientation descriptor $\tilde{h}_2$
	$d_1 \Rightarrow$ distance between vertical cells, orientation descriptor $\tilde{h}_2$
	$k_5 = N_y/d_1 \Rightarrow$ number vertical cells, orientation descriptor $\tilde{h}_2$
Gist	$m \Rightarrow$ number of orientations (Gabor filters), localization descriptor $\tilde{g}_1$
	$k_6 \Rightarrow$ number of horizontal blocks, localization descriptor $\tilde{g}_1$
	$l_2 \Rightarrow$ width of vertical blocks, orientation descriptor $\tilde{g}_2$
	$d_2 \Rightarrow$ distance between vertical blocks, orientation descriptor $\tilde{g}_2$
	$k_7 = N_y/d_2 \Rightarrow$ number of vertical blocks, orientation descriptor $\tilde{g}_2$

TABLE 2: Contents of the map, relative to localization and orientation estimation, per image included in the model im_j , $j = 1, \dots, n$.

Descriptor	Localization	Orientation
FS	$\mathbf{A}_j \in \mathbb{R}^{N_x \times k_1}$	$\Phi_j \in \mathbb{R}^{N_x \times k_2}$
r-PCA	$\mathbf{V} \in \mathbb{C}^{k_3 \times N_x \times N_y}$ $\tilde{p}_j \in \mathbb{C}^{k_3 \times 1}$	$\tilde{\phi}_j \in \mathbb{R}^{k_3 \times 1}$
HOG	$\tilde{h}_{1j} \in \mathbb{R}^{k_4 \times b \times 1}$	$\tilde{h}_{2j} \in \mathbb{R}^{k_5 \times b \times 1}$
Gist	$\tilde{g}_{1j} \in \mathbb{R}^{2 \cdot k_6 \cdot m \times 1}$	$\tilde{g}_{2j} \in \mathbb{R}^{k_7 \cdot m \times 1}$

From this resulting force, the acceleration of the particle is obtained from the 2nd Newton's law:

$$\tilde{a}_i(t) = \frac{\tilde{F}_i}{m_i}, \quad (3)$$

where $\tilde{a}_i(t)$ is the i th particle acceleration at time instant t , $\tilde{F}_i$ is the resulting force on particle i , and m_i is the i th particle mass. From this acceleration, the speed and position of particle i once it passed a period of time Δt can be calculated:

$$\begin{aligned} \tilde{v}_i(t + \Delta t) &= \tilde{v}_i(t) + \tilde{a}_i(t) \cdot \Delta t, \\ \tilde{p}_i(t + \Delta t) &= \tilde{p}_i(t) + \tilde{v}_i(t) \cdot \Delta t. \end{aligned} \quad (4)$$

This method, known as Euler integration, may not be stable if the step time is not low enough, which would increase the computational cost of the process. This is the reason why the Verlet integration is sometimes suggested. In this

integration method, the position and speed are updated at each iteration with the following expressions.

At the time instant $t = \Delta t$,

$$\begin{aligned} \tilde{p}_i(\Delta t) &= \tilde{p}_i(0) + \tilde{v}_i(0) \cdot \Delta t + 0.5 \cdot \tilde{a}_i(0) \cdot \Delta t^2, \\ \tilde{v}_i(\Delta t) &= \tilde{v}_i(0) + \tilde{a}_i(0) \cdot \Delta t. \end{aligned} \quad (5)$$

From this time instant,

$$\begin{aligned} \tilde{p}_i(t + \Delta t) &= 2 \cdot \tilde{p}_i(t) - \tilde{p}_i(t - \Delta t) + \tilde{a}_i(t) \cdot \Delta t^2, \\ \tilde{v}_i(t + \Delta t) &= \frac{\tilde{p}_i(t + \Delta t) - \tilde{p}_i(t)}{2 \cdot \Delta t}. \end{aligned} \quad (6)$$

4. Experiments

In this section, we compare the performance of the four description methods. First, we describe the sets of images we have used to carry out the experiments. Then, the evaluation is carried out from several points of view to fully uncover the goodness of each method in mapping tasks. We analyse the computational cost of the mapping process, the relationship between the image distance and the geometric distance, and the performance in topological map building.

