Hindawi Publishing Corporation
Journal of Sensors
Volume 2016, Article ID 1537891, 12 pages
http://dx.doi.org/10.1155/2016/1537891

Research Article

A Study of Visual Descriptors for Outdoor Navigation Using Google Street View Images

L. Fernández, L. Payá, O. Reinoso, L. M. Jiménez, and M. Ballesta

Department of Systems Engineering and Automation, Miguel Hernandez University, Avda. de la Universidad s/n, Elche, 03202 Alicante, Spain

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

Received 22 March 2016; Revised 24 August 2016; Accepted 2 November 2016

Academic Editor: Jose R. Martinez-de Dios

A comparative analysis between several methods to describe outdoor panoramic images is presented. The main objective consists in studying the performance of these methods in the localization process of a mobile robot (vehicle) in an outdoor environment, when a visual map that contains images acquired from different positions of the environment is available. With this aim, we make use of the database provided by Google Street View, which contains spherical panoramic images captured in urban environments and their GPS position. The main benefit of using these images resides in the fact that it permits testing any novel localization algorithm in countless outdoor environments anywhere in the world and under realistic capture conditions. The main contribution of this work consists in performing a comparative evaluation of different methods to describe images to solve the localization problem in an outdoor dense map using only visual information. We have tested our algorithms using several sets of panoramic images captured in different outdoor environments. The results obtained in the work can be useful to select an appropriate description method for visual navigation tasks in outdoor environments using the Google Street View database and taking into consideration both the accuracy in localization and the computational efficiency of the algorithm.

1. Introduction

Designing vehicles capable of navigating autonomously, in a previously unknown environment and with no human intervention, is a fundamental objective in mobile robotics. To achieve this objective, the vehicle must be able to build a model (or map) of the environment and to estimate its position within this model. A great variety of localization approaches can be found in the literature. In general, the position and orientation of the robot can be obtained from proprioceptive (odometer) or exteroceptive (laser, camera, or sonar) sensors, as presented in the works of Thrun et al. [1] and Gil et al. [2].

With the exteroceptive approach, the use of computer vision to create a representation of the environment is very extended due to the good relationship *quantity of information/cost* that the cameras offer. The research developed during the last years in the topic of map creation using visual information is enormous, and new algorithms are presented continuously. Usually, one of the key points of these algorithms is the description of the visual information to extract relevant information which is useful for the robot to estimate its position and orientation. In general, the problem can be approached from two points of view: local features extraction and global-appearance approaches. In the first one, a number of landmarks (distinctive points or regions) are extracted from each scene and each landmark is described to obtain a descriptor which is invariant against changes in the robot position and orientation. Murillo et al. [3] presented an algorithm that made use of the SURF (Speeded Up Robust Features) description method [4] to improve the performance of appearance-based localization methods using omnidirectional images in large data sets. On the other hand, global-appearance approaches consist in representing each scene by a single descriptor which is computed working with the scene as a whole, with no local feature extraction. This approach has recently become popular and some examples can be found. Rossi et al. [5] present a metric to compute the image similarity using the Fourier Transform of spherical omnidirectional images in order to carry out the localization of a mobile robot. Payá et al. [6] present a framework to carry out multirobot route following using an appearance-based approach with omnidirectional images to represent the environment and a probabilistic method to estimate the localization of the robot. Finally, Fernández et al. [7] deal with the problem of robot localization using the visual information provided by a single omnidirectional camera mounted on the robot, using techniques based on the global appearance of panoramic images and a Monte Carlo Localization (MCL) algorithm [8].

The availability of spherical images that represent outdoor environments is nowadays almost unlimited, thanks to the services of Google Street View. Furthermore, these images provide a complete 360-degree view of the scenery in the ground plane and 180-degree view vertically. Thanks to this great amount of information, these images can be used to carry out autonomous navigation tasks robustly. Using a set of these previously available spherical images as a dense visual map of an environment, it is possible to develop an autonomous localization and navigation system employing the images captured by a mobile robot or vehicle and comparing them with the map information in order to resolve the localization problem. This way, in this paper, we consider the use of the images provided by Google Street View as a visual map of the environment in which a mobile robot must be localized using the image acquired from an unknown position.

The literature regarding the navigation problem using Google Street View information is somewhat sparse but growing in recent years. For example, Gamallo et al. [9] proposed the combination of a low cost GPS with a particle filter to implement a vision based localization system that compares traversable regions detected by a camera with regions previously labeled in a map (composed of Google Maps images). The main contribution of this work is that a synthetic image of what the robot should see from the predicted position is generated and compared with the real observation to calculate the weight of each particle. Torii et al. [10] tried to predict the GPS location of a query image given the Google Street View database. This work presents a design of a matching procedure that considers linear combinations of bag-of-feature vectors of database images. With respect to indoor pose estimation, Aly and Bouguet [11] present an algorithm that takes as input spherical Google Street View images and as output their relative pose up to a global scale. Finally, Taneja et al. [12] proposed a method to refine the calibration of the Google Street View images leveraging cadastral 3D information.

The localization of the vehicle/robot can be formulated as the problem of matching the currently captured image with the images previously stored in the dense map (images in the database). Nowadays, a great variety of detection and description methods have been proposed in the context of visual navigation but, in our opinion, there exists no consensus on this matter when we use outdoor images.

Amorós et al. [13] carried out a review and comparison of different global-appearance methods to create descriptors of panoramic scenes in order to extract the most relevant information. The authors of this work developed a set of experiments with panoramic images captured in indoor environments to demonstrate the applicability of some appearance descriptors to robotic navigation tasks and to measure their quality in position and orientation estimation. However, as far as outdoor scenarios are concerned, there is no revision of methods that offer good results. This situation, combined with the fact that using Google Street View images has barely been tested in autonomous navigation systems, has motivated the work presented here. Following this philosophy, we made a comparison between different descriptors of panoramic images but, in this case, we used Google Street View images captured in outdoor environments. This is a more challenging problem due to several features: the openness of the images (i.e., the degree of dominance of some structures such as the sky and the road which do not add distinctiveness to the image), their changing lighting conditions, and the large geometrical distance between the points where the images were captured.

Taking these features into account, we consider that it is worth carrying out a comparative evaluation of the performance of different image descriptors under real conditions of autonomous outdoor localization, since it would be a necessary step prior to the implementation of a visual navigation framework. In this paper, we evaluate two different approaches: approaches based on local features and approaches based on global appearance. In both cases we test the performance of the descriptor depending on the main parameters that configure it and we make a graphical representation of the *precision* of each method versus the *recall* [14].

When a robot has to navigate autonomously outdoors, very often a rough estimation of the area where the robot moves is available, and the robot must be able to estimate its position in this wide zone. This work focus on this task; we assume the zone where the robot navigates is approximately known and it must estimate its position more accurately in this area. With this aim, two different wide areas have been chosen to evaluate the performance of the localization algorithms, and a set of images per area has been obtained from the Google Street View database.

