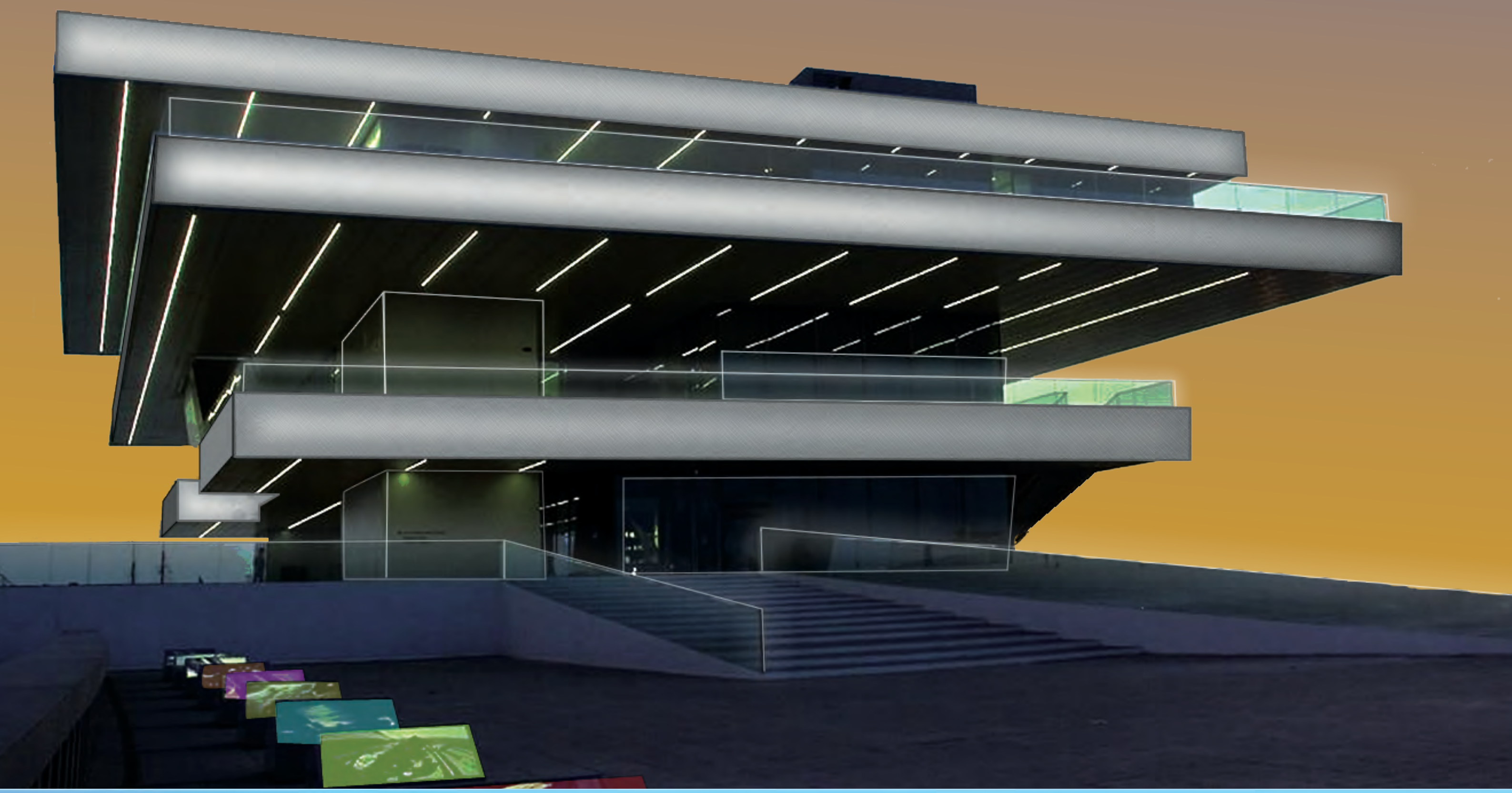
# INTED 2013

## 7th International Technology, Education and Development Conference

Valencia (Spain), 4th - 6th of March, 2013.

# CONFERENCE PROCEEDINGS



www.inted2013.org

# INTED 2013

## 7th International Technology, Education and Development Conference

Valencia (Spain), 4th - 6th of March, 2013.