4.1. Sets of Images. To carry out the experiments, we make use of two sets of images, captured with two different catadioptric systems. First, set 1 has been captured by us in a building of Miguel Hernández University (Spain). The images were captured along 6 different rooms in an office-like environment. Figure 3(a) shows a bird's eye view of this environment. The database is composed of 873 panoramic 64×256 -color images which have been captured on a dense 40×40 cm grid of points (red points in Figure 3(a)). Set 1 [30] is a challenging database due to the tendency to visual aliasing that presents the environment. There are many zones which, despite being geometrically far, present a similar visual appearance. Also, the images were captured in different times of day (changing lighting conditions) and the positions of some objects in the scenes are modified (e.g., changes in the state of doors). All the images were captured with an *Imaging Source DFK 21BF04* camera mounted on a *Pioneer 3-AT* robotic platform. The camera takes pictures of a hyperbolic mirror (*Eizoh Wide 70*) which is mounted on it with its axis aligned with the camera optic axis. The resulting omnidirectional images are transformed with a cylindrical projection to obtain their panoramic versions. The *P3-AT* robot has 4 drive wheels. Its maximum linear speed is equal to 0.7 m/s, its maximum turning speed is equal to 140 deg/s, and the minimum turning radius is null. The robot can move freely on the floor so the image capture process has 3 degrees of freedom: position on the ground floor with respect to a world coordinate system (x, y) and orientation with respect to the z -axis (θ) . Figure 3(b) shows the robot, the catadioptric system, and a sample image (omnidirectional and panoramic formats).

The second set of images has been captured by a third party [31]. It is composed of a set of panoramic grayscale images, captured in several rooms of a university and a

FIGURE 3: (a) Bird's eye view of the environment where set 1 was captured. (b) Catadioptric system mounted on the robot and sample scene captured in the corridor (omnidirectional and panoramic formats).

TABLE 3: Images set 2: rooms considered in the experiments and main parameters.

Room	Number of images	Grid	Size	Resolution
Laboratory	170	30×30 cm.	3×5 m	81×561 pixels
Corridor	200	50×50 cm.	5×10 m	81×561 pixels
Kitchen	108	10×10 cm.	1.2×1.0 m	81×583 pixels
Hall	242	10×10 cm.	2.2×1.2 m	81×583 pixels

flat. They were captured with a camera *ImagingSource DFK 4303* mounted on the robot *ActivMedia Pioneer 3-DX*. The hyperbolic mirror is the model *Accowle Wide View*. This is an interesting database because it presents different grid sizes in each room. It permits testing how this parameter influences the performance of the methods. Table 3 shows the rooms we have used and the main features of the images.

4.2. Computational Cost. Previously, Section 3.1 has outlined the contents of the map nodes. Now, the objective of this section consists in making a comparative evaluation of the

computational cost of the four description methods during the creation of the map nodes. This study will be carried out depending on the value of the most relevant parameters of each description method. Data set 1 is used to carry out this comparative evaluation. This is an interesting study as it allows us to know which algorithms could work in real time.

First, Figure 4 shows the computation time using (a) FS versus k_1 and k_2 , and (b) rotational PCA versus N_R . Second, Figure 5 shows the time when using HOG versus k_4 , l_1 , and d_1 . At last, Figure 6 shows *gist* with $m = 8$ versus k_6 , l_2 , and d_2 . In all cases, the time per image is depicted. The total time to build the map can be obtained by multiplying by 873 (number of images in set 1).

In the case of FS, as k_1 and k_2 increase, the time increases slightly. The cost to obtain the DFT of each row is the same, the difference is in the need of computing the magnitude and argument of a different number of components, which implies a low computational cost. In any case, the computational cost of FS is very low.