The remainder of the paper is organized as follows. In Section 2, we present the description methods evaluated in this work. In Section 3, the experimental setup and the databases we have used are described. Section 4 describes the method we have followed to evaluate the descriptors in a localization process. Section 5 presents the experimental results. Finally, in Section 6, we outline the conclusions and the future works.

2. Image Descriptors

In this section, we present five different image descriptors that are suitable to build a compact description of the appearance of each scene [13–15]. One of the methods, previously denoted as a feature-based approach, consists in representing the image as a set of landmarks extracted from the scene along with the description of such landmarks. The method selected for this landmarks description is SURF (Speeded Up Robust Features). The other methods chosen to carry out the comparative analysis are the following appearance-based methods: the two-dimensional Discrete Fourier Transform (DFT), the

Fourier Signature (FS), *gist,* and the Histogram of Oriented Gradients (HOG). Each method uses a different mechanism to express the global information of the scene. First, DFT and FS are based on the analysis in the frequency domain in two dimensions and one dimension, respectively. Second, the approach of *gist* we use is built from edges information, obtained through Gabor filtering and analyzed in several scales. Finally, HOG gathers systematic information from the orientation of the gradient in local areas of the image. The choice of these description methods will permit analyzing the influence of each kind of information in the localization process.

The initial objective of this study was to compare some global-appearance methods. However, we have decided to include in this comparative evaluation a local features description method to make a more complete study. With this aim, we have chosen SURF due to its relatively low computational cost comparing with other classical feature-based approaches.

The next subsections present briefly the description methods included in the comparative evaluation.

2.1. SURF and Harris Corner Detector. The Speeded Up Robust Features (SURF) were introduced by Bay et al. [4]. This study showed that SURF outperform existing methods with respect to repeatability, robustness, and distinctiveness of the descriptors. The detection method uses integral images to reduce the computational time and is based on the Hessian matrix. On the other hand, the descriptor represents a distribution of Haar-wavelet responses within the interest point neighborhood and makes an efficient use of integral images. In this work we only include the standard SURF descriptor, which has a dimension of 64 components per landmark, but there are two more versions: the extended version (E-SURF) with 128 elements and the upright version (U-SURF), that is not invariant to rotation and has a length of 64 elements. On the other hand, we perform the detection of the features using the Harris corner detector (based on the eigenvalues of the second moment matrix [16]) because our experiments showed that this method extracted most robust points in outdoor images comparing to the SURF extraction method.

This way the method we use in this work is a combination of these two algorithms. More specifically, the Harris corner detector is used to extract the features from the image, and the standard SURF descriptor is used to characterize and describe each one of the landmarks previously detected.

2.2. Two-Dimensional Discrete Fourier Transform. From an image $f(x, y)$ with N_x rows and N_y columns, the 2D Discrete Fourier Transform (DFT) can be defined as follows:

$$\begin{aligned} F[f(x,y)] &= F(u,v) \\ &= \frac{1}{N_y N_x} \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} f(x,y) \cdot e^{-2\pi j(ux/N_x + vy/N_y)}, \\ & u = 0,\ldots,N_x-1,\ v = 0,\ldots,N_y-1, \end{aligned} \tag{1}$$

where (u, v) are the frequency variables and the transformed function $F(u, v)$ is a complex function which can be decomposed into a magnitudes matrix and an arguments matrix. This transformation presents some interesting properties which are helpful in robot localization tasks. First, the most relevant information in the Fourier domain concentrates in the low frequency components, so it is possible to reduce the amount of memory and to optimize the computational cost by retaining only the first k_x rows and k_y columns in the transform. Second, when $f(x, y)$ is a panoramic scene, a translation in the rows and/or columns of the original image produces a change only in the arguments matrix [15]. This way, the magnitudes matrix contains information which is invariant to rotations of the robot in the ground plane, and the arguments matrix contains information that can be useful to estimate the orientation of the robot in this plane with respect to a reference image (using the DFT shift theorem).

Taking these facts into account, the global description of the image $f(x, y)$ consists of the magnitudes matrix $A(u, v)$ and the arguments matrix $\Phi(u, v)$ of its two-dimensional DFT. The dimensions of both matrices are $k_x < N_x$ rows and $k_y < N_y$ columns. On the one hand, $A(u, v)$ is useful to estimate the robot position and, on the other hand, the information in $\Phi(u, v)$ can be used to estimate the robot orientation.

2.3. Fourier Signature. The third image description method used in this comparative analysis is the Fourier Signature (FS), described initially by Menegatti et al. [17]. From an image $f(x, y)$ with N_x rows and N_y columns, the FS consists in obtaining the one-dimensional DFT of each row. This method presents some advantages, such as its simplicity, its low computational cost, and the fact that it exploits better the invariance against rotations of the robot in the ground plane when we work with panoramic views.

More specifically, the process to compute the FS consists in transforming each row x of the original panoramic image $\{f_x\} = \{f_{x,0}, f_{x,1}, \ldots, f_{x,N_y-1}\}$, $x = 0, \ldots, N_x - 1$, into the sequence of complex numbers $\{F_x\} = \{F_{x,0}, F_{x,1}, \ldots, F_{x,N_y-1}\}$, $x = 0, \ldots, N_x - 1$, according to the 1D-DFT expression:

$$\begin{aligned} F_{x,k} &= \sum_{n=0}^{N_y-1} f_{x,n} \cdot e^{-j(2\pi/N_y)kn}, \\ & k = 0,\ldots,N_y-1,\ x = 0,\ldots,N_x-1. \end{aligned} \tag{2}$$

The result is a complex matrix $F(x, v)$, where v is a frequency variable, which can be decomposed into a magnitudes matrix and an arguments matrix.

Thanks to the 1D-DFT properties it is possible to represent each row of $F(x, v)$ with the first coefficients since the most relevant information is concentrated in the low frequency components of each row in the descriptor, so it is possible to reduce the amount of memory by retaining only k_y first columns in signature $F(x, v)$. Also, when $f(x, y)$ is a panoramic scene, the modules matrix is invariant against robot rotations in the ground plane and the magnitudes matrix permits estimating the change in the robot orientation using the DFT shift theorem [15, 17, 18].

Taking these facts into account, the global description of the image $f(x, y)$ consists of the magnitudes matrix $A(x, v)$ and the arguments matrix $\Phi(x, v)$ of the Fourier Signature. The dimensions of both matrices are N_x rows and $k_y < N_y$ columns. First, the position of the robot can be estimated using the information in $A(x, v)$, since it is invariant to changes in robot orientation and second $\Phi(x, v)$ can be used to estimate the robot orientation.