# CONFERENCE PROCEEDINGS

**Published by**
International Association of Technology, Education and Development (IATED)
www.iated.org

**INTED2013 Proceedings**
7th International Technology, Education and Development Conference
March 4th-6th, 2013 — Valencia, Spain

**Edited by**
L. Gómez Chova, A. López Martínez, I. Candel Torres
International Association of Technology, Education and Development
IATED

**ISBN: 978-84-616-2661-8**
**Depósito Legal: V-424-2013**

Book cover designed by
J.L. Bernat

# WELCOME INTRODUCTION

**Dear INTED2013 participants,**

We are delighted to welcome you all to this 7th International Technology, Education and Development Conference.

Today, we are living in a technology-based society where education and innovation are the key to the world's development and progress. For this reason, it is essential to be updated with new teaching and learning methodologies and explore new horizons in educational cooperation.

INTED2013 welcomes experts from more than 65 different countries. This will be the best occasion to open your minds to other experiences and share best practices in education. This forum will also provide an ideal networking platform to make new international contacts and partnerships.

We hope you benefit from the conference programme, its interactive sessions and social activities where you will be able to meet other educators, researchers and technologists from all continents and cultures.

Additionally, we invite you to discover and enjoy the beautiful city of Valencia. Do not miss the opportunity to walk around and visit its impressive architecture, historical buildings, green areas and lovely beaches.

Thank you very much for participating in INTED2013 and for contributing to this inspiring international forum.

We hope you enjoy your time with us!

***INTED2013 Organising Committee***

# INTED2013 COMMITTEE AND ADVISORY BOARD

# CONFERENCE SESSIONS

## ORAL SESSIONS, 4th March 2013.

Mobile Learning
Joint Educational and Research projects
Collaborative and Problem-based Learning
University-Industry Collaboration (1)
Technological Issues in Foreign Languages Education (1)
Evaluation and Assessment of Student Learning
Impact of Education on Development

Research in Technology in Education
International Projects
Collaborative Experiences in Web Technologies
University-Industry Collaboration. Workplace Learning
Technological Issues in Foreign Languages Education (2)
Virtual Universities
Diversity and Ethical issues in Education

Educational Software and Serious Games
Curriculum Design. Strategies, Principles and Challenges
Collaborative and Virtual Learning Environments
University-Industry Collaboration (2)
Experiences in Foreign Languages Education
ICT Skills and Competencies among Teachers
Experiences in Business & Adm. Mgmt. Education

Blended Learning
New Experiences for Curriculum Design
Learning and Teaching Methodologies (1)
Adult and Vocational Education
Experiences in Foreign Languages Education & Research
Pre-service Teacher Experiences
Technological Issues in Business Adm. & Mgmt. Education

## POSTER SESSIONS, 4th March 2013.

Poster Session 1. Research and General Issues in Education

Poster Session 2. Pedagogical Innovations and New Trends in Education

**ORAL SESSIONS, 5th March 2013.**

Web 2.0 and Social Networking (1)
New Trends in the Higher Education Area
Quality assurance in Education (1)
Research in Education
Experiences in Engineering Education
Experiences in Biosciences Education & Research
Barriers to Learning

Web 2.0 and Social Networking (2)
New projects and innovations in Education (1)
Quality assurance in Education (2)
Virtual Universities and Distance Education
Research in Engineering Education
Pedagogical Innovations and Experiences in Health
Enhancing learning and the undergraduate experience (1)

Social media & Social networks in Education
New projects and innovations in Education (2)
New challenges for the Higher Education Area (1)
Primary & Secondary Education
Technological Issues in Engineering Education
Technology-Enhanced Learning in Health Education
Enhancing learning and the undergraduate experience (2)

Technology-Enhanced Learning
Learning and Teaching Methodologies (2)
New challenges for the Higher Education Area (2)
Learning Experiences in Primary and Secondary Education
Language Learning Innovations (1)
Technological Issues in Education
Pedagogical & Didactical Innovations in Architecture Education

E-learning
General Issues. Organizational, legal and financial issues
New challenges for the Higher Education Area (3)
Educational Research
Language Learning Innovations (2)
E-content Management and Development
Education and Globalization