As far as rotational PCA is concerned, Figure 4 shows how the time increases exponentially as N_R does, arriving at up to 110 seconds per image (27 hours to build the whole

FIGURE 4: Computational cost to obtain the nodes' descriptors using (a) Fourier Signature and (b) rotational PCA.

FIGURE 5: Computational cost to obtain the nodes' descriptors using HOG.

FIGURE 6: Computational cost to obtain the nodes' descriptors using *gist*.

map), when $N_R = 32$ rotations. It has been impossible to consider a higher number of rotations due to the enormous requirements of memory during the process.

If we analyse now HOG, on the one hand, the influence of k_4 is low and, on the other hand, time increases linearly when l_1 does. At last, when d_1 increases, the time decreases as fewer vertical cells are considered. In general, HOG presents a substantially higher computational cost compared to FS; despite it, the algorithm is quick enough to permit carrying out the mapping process in real time, as the robot explores the unknown environment.

At last, the computational cost of *gist* is, in general, approximately 10 times the cost of FS and similar to HOG. All of FS, HOG, and *gist* are computationally feasible algorithms. Nevertheless, rotational PCA could only be used if the mapping process is allowed to be done offline. Also, the maximum number of rotations included in the map has been $N_R = 32$. This means that the resolution in orientation estimation will be low. Anyway, even though the computational cost of rotational PCA had been low enough, this algorithm

would not have permitted building maps online since all the training images must be available to start the process (unless any incremental PCA algorithm is used [32], which would add more computational cost to the process and make it unbearable in real time). The other three algorithms do not present this disadvantage since they are inherently incremental methods (each image is described independently on the rest of images so the robot can build the map as it is exploring the unknown environment).

4.3. Image Distance versus Geometric Distance. Once we know the computational cost of the description methods, the objective of this section consists in carrying out several experiments to test the applicability of these methods to the creation of topological maps.

The first experiment consists in studying the relationship between the geometrical distance between the positions where two images have been captured and the distance between the descriptors of these two images. The behaviour of this distance should be monotonically increasing and

linear, at least in a close interval around the point where the reference image was captured.

To carry out this study, several distance measures are taken into consideration. First, these distances are formalized. If we have two descriptors $\vec{r} \in \mathbb{R}^{l \times 1}$ and $\vec{s} \in \mathbb{R}^{l \times 1}$, where r_i and s_i are the i th components of $\vec{r}$ and $\vec{s}$, with $i = 1, \dots, l$. The distance between these descriptors can be defined as follows.

(a) *Weighted Metric Distance.* Consider

$$\text{dist}_p(\vec{r}, \vec{s}) = \left(\sum_{i=1}^l \omega_i \cdot |r_i - s_i|^p \right)^{1/p}. \quad (7)$$

If we consider $\omega_i = 1$, $i = 1, \dots, l$, the Minkowski distance is obtained. Two particular cases will be considered: dist_1 (*cityblock* distance), which is defined from the Minkowski distance with $p = 1$, and dist_2 (Euclidean distance), doing $p = 2$.

(b) *Pearson Correlation Coefficient.* It is a similitude coefficient that can be obtained as follows:

$$\text{sim}_{\text{pea}}(\vec{r}, \vec{s}) = \frac{\vec{r}_d^T \cdot \vec{s}_d}{|\vec{r}_d| |\vec{s}_d|}, \quad (8)$$

where $\vec{r}_d = [r_1 - \bar{r}, \dots, r_l - \bar{r}]$ and $\vec{s}_d = [s_1 - \bar{s}, \dots, s_l - \bar{s}]$, $\bar{r} = (1/l) \sum_j r_j$, $\bar{s} = (1/l) \sum_j s_j$. It takes values in the range $[-1, +1]$. From this similitude coefficient, a distance measure can be defined as follows:

$$\text{dist}_{\text{pea}}(\vec{r}, \vec{s}) = 1 - \text{sim}_{\text{pea}}(\vec{r}, \vec{s}). \quad (9)$$