2.4. Gist. The concept of the *gist* of an image can be defined as an abstract representation that activates the memory of scene categories [19]. The *gist*-based descriptors try to represent the image by obtaining its essential information simulating the human perception system and its ability to recognize a scene through the identification of color saliency or remarkable structures. Torralba [20] presents a model to obtain global scene features, working in several spatial frequencies and using different scales based on Gabor filtering. They use these features in a scene recognition and classification task. In previous works [13] we employed a *gist*-Gabor descriptor in order to obtain frequency and orientation information. Due to the good results obtained in indoor environments when the mobile robot presents 3 DOF (degrees of freedom) movements on the ground plane, the fourth method employed in the comparative analysis presented in this paper is the *gist* descriptor of panoramic images.

The method starts with two versions of the initial panoramic image $f(x, y)$: the original one, with N_x rows and N_y columns, and a new version after applying a Gaussian low-pass filter and subsampling to a new size equal to $0.5 \cdot N_x \times 0.5 \cdot N_y$. After that, both images are filtered with a bank of n_f Gabor filters whose orientations are evenly distributed to cover the whole circle. Then, to reduce the amount of information, the pixels into both images are grouped into k_1 horizontal blocks per image, whose width is equal to N_y in the first image and $0.5 \cdot N_y$ in the second one. The average value of the pixels in each group is calculated and all this information is arranged into a final descriptor, which is a column vector $\vec{g}$ with $2 \cdot k_1 \cdot n_f$ components. This descriptor is invariant against rotations of the vehicle on the ground plane. More information about the method can be found in [13].

2.5. Histogram of Oriented Gradients. The Histogram of Oriented Gradients (HOG) descriptors are based on the orientation of the gradient in local areas of an image. It was described initially by Dalal and Triggs [21]. More concisely, it consists first in obtaining the magnitude and orientation of the gradient of each pixel of the original image. This image is divided then into a set of cells and a histogram of gradient orientation is compiled for each cell, aggregating the information of the gradient orientation of each pixel within the cell, weighting with the magnitude of the pixel.

The omnidirectional images captured from a specific position of the ground plane contain the same pixels in a row, independently on the orientation of the robot in this plane, but in a different order. Taking this fact into account, if we calculate the histogram of cells that have the same width of the original image, we obtain a descriptor which is invariant against rotations of the robot.

The method we use is described in depth in [22] and can be summarized as follows. The initial panoramic image $f(x, y)$ with N_x rows and N_y columns is first filtered to obtain two images with the information of the horizontal and vertical edges, $f_x(x, y)$ and $f_y(x, y)$. From these two images, the magnitude of the gradient and its orientation is obtained, pixel by pixel, and the results are stored in matrices $M(x, y)$ and $\Theta(x, y)$, respectively. Matrix $\Theta(x, y)$ is then divided into k_2 horizontal cells, whose width is equal to N_y. For each cell, an orientation histogram with b bins is compiled. During this process, each pixel in $\Theta(x, y)$ is weighted with the magnitude of the corresponding pixel in $M(x, y)$. At the end of the process, the set of histograms constitutes the final descriptor $\vec{h}$ which is a column vector with $k_2 \cdot b$ components.

3. Experiments Setup

The main objective of this work consists in carrying out an exhaustive evaluation of the performance of the description methods presented in the previous section. All these methods will be included in a localization algorithm and their performance will be evaluated and compared both in terms of computational cost and localization accuracy. The results of this comparative evaluation will give us an idea of which is the description method that offers the best results in outdoor environments when using Google Street View images.

With this aim, two different regions in the city of Elche (Spain) have been selected and the Google Maps images of these two areas have been obtained and stored in two data sets. Each one of these data sets will constitute a map and will be used subsequently to estimate the position of the vehicle within the map by comparing the image captured by the vehicle from the unknown position with the images previously stored in each map.

The main features of the two sets of images are as follows.

Set 1. Set 1 consists of 177 full spherical panorama images with resolution generally up to 3328 × 1664 pixels. Each image covers a field of view of 360 degrees in the ground plane and 180 degrees vertically. Figure 1 shows the GPS position where each image was captured (blue dots) and two examples of the panoramic images after a preprocessing process. This set corresponds with a mesh topography database that contains images of various streets and open areas. The images cover an area of approximately 700 m × 300 m.

Set 2. Set 2 consists of 144 full spherical panorama images. The images have been captured along the same street with a linear topology covering approximately 1700 m. The appearance of these images is more urban. Figure 2 shows the GPS position where each image was captured (blue dots) and three examples of the panoramic images after a preprocessing process.

3.1. Image Preprocessing and Map Creation. Due to the wide vertical field of view of the acquisition system, the sky is

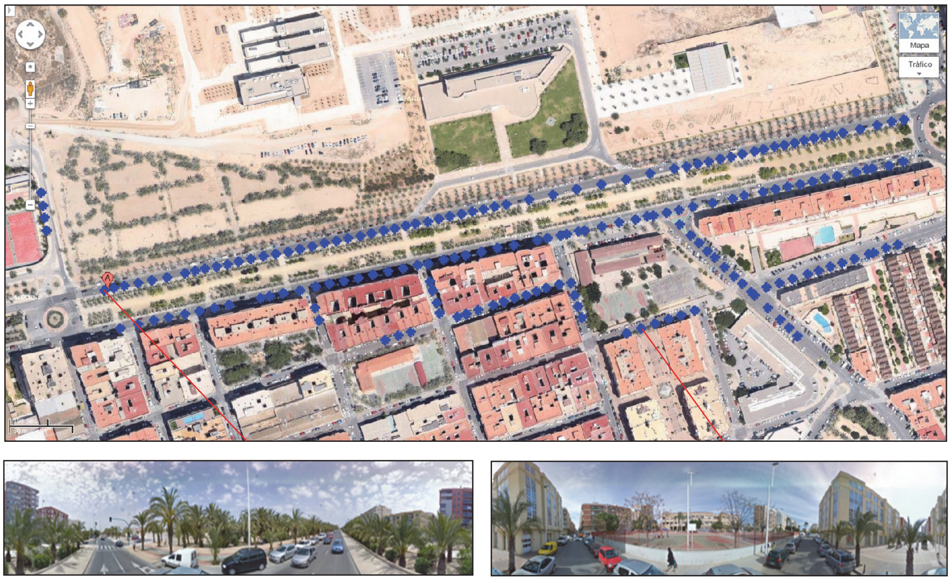


FIGURE 1: Bird eye's view of the region chosen as map 1 prior to the localization experiment. The blue dots represent the coordinates where the images of the Set 1 were captured. Two examples of Google Street View images after a preprocessing step are also shown.