**POSTER SESSIONS, 5th March 201312.**

Poster Session 1. Technology in Education and University-Industry Collaboration

Poster Session 2. Experiences in Education and Curriculum Design

## VIRTUAL SESSIONS

Barriers to Learning, Digital Divide & Diversity issues in Education
Collaborative and Problem-based Learning
Competence Evaluation
Computer Supported Collaborative Work
Curriculum Design and Innovation
E-content Management and Development
E-learning, M-learning and Blended Learning
Educational Software and Serious Games
Enhancing learning and the undergraduate experience
Evaluation and Assessment of Student Learning
Experiences in Education
Experiences in Research in Education
Globalization and Ethical issues in Education
International Projects
Learning and Teaching Methodologies
Learning Experiences in Primary and Secondary School
Lifelong Learning
Links between Education and Research
New challenges for the Higher Education Area
New projects and innovations
New Trends in the European Higher Education Area
Pedagogical & Didactical Innovations
Quality assurance in Education
Research Methodologies
Research on Technology in Education
Student Support in Education
Technological Issues in Education
Technology-Enhanced Learning
Tutoring and Coaching
University-Industry Collaboration
Virtual Universities and Distance Education

# ABOUT INTED2013 Proceedings CD

## HTML Interface: Navigating with the Web browser

This CD includes all presented papers at INTED2013 conference. It has been formatted similarly to the conference Web site in order to keep a familiar environment and to provide access to the papers trough your default Web browser (open the file named " INTED2013.html").

An Author Index, a Session Index, and the Technical Program are included in HTML format on this disk to aid you in finding conference papers. Using these HTML files as a starting point, you can access other useful information related to the conference.

The links in the Session List jump to the corresponding location in the Technical Program. The links in the Technical Program and the Author Index open the selected paper in a new window. These links are located on the titles of the papers and the Technical Program or Author Index window remains open.

## Full Text Search: Searching INTED2013 index file of cataloged PDFs

If you have Adobe Acrobat Reader version 6 or later (www.adobe.com), you can perform a full-text search for terms found in INTED2013 proceedings papers.

*Important:* To search the PDF index, you must open Acrobat as a stand-alone application, not within your web browser, i.e. you should open directly the file "INTED2013.pdf" in the CD with your Adobe Acrobat or Acrobat Reader application.

This PDF file is attached to an Adobe PDF index that allows text search in all PDF papers by using the Acrobat search tool (not the same as the find tool). The full-text index is an alphabetized list of all the words used in the collection of conference papers. Searching an index is much faster than searching all the text in the documents.

*To search the INTED2013 Proceedings index:*

1. Open the Search PDF pane through the menu "Edit > Advanced Search" or click in the PDF bookmark titled "SEARCH PAPERS CONTENT".
2. The "INTED2013_index.pdx" should be the currently selected index in the Search window (if the index is not listed, click Add, locate the index file .pdx on the CD, and then click Open).
3. Type the search text, click Search button, and then proceed with your query.

*For Acrobat 9:*

1. In the “Edit” menu, choose “Search”. You may receive a message from Acrobat asking if it is safe to load the Catalog Index. Click “Load”.
2. A new window will appear with search options. Enter your search terms and proceed with your search as usual.

*For Acrobat 8:*

1. Open the Search window, type the words you want to find, and then click Use Advanced Search Options (near the bottom of the window).
2. For Look In, choose Select Index.
3. In the Index Selection dialog box, select an index, if the one you want to search is available, or click Add and then locate and select the index to be searched, and click Open. Repeat as needed until all the indexes you want to search are selected.
4. Click OK to close the Index Selection dialog box, and then choose Currently Selected Indexes on the Look In pop-up menu.
5. Proceed with your search as usual, selecting other options you want to apply, and click Search.

*For Acrobat 7 and earlier:*

1. In the “Edit” menu, choose “Full Text Search”.
2. A new window will appear with search options. Enter your search terms and proceed with your search as usual.