(c) *Inner Product.* It is also a similitude coefficient that can be calculated as the scalar product between the two vectors to compare

$$\text{sim}_{\text{cos}}(\vec{r}, \vec{s}) = \frac{\vec{r}^T \cdot \vec{s}}{|\vec{r}| |\vec{s}|}. \quad (10)$$

As shown in the equation, $\vec{r}$ and $\vec{s}$ are usually normalized. In this case, this measure is known as *cosine similitude* and takes values in the range $[-1, +1]$. The corresponding distance value is

$$\text{dist}_{\text{cos}}(\vec{r}, \vec{s}) = 1 - \text{sim}_{\text{in}}(\vec{r}, \vec{s}). \quad (11)$$

(d) *Other Distance Measures.* Other distance measures have been considered in the study, as they have provided good results when applied to very-high dimensional data in clustering tasks [33]. We name them log and root distances:

$$\begin{aligned} \text{dist}_{\log}(\vec{r}, \vec{s}) \\ = -\log_{10} \left(1 - \frac{1}{l} \sum_{i=1}^l \frac{|r_i - s_i|}{\max(r_i) - \min(r_i)} \right), \end{aligned} \quad (12)$$

where $\max(r_j)$ and $\min(r_j)$ are, respectively, the maximum and minimum value among the j components of the n vectors in $\mathbf{R} = \{\vec{r}_1, \dots, \vec{r}_n\}$. This way, the distance does not only depend on $\vec{r}$ and $\vec{s}$, but also on the set of vectors in $\mathbf{R}$:

$$\text{dist}_{\text{root}}(\vec{r}, \vec{s}) = \sqrt{\frac{1}{l} \sum_{i=1}^l \left(\frac{r_i - s_i}{r_i + s_i} \right)^2}. \quad (13)$$

To study the relation between the image distance (distance between the descriptors of two images) and the geometric distance (Euclidean distance between the points where these images were captured) the rooms *kitchen* and *hall* of data set 2 have been used, since these are the two rooms whose grid presents a higher resolution (10×10 cm). In both cases, from a reference point, some sets of scenes both horizontally and vertically have been taken and the distance between the reference image and all of them has been obtained. The next figures show the results obtained (average distance and variance) after this set of experiments.

First, Figure 7 shows the distance results when using FS. This figures show how, in this case, the different distance measures present quite similar results. In a close interval to the reference image, the image distance increases (quite linearly in the case of the correlation and cosine distances). However, they present a nondesirable behaviour since they reach a maximum and then they begin to decrease. The cosine distance is not shown as it provides a very similar result to the correlation.

Next, Figure 8 shows the results when using rotational PCA to describe scenes. In all cases, $k_3 = 200$ components have been used. The result obtained with the distance *cityblock* is remarkable because, despite being the simplest measure, it behaves quite linearly when the number of rotations is high enough (but it presents a local minimum in the middle). Logarithm and root distances present also relatively good results. The data in Figure 9 allow us to analyse the influence of the number of PCA components. In all cases, including a low number of components (very compact descriptors), the behaviour is quite linear and monotonous with some distance measures.

Thirdly, Figure 10 shows the results obtained when the images are described through HOG. In all cases, the results are quite similar to the FS. However, the local maximum is reached in a closer point to the reference image. This fact limits the validity range of the computed distance.

To finish the distance results, we show the results obtained with *gist* in Figure 11. In this case, thanks to the linearity and monotony, the results obtained with correlation (and cosine) must be highlighted.

As a final conclusion, the FS and HOG descriptors present a limited utility to estimate the topological distance between images, provided that the behaviour of the distances is not monotonous (FS presents a larger useful interval). Rotational PCA presents a relatively good behaviour when using *cityblock*, Euclidean, and Minkowski distance. At last, the excellent performance of *gist* with the *cityblock* and correlation distances must be highlighted, due to their monotony and linearity. The goodness of this configuration suggests

FIGURE 7: FS. Image distance versus geometric distance, depending on k_1 . Distance measure: (a) *cityblock*, (b) Euclidean, (c) weighted $p = 3$, $\omega_i = 0.95^i$, (d) correlation, (e) logarithm, and (f) root.