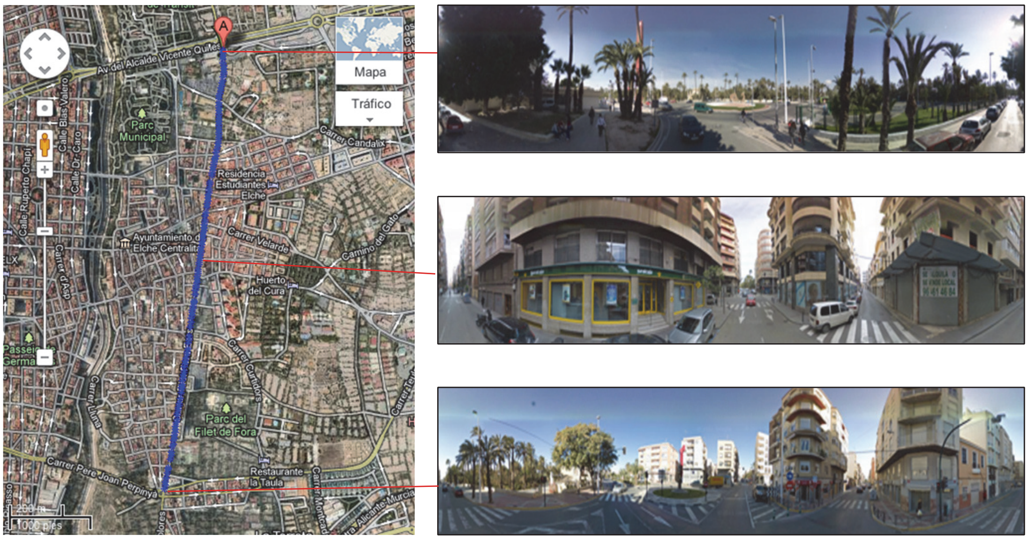


FIGURE 2: Bird eye's view of the region chosen as map 2 prior to the localization experiment. The blue dots represent the coordinates where the images of Set 2 were captured. Three examples of Google Street View images after a preprocessing step are also shown.

often a big portion of the Google Street View images. The appearance of this area will be very prone to changes when the localization process is carried out in a different time of day with respect to the time of day when the map was captured. Taking this fact into account, a preprocessing step has been carried out to remove part of the sky in the scenes.

Once part of the sky has been removed from all the scenes, the images are converted into grayscale and their resolution is reduced to 512 × 128 pixels, to ensure the computational viability of the algorithms.

After that, each image will be described using the five description methods presented in Section 2. At the end, one map will be available per image set and per description method. Each map will be composed of the set of descriptors of each panoramic scene.

3.2. Localization Process. Once the maps are available, in order to evaluate the different visual descriptors introduced in Section 2 to solve the localization problem, we also make use of Google Street View images.

To carry out the localization process, first we choose one of the images of the database (named as *test image*). In this moment, this image is removed from the map. Second, we compute the descriptor of the test image (using one of the methods presented in Section 2) and obtain the distance between this descriptor and the descriptors of the rest of the images stored in the corresponding map. As a result, the images of the map are arranged from the nearest to the farthest using the image distance as arranging criterion.

The result of the localization algorithm is considered correct if the first image it returns was captured on the map point which is geometrically the closest to the test image capture point (the GPS coordinates are used with this aim). We will refer to this case as a correct localization in *zone 1*. However, since this is a quite challenging and restrictive problem, it is also interesting to know if the first image that the algorithm returns was captured on one of the two geometrically closest points to the test image capture point (*zone 2*) or even on one of the three geometrically closest points (*zone 3*). The first case is the ideal one, but we are also interested in the other cases as they will indicate if the algorithm is returning an image in the surroundings of the actual position of the test image (i.e., the localization algorithm detects that the robot is in a zone close around its actual position).

This process is repeated for each description method, using all the images of Sets 1 and 2 as test images. In brief, the procedure to test the localization methods previously explained consists in the following steps, for each image and description method:

(1) Extracting one image of the set (denoted as test image); then, this test image is eliminated from the map

(2) Calculating the descriptor of the test image

(3) Calculating the distance between this descriptor and all the map descriptors, which we named *image distance*

(4) Retaining the most similar descriptor and studying if it corresponds to one image that has been captured in the surroundings of the test image capture point (*zone 1, 2,* or *3*)

As a result, the next data are retained for an individual test image: the *image distance* between the test image descriptor and the most similar map descriptor, D^t, and the localization results in *zone 1* (correct or wrong match), m_1^t, in *zone 2*, m_2^t, and in *zone 3*, m_3^t. After repeating this process with all the test images, the results will consist of four vectors, whose dimension is equal to the number of test images. The first vector, $\vec{D}$, contains the distances, D^t, and the other three, $\vec{m}_1$, $\vec{m}_2$, and $\vec{m}_3$, contain, respectively, the information of correct or incorrect matches in *zones 1, 2,* and *3*.

4. Evaluation Methods

In this work, the localization results are expressed by means of *recall* and *precision* curves [14]. To build them, the components of the vectors $\vec{D}$, $\vec{m}_1$, $\vec{m}_2$, and $\vec{m}_3$ are equally sorted in ascending order of the distances that appear in the first vector. The resulting sorted vectors of correct and wrong matches are then used to calculate the values of *recall* and *precision*. Let us focus on the sorted vector of matches in *zone 1*, $\vec{m}_1^s$. First, for each component in this vector, the *recall* is calculated as the number of correct matches obtained so far with respect to the total number of test images. Second, for each component in the same vector, the *precision* is obtained as the number of correct matches obtained so far with respect to the number of test images considered so far. Then, with the information contained in these vectors, a *precision versus recall* curve is built, corresponding to the localization in *zone 1*. This is repeated with the sorted vectors $\vec{m}_2^s$ and $\vec{m}_3^s$ to obtain the localization results in *zone 2* and *zone 3*.

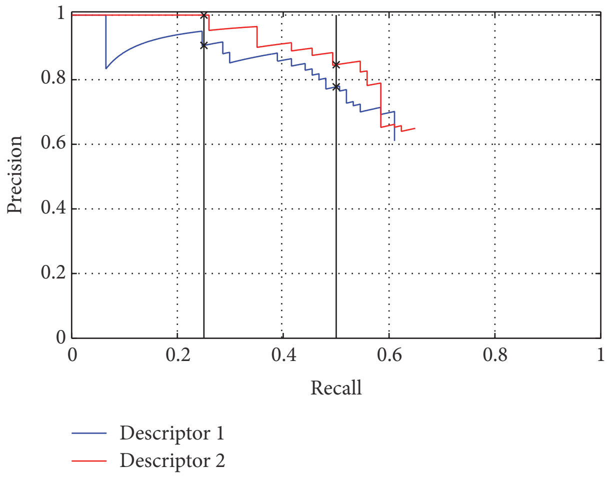


Figure 3: Two sample *recall-precision* graphs obtained after carrying out the localization experiments with two different description methods.