# AN EDUCATIONAL SOFTWARE TO COMPARE APPEARANCE IMAGE DESCRIPTORS IN ROBOT LOCALIZATION

**L. Payá, F. Amorós, O. Reinoso, L. Fernández, A. Gil**

*Departamento de Ingeniería de Sistemas y Automática, Miguel Hernández University (SPAIN)*
*lpaya@umh.es, famoros@umh.es*

## Abstract

In the field of mobile robots, the task of estimating the position of the robot in the environment where it evolves is a very important step during the design of any application where the robot must move autonomously. All the features concerned in localization are deeply studied in advanced subjects about robotics. As it is a topic in constant innovation as new methods are continuously appearing, and some of these methods may be mathematically complex, it is necessary to provide students with a tool that allows them to understand how these algorithms work and how they can configure them to optimize the localization process.

In this work, we present a tool we have developed to be used in computer vision and robotics subject. Using this tool, the student will be able to test and understand many localization concepts in robotics, using visual information and appearance-based methods. This tool offers an easy and intuitive graphical interface that guides the student through the configuration of the algorithms. Some databases have been added to this tool, composed of several sets of real indoor images, captured in a real environment under realistic lighting conditions. Thanks to this software, the students can learn how the appearance-based approach works and the importance of a good choice of the parameters. It will allow the student to develop more complicated applications in the map building and localization field.

Keywords: Computer vision, mobile robots, appearance-based methods, educational software, higher education.

## 1 INTRODUCTION

When a mobile robot has to carry out a task autonomously in an unknown environment, it needs an internal representation or map of the environment. Using the information perceived by the sensors it is equipped with, and comparing it with the information stored in the map, the robot must be able to estimate its position in the environment and the necessary control action to go to the target points to complete its task.

Very often, the robot uses the information captured by a camera mounted on it as input data to solve the map building and localization processes. Also, it is usual to work with omnidirectional cameras, which provide panoramic images of the environment around the robot [1]. These cameras constitute a good option in localization tasks due to the richness of the information they offer and their relatively low cost comparing to other kinds of sensors.

The research developed in the topic of map creation and localization using visual information during the last years is enormous, and new algorithms are published continuously. This way, both computer vision and robotics constitute two fields in Engineering which are in a continuous development and that permit solving with robustness tasks that imply the autonomous movement of a robot in a real environment [2].

In a subject of our PhD program, in the Miguel Hernandez University (Spain), there is an optional subject where these topics are studied. In this subject, students have to learn the different strategies that can be used to achieve these aims, using as input information the images that an omnidirectional camera, which is mounted on the robot, provides. Among these strategies, the appearance-based approach attracts the interest of researchers mainly due to its simplicity and robustness, and its ability to work in unstructured environments. In these methods, the information in the scenes is used as a whole, without any feature extraction thus, localization is achieved based on the matching of the whole images. Taking these concepts into account, the approach consists of two steps:


*ISBN: 978-84-616-2661-8*

1. First, the robot goes through the unknown environment, while it captures some images. It is usual that a new image is acquired when it is different enough from the previously captured one. As images are very high dimensional data, it is very important to extract the most relevant information before storing them. Several techniques may be used with this aim. In this educational software, we have implemented some of the most important appearance-based techniques: Fourier-based approaches, Principal Components Analysis (PCA) approaches and gradient-based approaches. So, the student will be able to test them and he/she can compare in which situations each of the approaches works better and how they must be configured to optimize their performance.
2. Once the map is finished, the robot will be able to compute its location within it. With this goal, the robot has to capture a new image, compress it and compare it with the data previously stored in the map. As a result, the current position and orientation of the robot should be computed. With this software, the student can test different techniques to perform localization.

We have included some databases of images captured in several real indoor environments, under realistic lighting conditions, so that the student can test the mapping algorithms. Also, some intermediate images are included so that he/she can test the localization algorithms.

We are aware of the fact that sometimes, students get lost in the classroom as the algorithms they study can be mathematically quite complex. That is the reason why we have developed this interactive tool. We expect it is useful for the students to fully understand the appearance-based methods and other basic computer vision and robotics concepts. As it provides real data, we expect that they learn how to face the problems that could appear in a realistic application and to design new algorithms and improve the existing ones.

The tool is fully interactive. Both the mapping and the localization processes are fully configurable, so that student can test how a correct tuning of all the parameters is very important to obtain acceptable results.

The remainder of the paper is structured as follows; section 2 presents the principles the students study in the theoretical classes. In section 3 we outline the map building and localization process in a real application. In section 4 we detail how the software tool works. Section 5 presents some implementation details and, at last, in section 6, we present the conclusions of the work.