FIGURE 8: Rotational PCA. Image distance versus geometric distance, depending on N_R . Distance measure: (a) *cityblock*, (b) Euclidean, (c) weighted $p = 2$, $\omega_i = 0.95^i$, (d) correlation, (e) logarithm, and (f) root.

FIGURE 9: Rotational PCA. Image distance versus geometric distance, depending on k_3 . (a) cityblock distance and $N_R = 64$, (b) logarithm distance and $N_R = 64$, and (c) Euclidean distance and $N_R = 4$.

that it could be the first option to implement a topological mapping algorithm.

4.4. Topological Model. This section reflects the last experiment carried out. The algorithm presented in Section 3.2 has been used to build several topological maps using the data set 2. This data set presents different grid steps, depending on the room considered. This way, it allows us to study the influence of this important parameter.

As far as the configuration of the mass-spring-damper algorithm is concerned, the most critical parameter is the spring constant. If we consider that all the springs have the same elastic constant, the results are not consistent, because the presence of *visual aliasing* in the environment introduces undesired forces in the system. To avoid this effect, each elastic constant is calculated depending on the distances

between the descriptors of the two particles i and j linked by this spring, according to the following expression:

$$k_{ij} = \min \left(\frac{c}{\text{dist}(\vec{d}_i, \vec{d}_j)^2}, 100 \right), \quad (14)$$

$$i, j = 1, \dots, n, \quad i \neq j,$$

where c is the average slope measured on Figures 7–11, depending on the selected descriptor and parameters. The value of k_{ij} has been limited to 100 to avoid the presence of very high efforts.

At last, the natural length of each spring is equal to the distance between the descriptors of the particles linked by the spring:

$$l_{ij}^0 = \text{dist}(\vec{d}_i, \vec{d}_j). \quad (15)$$

FIGURE 10: HOG. Image distance versus geometric distance, depending on k_4 . Distance measure: (a) *cityblock*, (b) *Euclidean*, (c) *weighted* $p = 3$, $\omega_i = 1$, (d) *correlation*, (e) *logarithm*, and (f) *root*.

FIGURE 11: *Gist*. Image distance versus geometric distance, depending on k_6 . Distance measure: (a) *cityblock*, (b) Euclidean, (c) weighted $p = 3$, $\omega_i = 1$, (d) correlation, (e) logarithm, and (f) root.

FIGURE 12: Topological maps created in the *hall*, data set 2. The grid size is 10×10 cm.

To finish, all the particles are considered to have the same mass $m_i = 1$, $i = 1, \dots, n$, since our experiments have shown that it is not a relevant parameter. The damping constant of all the dampers is set to $\kappa_{ij} = 0.6$. Thanks to this dynamic friction, the behaviour of the system tends to be overdamped and more stable, permitting a gradual evolution from the initial position to the steady state, without large oscillations. To finish, we have defined the time step $\Delta t = 0.03$ s. It is an important parameter that influences both the settling time and the stability of the resulting system. A low value supposes a high settling time and a high value makes this time lower but the movement between two consecutive iterations may be so high that the system could destabilize.

After a complete bank of experiments, the best results have been obtained with the *gist* descriptor with $k_6 = 16$ blocks, $m = 16$ orientations, and correlation distance and with the FS descriptor with $k_1 = 32$ blocks and correlation distance. These results are in line with Section 4.3. HOG has not provided good results, as Figure 10 suggested.

Figures 12, 13, and 14 show some of the topological maps created in three different rooms of data set 2 (*hall*, *laboratory*, and *corridor*, resp.). We show the results of these rooms because they have different grid size. In each room, several sets of images, with different size and distribution along all the space of the room, have been chosen. Then, the mapping algorithm has been applied. The final distribution of each map is shown. In these maps, the lines are drawn just with representative purposes (when the algorithm starts, it has no information about the initial positions nor about the vicinity relations).