In our experiments, the most important piece of information of this type of graphs is the final point because it shows the global result of the experiment (final precision after considering all the test images). However, additional relevant information can be extracted from them, because the graph also shows the ability of the localization algorithm to find the correct match while considering a specific *image distance* threshold. As explained in the previous paragraph, the results have been arranged considering the ascending value of distances. Taking it into account, as the recall increases, the threshold also does. For this reason, the evolution of the *recall-precision* curves contains information about the robustness of the algorithm with respect to a specific *image distance* threshold. If the *precision* values stay high, independently of the *recall*, this shows the presence of a lower number of wrong results under this distance threshold. Figure 3 shows two sample *recall-precision* curves obtained after running the localization algorithm with all the test images and two different description methods, considering *zone 1*. Both curves show a similar final *precision* value, between 0.6 and 0.65. However, the evolutions present a different behavior. As an example, if we consider as threshold the distance associated to *recall* = 0.25, according to the graph, the precision of descriptor 1 is 100%, but the precision of descriptor 2 is 90%. This means that, considering the selected *image distance* threshold, 25% of correct localizations are achieved with 100% of probability using descriptor 1 and with a 90% of probability using descriptor 2. This study can be carried out considering any value for the *image distance* threshold.

Before running the algorithm, it is necessary to define the *image distance*. We use two different distance measures depending on the kind of descriptor used.

First, in the case of the feature-based method (SURF-Harris), it is necessary to extract the interest points prior to describing the appearance of the image. We propose using the Harris corner detector [16] to extract visual landmarks from the panoramic images. After that, each interest point is described using standard SURF. To compare the test image with the map images, first we extract and describe the interest points from all the images. After that, a matching process is carried out with these points. The points detected in the test image, captured with a particular position and orientation of the vehicle, are searched in the map images. The performance of the matching method is not the scope of this paper; we only employ it as a tool. Once the matching process has been carried out, we evaluate the performance of the descriptor taking into account the number of matched points, so that we will consider as closest image the one that presents more matching points with the test image. More concisely, we compute the distance between the test image t and any other image of the map j as

$$D_{\text{fea}}^{tj} = 1 - \left(\frac{NM^{tj}}{\max_j \left(\overrightarrow{NM^{t}} \right)} \right), \tag{3}$$

where NM^{tj} is the number of matches between the images t and j, $\overrightarrow{NM^{t}} = [NM^{t1}, \ldots, NM^{tn_{\text{map}}}]$ is a vector that contains the number of matches between the image t and every image of the map, and n_{map} is the number of images in the map.

Second, in the case of appearance-based methods (2D DFT, FS, *gist,* and HOG), no local information needs to be extracted from the images. Instead, the appearance of the whole images is compared. This way, the global descriptor of the test image is calculated and the distances between it and the descriptors of the map images are obtained. The Euclidean distance is used in this case, defined as

$$D_{E}^{tj} = \sqrt{\sum_{m=1}^{M} \left(\vec{d}_{t}^{\,m} - \vec{d}_{j}^{\,m} \right)^{2}}, \tag{4}$$

where $\vec{d}_t$ is the descriptor of the test image t, $\vec{d}_j$ is the descriptor of the map image j, and M is the size of the descriptors. This distance is normalized to obtain the final distance between the images t and j, according to the next expression:

$$D_{\text{app}}^{tj} = \frac{D_{E}^{tj}}{\max_j \left(\vec{D}_{E}^{t} \right)}, \tag{5}$$

where D_E^{tj} is the Euclidean distance between the descriptor of the test image t and the map image j, $\vec{D}_E^{t} = [D_E^{t1}, \ldots, D_E^{tn_{\text{map}}}]$ is a vector that contains the Euclidean distance between the descriptor of the image t and all the images in the map, and n_{map} is the number of images in the map.

It is important to note that the algorithm must be able to estimate the position of the robot with accuracy, but it is also important that the computational cost is adequate, to know whether it would be feasible to solve the problem in real time. To estimate the computational cost, we have computed, considering both maps in the experiments, the necessary time to calculate the descriptor of each test image, to compute the distance to the map descriptors and to detect the most similar descriptor. We must take into account that the descriptors of all the map images can be computed prior to the localization, in an off-line process. Apart from the time, we have also estimated the amount of memory needed to store each image descriptor.

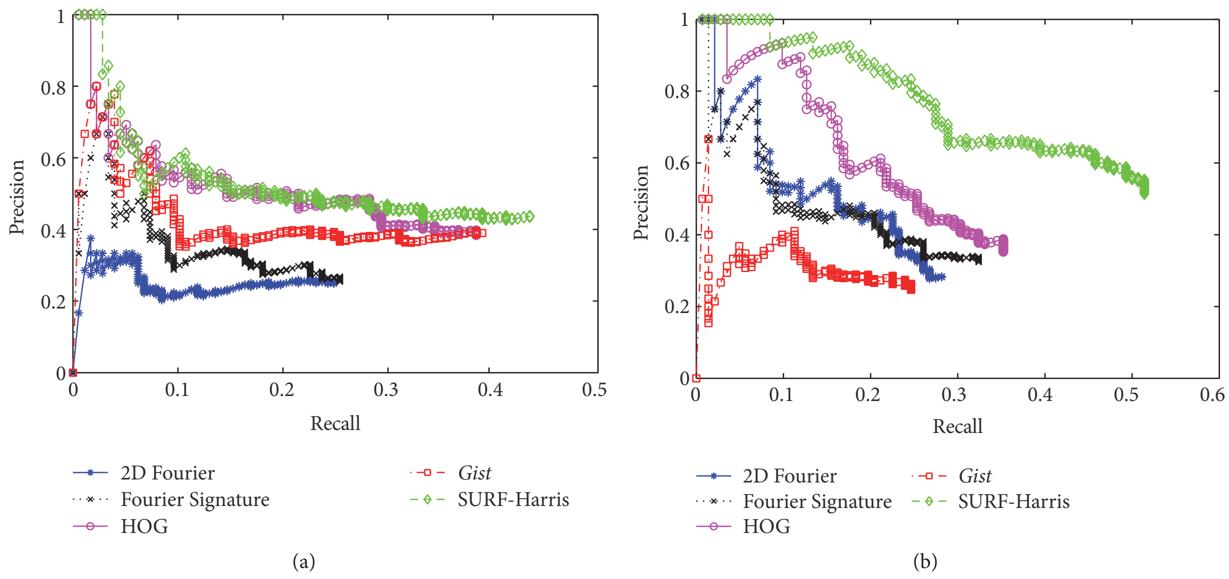
To finish, we also propose to study the relationship *distance between two image descriptors* versus *geometric distance between the capture points of these two images*. Ideally, the distance between the descriptors must increase as the geometric distance between capture points does (i.e., it must not present any local minima). This information is very interesting in applications such as map building, where the robot must be able to build a map using as input information only the distance between image descriptors. It is also important when it is necessary to estimate the position of the vehicle at halfway points within the grid map. Additionally, it may help to detect if the problem of *visual aliasing* is present in the environment (i.e., two zones which are geometrically far may present a similar visual appearance, which might lead to errors in the mapping and localization process).