# 2 APPEARANCE-BASED METHODS IN ROBOTICS. MAP BUILDING AND LOCALIZATION

An appearance-based approach for robot navigation implies using the information of the whole images, without extracting any kind of landmark. If we have a map of the environment containing several scenes from it, during the navigation the robot has to capture an image and compare it with those in the map to know its localization. If this auto-location process were carried out with high resolution images, the computational cost of the necessary operations would be unbearable. However, it is possible to compress the information of the image with a feature extraction method.

In visual object recognition systems, different techniques have been used to reduce the dimensionality of the data, allowing the classifier to work on a smaller number of attributes. In the tool, we have included some of the main compression techniques: PCA (Principal Component Analysis) [3], Fourier transform-based techniques [4], HOG [5] and GIST approaches [6]. In the next paragraphs, we make a brief description of these methods.

## 2.1 Fourier-based techniques

When working with panoramic images, a Fourier-based compact representation can be used [4]. It consists in expanding each row of the panoramic image using the Discrete Fourier Transform. After this process, each image is represented by a vector $d_i, i \in \{1, \ldots, n\}$ where $n$ is the number of images of the environment to map. The size of each vector is $mxk_1$, where $m$ is the number of rows of the panoramic images and $k_1$ is the number of Fourier components retained to build the descriptor. $k_1$ is a configurable parameter. The larger is $k_1$, the most information is retained from the scenes but the higher is the computational cost.

This Fourier Signature presents two interesting properties. The most relevant information concentrates in the low frequency components of each row, and it presents rotational invariance. These rotations lead to two panoramic images that are the same but shifted along the horizontal axis. The biggest advantage of the Fourier methods comparing to PCA is that the first offer rotational invariance, what is very interesting in robot localization applications.

## 2.2 PCA-Based Techniques

The Principal Components Analysis makes possible the representation of a set of *n* images with *MxN* pixels each, in a new subspace where the images are transformed in a feature vector, known as projection of the image, $\vec{p}_i \in \Re^{k_2 x1}, i = 1,\ldots,n$, being $k_2$ the PCA features that preserve the most of the variance of the database [3]. However if we apply PCA directly over the set of panoramic images, then we will obtain a map without rotational invariance, since it keeps information of one orientation for each scene, but no for other possible orientations. A possible solution to this problem consists in applying PCA over the Fourier Signature of the scene. This presents rotational invariance and we get a double compression effect.

## 2.3 HOG (Histogram of Oriented Gradient) Techniques

The Histogram of Oriented Gradient (HOG) descriptors [5] are based on the orientation of the gradient in local areas of an image. The first step to apply HOG to an image is to compute the spatial derivatives of the image along the *x* and *y*-axes ($U_x$ and $U_y$). These derivatives can be obtained by calculating the convolution of the images with Gaussian filters with different variance. Once the convolution of the image is made, we can get the magnitude and direction values of the gradient at each pixel:

$$|G| = \sqrt{U_x^2 + U_y^2} \qquad \theta = \arctan\left(U_x / U_y\right)$$

After that, the orientation binning is computed by dividing the image in cells, and creating the histogram of each cell. The histogram is computed based on the gradient orientation of the pixels within the cell, weighted with the corresponding module value. The number of histogram divisions is 8 in our experiments, and the orientation angle varies between *-90º* and *90º*. Each image is represented through the histogram of every cell ordered into a vector.

An omnidirectional image contains the same pixels in a row although the image is rotated, but in a different order. We can take profit of this characteristic to carry out the location of the robot by means of calculating the histogram with cells having the same width as the image (fig. 1, up). This way, we obtain an array of rotational invariant features.

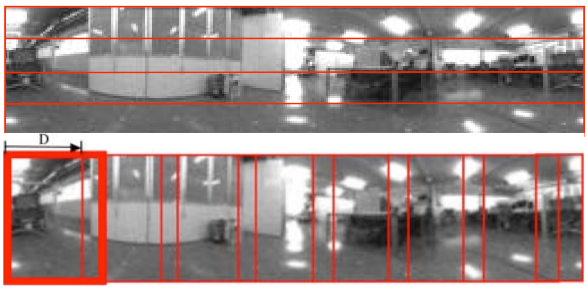



Figure 1. Cells used to compute the current position and the orientation of the robot with HOG.