The figures show that, despite the different grid size, relatively good results are achieved in all cases. This way, global-appearance descriptors prove to be a good choice for the creation of topological maps where the concepts of closeness and farness are included.

Comparing to feature-based techniques, in a previous work [34], a new global-appearance description method was proposed and a preliminary comparison with a classical global-appearance method (the Fourier Signature) and a feature-based method (SIFT features) was carried out. The results showed that global-appearance descriptors are robust to solve the localization process and their computational cost is relatively low, improving the performance of local feature descriptors.

To finish, Table 4 shows a final comparison of the performance of the four methods in mapping tasks. First, to compare the computational cost, the table shows the minimum and the maximum necessary time ($t_{\min}$ and $t_{\max}$, resp.) to include each image in the model. Second, to study the relation between the image distance and the geometric distance, a least squares linear fit has been carried out with all the curves in Figures 7–11. In all cases, the origin has been weighted to ensure that the resulting line passes through it. The table shows the results of the best fit: the slope m , the coefficient r^2 , and the values of the parameters.

5. Conclusion and Future Works

This paper has focused on the study of the mapping problem. It has been addressed from a topological point of view, using the information provided by an omnidirectional vision sensor to build the model, and methods based on global appearance to extract relevant information from the scenes. The work has carried out a comparative evaluation between some renowned description methods in map building tasks.

The main contributions of the paper include an exhaustive study of visual appearance techniques (FS, PCA, HOG, and *gist*) and the adaptation of some of these algorithms to store position and orientation information from panoramic scenes. Also, the computational cost to build the nodes

FIGURE 13: Topological maps created in the *laboratory*, data set 2. The grid size is 30×30 cm.

TABLE 4: Performance of the description methods: computational cost per image to build the model and best linear fit of the image distance versus the geometric distance.

Descriptor	Time		m	r^2	Best linear fit	
	$t_{\min}$ (ms)	$t_{\max}$ (ms)			Parameters	Distance
FS	13	17	0.50	0.9734	$k_1 = 4$	Weighted
r-PCA	675	111740	$2.7 \cdot 10^5$	0.9856	$k_3 = 50, N_R = 64$	<i>Cityblock</i>
HOG	49	456	0.17	0.9576	$k_4 = 16$	Correlation
<i>Gist</i>	110	333	14	0.9902	$k_6 = 4$	Correlation

of the map has been studied, including the influence of the most relevant parameters. This study has revealed that FS, HOG, and *gist* present a reasonable computational cost and, from this point of view, their use could be feasible in real time applications. Besides this, the performance of the descriptors has been tested in mapping tasks. First, we have focused on the relation between the image distance and the geometric distance, which allows us to know the descriptors that best reflect an idea of closeness and farness, since they are two important concepts to reflect in the map. All the description methods have been tested along with several distance measures, and the results have shown that *gist* and FS descriptors with certain distance measures present positive results. Second, a mass-spring-damper method has been implemented to build topological maps, their parameters have been tuned and several experiments have been carried out. To finish, several topological maps have been built, including not only connectivity but also closeness and farness concepts. The results have shown the goodness of the mapping approach and of the parameters tuning.

These results have demonstrated that global-appearance methods are a feasible approach to solve the mapping task. Thanks to them, the robot can build a model of the environment that goes beyond the classical *topological maps* since the model is a version of the original grid except for a scale factor. This suggests that the model could be used to estimate with accuracy the position and orientation of the robot in the environment, with computational efficiency. This fact may have interesting implications in future developments in the field of mobile robotics. As an example, this concept can be used to build hybrid maps that arrange the information in several layers, with different accuracy: a high level layer that permits carrying out a rough and quick localization and a lower layer that contains information with geometric accuracy and allows the robot to refine the estimation of its position. Global-appearance methods can be used on their own or in conjunction with feature-based techniques to develop algorithms that face these problems efficiently.