5. Experimental Results

As stated in the previous section, with the purpose of establishing the capacity of each descriptor to correctly localize the robot (or vehicle), we have built *recall-precision* curves to reflect the results of each experiment. Figure 4 shows this graphical representation using (a) the first and (b) the second set of images (denoted as Sets 1 and 2 in previous sections). To build this figure, we consider the localization results in *zone 1*. This way, the figure shows the ability of the localization algorithm to correctly detect which image of the map was captured closer to the test image. This is the most restrictive case.

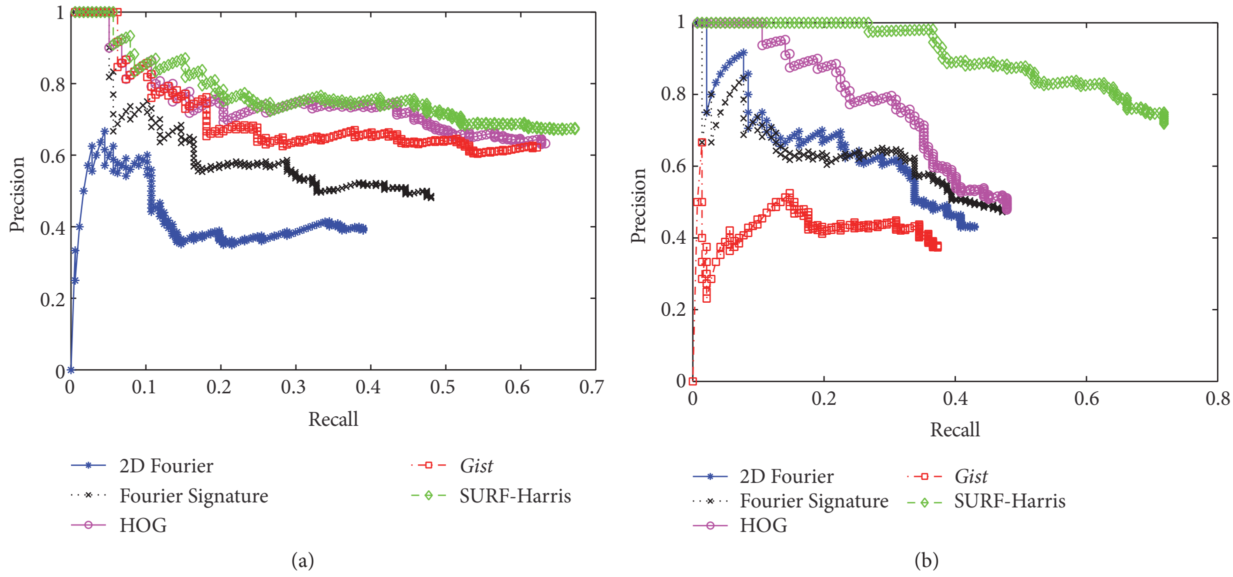
Apart from it, the performance of the localization algorithm in *zones 2* and *3* has also been studied. This way, Figure 5 shows the results of the localization process in *zone 2* using (a) Set 1 and (b) Set 2. Finally, Figure 6 shows the localization results in *zone 3* using (a) Set 1 and (b) Set 2. This is the least restrictive case among the three studied.

In all cases, the results show that the SURF-Harris descriptor presents a relatively better performance comparing to the other descriptors, in terms of accuracy and using both image sets. As far as the methods based on global appearance are concerned, the good behavior of HOG can be highlighted. In the case of the localization in *zone 2* it reaches 60% and 50% of precision in Sets 1 and 2, respectively. These results can be considered relatively good, taking into account the fact that the localization process is solved in an absolute way (i.e., we consider that no information about the previous position of the robot is available and the test image is compared with all the images stored in the data sets). In a real application, it



(a) (b)

Figure 4: Results of the localization algorithm considering the correct matches in *zone 1* using (a) Set 1 and (b) Set 2. The results of each description method are shown as different recall-precision curves.



(a) (b)

Figure 5: Results of the localization algorithm considering the correct matches in *zone 2* using (a) Set 1 and (b) Set 2. The results of each description method are shown as different recall-precision curves.

is usual to make use of any kind of probabilistic algorithm to estimate the position of the robot taking into account its previous estimated position. This is expected to provide a higher accuracy. We expect to develop this type of algorithms and tests in a future work.

Some additional conclusions can be reached by comparing the performance of the methods in open areas (Set 1) and urban areas (Set 2). In open areas, the performance of SURF-Harris, HOG, and *gist* is quite similar and relatively good in all cases, and the methods based on the Discrete Fourier Transform tend to present worse results. However, in the case of urban areas, SURF-Harris outperforms the other methods, and *gist* is the one that presents the worst results.

Apart from the localization accuracy, it is also important to study the computational cost of the process, since in a real application it would have to run in real time, as the robot is navigating through the environment. This way, we have obtained in all cases the necessary time to calculate the descriptor of the test image on the one hand and to compare it with the descriptors stored in the map, to detect the most similar descriptor and to analyze the results on the other hand. The average computational time of the localization process

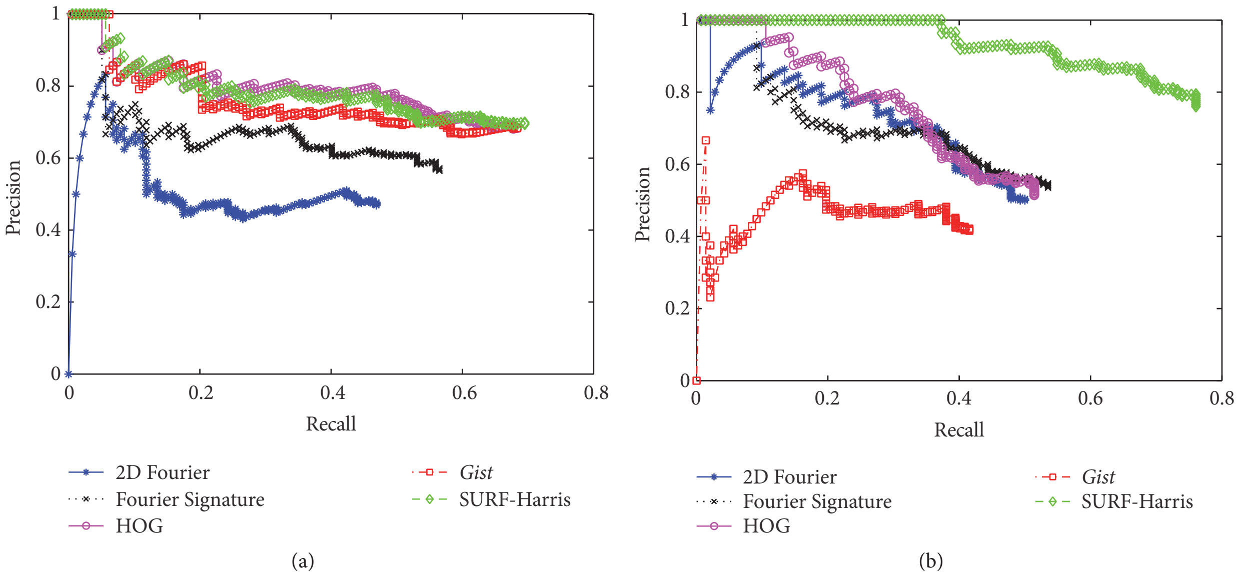


Figure 6: Results of the localization algorithm considering the correct matches in *zone 3* using (a) Set 1 and (b) Set 2. The results of each description method are shown as different recall-precision curves.