All these facts encourage us to go into this framework in depth. To build a fully autonomous mapping and localization

FIGURE 14: Topological maps created in the *corridor*, data set 2. The grid size is 50×50 cm.

system several future works should be considered. First, the image collection process could be automated to obtain an optimal representation of the environment. Second, this model must be used to estimate the current position and orientation of the robot taking into account typical situations such as changes in lighting conditions or visual occlusions. At last, both processes could be integrated in a topological SLAM system that carries out both the model creation and the localization from the scratch. To optimize these algorithms we also consider carrying out a complete comparison between global-appearance and feature-based techniques as a future work.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

Acknowledgments

This work has been supported by the Spanish Government through the project DPI 2013-41557-P, *Navegación de Robots en Entornos Dinámicos Mediante Mapas Compactos con Información Visual de Apariencia Global*, and by the Generalitat Valenciana through the project GV/2015/031: *Creación de*

Mapas Topológicos a Partir de la Apariencia Global de un Conjunto de Escenas.

References

- [1] J. Gaspar, N. Winters, and J. Santos-Victor, "Vision-based navigation and environmental representations with an omnidirectional camera," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 890–898, 2000.
- [2] S. Thrun, "Robotic mapping: a survey, in exploring artificial intelligence," in *The New Milenium*, pp. 1–35, Morgan Kaufmann, San Francisco, Calif, USA, 2003.
- [3] A. Gil, Ó. Reinoso, M. Ballesta, M. Juliá, and L. Payá, "Estimation of visual maps with a robot network equipped with vision sensors," *Sensors*, vol. 10, no. 5, pp. 5209–5232, 2010.
- [4] E. Garcia-Fidalgo and A. Ortiz, "Vision-based topological mapping and localization methods: a survey," *Robotics and Autonomous Systems*, vol. 64, pp. 1–20, 2015.
- [5] K. Konolige, E. Marder-Eppstein, and B. Marthi, "Navigation in hybrid metric-topological maps," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '11)*, pp. 3041–3047, Shanghai, China, May 2011.
- [6] A. Angeli, S. Doncieux, J.-A. Meyer, and D. Filliat, "Visual topological slam and global localization," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '09)*, pp. 4300–4305, IEEE, Kobe, Japan, May 2009.
- [7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [8] C. Valgren and A. J. Lilienthal, "SIFT, SURF & seasons: appearance-based long-term localization in outdoor environments," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 149–156, 2010.
- [9] A. C. Murillo, J. J. Guerrero, and C. Sagüés, "SURF features for efficient robot localization with omnidirectional images," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 3901–3907, IEEE, Rome, Italy, April 2007.
- [10] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: speeded up robust features," in *Computer Vision—ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7–13, 2006. Proceedings, Part I*, A. Leonardis, H. Bischof, and A. Pinz, Eds., vol. 3951 of *Lecture Notes in Computer Science*, pp. 404–417, Springer, Berlin, Germany, 2006.
- [11] J. Kosecka, L. Zhou, P. Barber, and Z. Duric, "Qualitative image based localization in indoors environments," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. II-3–II-8, Madison, Wis, USA, June 2003.
- [12] E. Menegatti, T. Maeda, and H. Ishiguro, "Image-based memory for robot navigation using properties of omnidirectional images," *Robotics and Autonomous Systems*, vol. 47, no. 4, pp. 251–267, 2004.
- [13] F. Rossi, A. Ranganathan, F. Dellaert, and E. Menegatti, "Toward topological localization with spherical fourier transform and uncalibrated camera in," in *Proceedings of the International Conference on Simulation, Modeling and Programming for Autonomous Robots (SIMPAN '08)*, pp. 319–350, Springer, Venice, Italy, 2008.
- [14] A. Gil, O. M. Mozos, M. Ballesta, and O. Reinoso, "A comparative evaluation of interest point detectors and local descriptors for visual SLAM," *Machine Vision and Applications*, vol. 21, no. 6, pp. 905–920, 2010.
- [15] A. Guérin-Dugué and A. Oliva, "Classification of scene photographs from local orientations features," *Pattern Recognition Letters*, vol. 21, no. 13-14, pp. 1135–1140, 2000.
- [16] M. Kirby, *Geometric Data Analysis: An Empirical Approach to Dimensionality Reduction and the Study of Patterns*, Wiley-Interscience, 2001, <http://books.google.es/books?id=nRmFQgAACAAJ>.
- [17] A. Leonardis and H. Bischof, "Robust recognition using eigen-images," *Computer Vision and Image Understanding*, vol. 78, no. 1, pp. 99–118, 2000.
- [18] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.
- [19] M. Jogan and A. Leonardis, "Robust localization using eigenspace of spinning-images," in *Proceedings of the IEEE Workshop on Omnidirectional Vision*, pp. 37–44, IEEE, Hilton Head Island, SC, USA, June 2000.
- [20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*, vol. 1, pp. 886–893, San Diego, Calif, USA, June 2005.
- [21] Q. Zhu, S. Avidan, M.-C. Yeh, and K.-T. Cheng, "Fast human detection using a cascade of histograms of oriented gradients," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 2, pp. 1491–1498, IEEE, June 2006.
- [22] M. Hofmeister, M. Liebsch, and A. Zell, "Visual self-localization for small mobile robots with weighted gradient orientation histograms," in *Proceedings of the 40th International Symposium on Robotics (ISR '09)*, pp. 87–91, IFR, Barcelona, Spain, March 2009.
- [23] L. Payá, F. Amorós, L. Fernández, and O. Reinoso, "Performance of global-appearance descriptors in map building and localization using omnidirectional vision," *Sensors*, vol. 14, no. 2, pp. 3033–3064, 2014.
- [24] A. Oliva and A. Torralba, "Building the gist of ascene: the role of global image features in recognition," *Progress in Brain Reasearch*, vol. 155, pp. 23–36, 2006.
- [25] I. Biederman, "Aspects and extension of a theory of human image understanding," in *Computational Processes in Human Vision: An Interdisciplinay Perspective*, Ablex, Norwood, NJ, USA, 1988.
- [26] C. Siagian and L. Itti, "Biologically inspired mobile robot vision localization," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 861–873, 2009.
- [27] C.-K. Chang, C. Siagian, and L. Itti, "Mobile robot vision navigation & localization using gist and saliency," in *Proceedings of the 23rd IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems (IROS '10)*, pp. 4147–4154, IEEE, Taipei, Taiwan, October 2010.
- [28] A. C. Murillo, G. Singh, J. Kosecká, and J. J. Guerrero, "Localization in urban environments using a panoramic gist descriptor," *IEEE Transactions on Robotics*, vol. 29, no. 1, pp. 146–160, 2013.
- [29] A. Selle, M. Lentine, and R. Fedkiw, "A mass spring model for hair simulation," *ACM Transactions on Graphics*, vol. 27, no. 3, pp. 64:1–64:11, 2008.
- [30] Automation-Robotics and Computer Vision Research Group (ARVC), "Quorum 5 set of images," Miguel Hernández University, Elche, Spain, <http://arvc.umh.es/db/images/quorumv/>.

- [31] R. Möller, A. Vardy, S. Kreft, and S. Ruwisch, "Visual homing in environments with anisotropic landmark distribution," *Autonomous Robots*, vol. 23, no. 3, pp. 231–245, 2007.
- [32] M. Artač, M. Jogan, and A. Leonardis, "Mobile robot localization using an incremental eigenspace model," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1025–1030, IEEE, May 2002.
- [33] H. Spat, *Clustering Analysis Algorithms for Data Reduction and Classification of Objects*, Ellis Horwood, New York, NY, USA, 1982.
- [34] Y. Berenguer, L. Payá, M. Ballesta, and O. Reinoso, "Position estimation and local mapping using omnidirectional images and global appearance descriptors," *Sensors*, vol. 15, no. 10, pp. 26368–26395, 2015.