Table 1: Average computational cost of the description algorithms studied, per test image. For each description method and data set, the table shows first the necessary time to obtain the test image descriptor and second the time to compare it with the descriptors of the map and to obtain the final localization result.

	2D Fourier	Fourier Signature	*Gist*	HOG	SURF-Harris
Data Set 1 Descriptor	0.0087 s	0.0080 s	0.4886 s	0.0608 s	0.5542 s
Data Set 1 Match	0.0015 s	0.0058 s	0.0006 s	0.0008 s	25.8085 s
Data Set 2 Descriptor	0.0085 s	0.0079 s	0.4828 s	0.0621 s	0.5389 s
Data Set 2 Match	0.0012 s	0.0047 s	0.0005 s	0.0006 s	19.3931 s

after considering all the test images is shown in Table 1. To obtain the results of this table, the algorithms have been implemented using Matlab.

With respect to the computational cost, the methods based on the Fourier Transform are significantly faster than the rest, while SURF-Harris presents a considerably high computational cost. About the necessary time to compare two descriptors, *gist* and HOG are the fastest methods. In the case of SURF-Harris, the brute force match method implemented results in a relatively high computational cost. This method has been chosen to make a homogeneous comparison with the other global-appearance methods. However, in a real implementation, a bag-of-words based approach [23] would improve the computational efficiency of the algorithm.

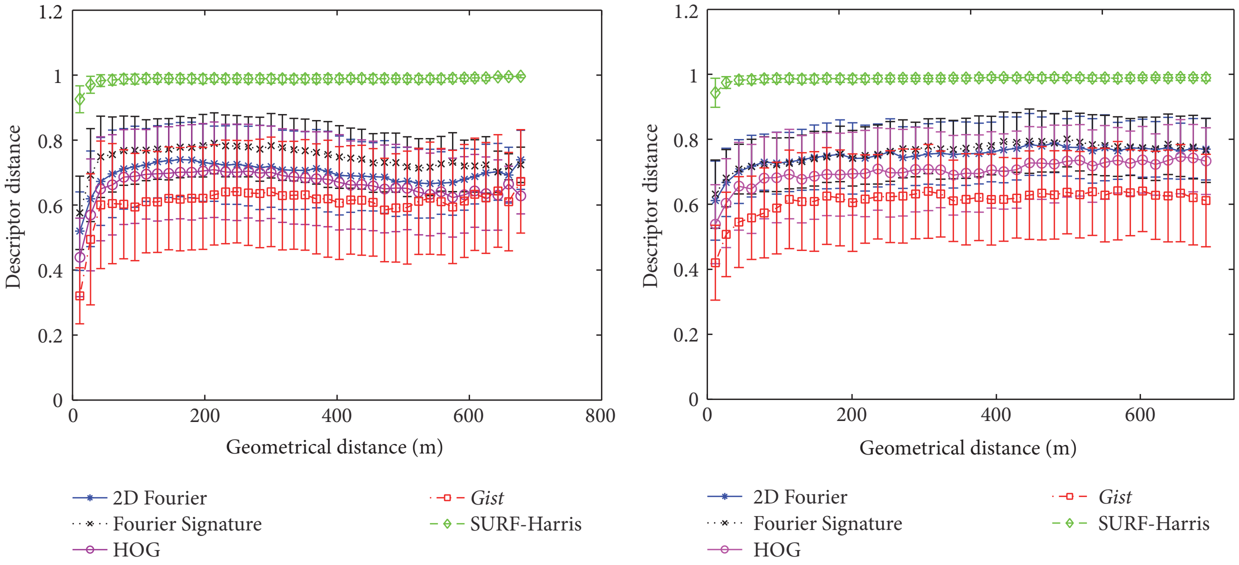
At last, we have obtained the average memory size needed to store each descriptor. The results are shown in Table 2. *Gist* is the most compact descriptor (it is able to compress the information in each scene significantly) while SURF-Harris needs more memory size.

Considering these results jointly with the precision in localization, we could say that the SURF-Harris descriptor shows very good results in location accuracy but its computational cost makes it unfeasible to solve a real application. HOG, which is the second in terms of accuracy, also has a very good computational cost, so we consider it interesting to study more thoroughly this descriptor as future work and to implement more advanced versions of this method to try to optimize the accuracy. Likewise, other types of distances to compare images could also be studied, apart from the Euclidean distance. For the same reasons, we also consider it appropriate to examine more thoroughly the *gist* descriptors, as well as using other methods to extract the *gist* of a scene apart from the orientation information (e.g., from the color information).

As a final experiment, we have studied the relationship *distance between two image descriptors* versus *geometric distance between the capture points of these two images*. As stated at the beginning of the section, this information is very interesting in some applications such as the construction of maps from the images, with geometric precision, or the localization of the vehicle at halfway points of the grid of the map. It is important that the distance between descriptors grows as the geometric distance does. Figure 7 shows the results obtained using (a) Set 1 and (b) Set 2. To obtain these figures, an image has been set as a reference image, and the distance between the reference image descriptor and the other descriptors has been calculated. The figure shows this distance versus the geometric distance between the capture points of each image and the capture point of the reference image. In both cases, this relationship is monotonically increasing up to a geometric

Table 2: Necessary memory to store each descriptor.

	2D Fourier	Fourier Signature	*Gist*	HOG	SURF-Harris
Descriptor	16384 bytes	32768 bytes	4096 bytes	8192 bytes	110400 bytes



(a)

(b)

Figure 7: Relationship *distance between two image descriptors* versus *geometric distance between the capture points of these two images* using (a) Set 1 and (b) Set 2.

distance of approximately 100 meters. From this point it tends to stabilize with a relatively high variance. The exception is the local features descriptor, which stabilizes at the final value from a very small geometric distance. However, the appearance-based descriptors exhibit a more linear behavior around each image.

6. Conclusions and Future Works

In this paper, we have carried out a comparative evaluation of several description methods of scenes, considering the performance of these methods to accurately solve an absolute localization problem in a large real outdoor environment. We evaluated two different approaches of visual descriptors, local features descriptors (SURF-Harris), and global-appearance descriptors (2D Discrete Fourier Transform, Fourier Signature, HOG, and *gist*).

All the tests have been carried out with images of Google Street View, captured under realistic conditions. Two different areas of a city have been considered, an open area and a purely urban area with narrower streets. The capture points of each area present different topography. The first one is a grid map that covers several streets and avenues and the second one a linear map (i.e., the images were captured when the mobile traversed a linear path on a narrow street).

Some different studies have been performed. First, we have evaluated the accuracy of the localization process. To do this, *recall and precision* curves have been computed to compare the performance of each description method. We plot the *recall and precision* curves for both areas, taking into account different levels of accuracy to consider that the localization result is correct. In these experiments, the computational cost of the localization process has also been analyzed.

We have also studied each descriptor in terms of behavior of the descriptor distance comparing to geometrical distance between image capture points. To do this, we plot a curve that represents the descriptor distance versus the geometrical distance between capture points. This measure is very useful for performing navigation tasks, since thanks to it we can estimate the range of use of the descriptor.

It is noticeable that the SURF-Harris descriptor is the most suitable descriptor in terms of precision in localization, but it presents a smaller zone of work in terms of Euclidean distance between descriptors. The HOG descriptor has shown a relatively good performance to solve the localization problem and presents a good response of the descriptors distance versus geometrical distance between capture points. If we analyze jointly the results of both experiments and take into account the computational cost (Tables 1 and 2) we conclude that, although the SURF-Harris descriptor presents the best results in terms of recall and precision curves, it does not allow us to work in real time. Therefore, taking into account that HOG is the descriptor that presents the second best results in terms of recall and precision curves and allows us to work in real time, we can conclude that the HOG is the most suitable descriptor.

We plan to extend this work to (a) capture a real outdoor trajectory traveling along several streets and capturing omnidirectional images using a catadioptric vision system, (b) combine the information provided by this vision system and the images of the Google Street View, and (c) evaluate the performance of the best descriptors in a probabilistic localization process.

Competing Interests

The authors declare that they have no competing interests.

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 Projects *AICO*/2015/021, *Localización y Creación de Mapas Visuales para Navegación de Robots con 6 GDL,* and *GV*/2015/031, *Creación de Mapas Topológicos a Partir de la Apariencia Global de un Conjunto de Escenas*.

References

[1] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust Monte Carlo localization for mobile robots," *Artificial Intelligence*, vol. 128, no. 1-2, pp. 99–141, 2001.

[2] A. Gil, O. Reinoso, M. A. Vicente, C. Fernández, and L. Payá, "Monte carlo localization using SIFT features," in *Pattern Recognition and Image Analysis*, vol. 3522 of *Lecture Notes in Computer Science*, pp. 623–630, 2005.

[3] A. Murillo, J. Guerrero, and C. Sagüés, "Surf features for efficient robot localization with omnidirectional images," in *Proceedings of the IEEE International Conference on Robotics & Automation*, San Diego, Calif, USA, 2007.

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: speeded up robust features," in *Proceedings of the 9th European Conference on Computer Vision*, Graz, Austria, 2006.

[5] F. Rossi, A. Ranganathan, F. Dellaert, and E. Menegatti, "Toward topological localization with spherical fourier transform and uncalibrated camera," in *Proceedings of the International Conference on Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR '08)*, pp. 319–330, Venice, Italy, 2008.

[6] L. Payá, O. Reinoso, F. Amoros, L. Fernández, and A. Gil, "Probabilistic map building, localization and navigation of a team of mobile robots. application to route following," in *Multi-Robot Systems: Trends and Development*, pp. 191–210, 2011.

[7] L. Fernández, L. Payá, D. Valiente, A. Gil, and O. Reinoso, "Monte Carlo localization using the global appearance of omnidirectional images: algorithm optimization to large indoor environments," in *Proceedings of the 9th International Conference on Informatics in Control, Automation and Robotics (ICINCO '12)*, pp. 439–442, Rome, Italy, July 2012.

[8] M. Montemerlo, *FastSLAM: a factored solution to the simultaneous localization and mapping problem with unknown data association [Ph.D. thesis]*, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pa, USA, 2003.

[9] C. Gamallo, P. Quintía, R. Iglesias-Rodríguez, J. V. Lorenzo, and C. V. Regueiro, "Combination of a low cost GPS with visual localization based on a previous map for outdoor navigation," in *Proceedings of the 11th International Conference on Intelligent Systems Design and Applications (ISDA '11)*, pp. 1146–1151, Cordoba, Spain, November 2011.

[10] A. Torii, J. Sivic, and T. Pajdla, "Visual localization by linear combination of image descriptors," in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCV Workshops '11)*, pp. 102–109, Barcelona, Spain, November 2011.

[11] M. Aly and J.-Y. Bouguet, "Street view goes indoors: automatic pose estimation from uncalibrated unordered spherical panoramas," in *Proceedings of the IEEE Workshop on the Applications of Computer Vision (WACV '12)*, pp. 1–8, Breckenridge, Colo, USA, January 2012.

[12] A. Taneja, L. Ballan, and M. Pollefeys, "Registration of spherical panoramic images with cadastral 3d models," in *Proceedings of the International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT '12)*, pp. 479–486, Zurich, Switzerland, 2012.

[13] F. Amorós, L. Payá, O. Reinoso, and L. Jiménez, "Comparison of global-appearance techniques applied to visual map building and localization," in *Proceedings of the International Conference on Computer Vision Theory and Applications*, pp. 395–398, Rome, Italy, 2012.

[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] L. Payá, L. Fernandez, O. Reinoso, A. Gil, and D. Ubeda, "Appearance-based dense maps creation. Comparison of compression techniques with panoramic images," in *Proceedings of the International Conference on Informatics in Control, Automation and Robotics (INSTICC '09)*, pp. 238–246, Milan, Italy, 2009.

[16] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of the Alvey Vision Conference*, pp. 23.1–23.6, Manchester, UK, 1988.

[17] 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.

[18] L. Payá, L. Fernandez, A. Gil, and O. Reinoso, "Map building and Monte Carlo localization using global appearance of omnidirectional images," *Sensors*, vol. 10, no. 12, pp. 11468–11497, 2010.

[19] A. Friedman, "Framing pictures: the role of knowledge in automatized encoding and memory for gist," *Journal of Experimental Psychology: General*, vol. 108, no. 3, pp. 316–355, 1979.

[20] A. Torralba, "Contextual priming for object detection," *International Journal of Computer Vision*, vol. 53, no. 2, pp. 169–191, 2003.

[21] 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)*, pp. 886–893, Washington, DC, USA, June 2005.

[22] F. Amorós, L. Payá, O. Reinoso, L. Fernández, and J. Marín, "Visual map building and localization with an appearance-based approach—comparisons of techniques to extract information of panoramic images," in *Proceedings of the 7th International Conference on Informatics in Control, Automation and Robotics*, pp. 423–426, 2010.

[23] D. Filliat, "A visual bag of words method for interactive qualitative localization and mapping," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '07)*, pp. 3921–3926, Roma, Italy, April 2007.