However, to know the relative orientation between two rotated images, vertical windows (cells) are used, with the same height of the image, and variable width and distance (fig. 1, down). Arranging the histograms of these windows in a different way, we obtain the same results as calculating the histogram of an image rotated a proportional angle to the *D* distance. The angle resolution that can be computed between two shifted images is proportional to that distance:

$$Angle(^\circ) = \frac{D * 360}{Width \quad of \quad the \quad image}$$

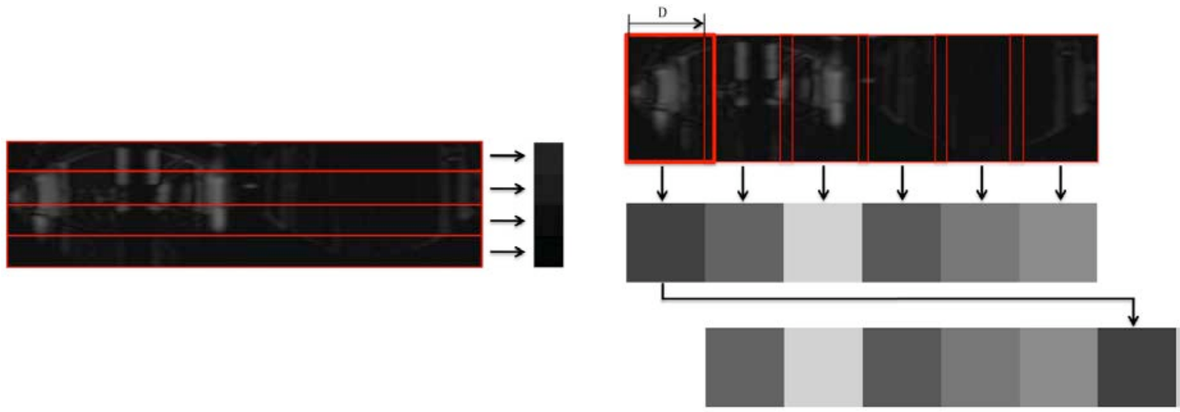
## 2.4 GIST-Based Techniques

The GIST of a scene is defined as an abstract representation that activates the memory of scenes' categories [7]. The descriptors based on GIST try to obtain the essential information of the image simulating the human perception system through its ability to recognise a scene through the identification of colour or remarkable structures, avoiding the representation of specific objects.

Some authors have studied the problem of image categorization considering features based on the human ability to recognise images [6,8]. These features are spatial frequencies in different scales based on Gabor filtering. The descriptor we propose is called GIST-Gabor because it is based in Gabor filtering masks [9] in order to obtain frequency and orientation information.

The first step to build the descriptor consists in creating a bank of the Gabor masks with different resolutions and orientations. After that, we filter the image with the set of masks. The orientation of the filter depends on the number of masks of each level because they are equally distributed between 0 and 180 degrees. The filtering is done in the frequency domain, so we compute the bidimensional Fourier Transform of the scene, and multiply each element of the image with its corresponding element of the mask. The results encode different structural information depending on the mask used.

An omnidirectional image contains the same pixels in a row although the image is rotated. So, to create the descriptor, we calculate the average pixel's value within cells with the same width as the image (fig. 2a). We repeat this operation for every filtered image, obtaining an array of rotational invariant characteristics. To know the relative orientation between two rotated images, we use vertical windows. These vertical windows have the same height of the panoramic image. The distance between vertical windows and their width are variables (Fig. 2b).




(a) (b)

Figure 2. Extraction of the Descriptor Values from a filtered image for (a) location and (b) phase estimation.

# 3 DESCRIPTION OF THE EXPERIMENTS

## 3.1 Database creation

To build the databases in this application, we have worked with a Pioneer P3-AT (fig. 3a) robot that is equipped with a catadioptric system (fig. 3b), consisting on a forward-looking camera and a parabolic mirror that provides omnidirectional images of the environment. To work with this information in an efficient way, the omnidirectional images are transformed to panoramic images, as shown of fig. 3c, with a size of *64x256* pixels.

To carry out the experiments, we have captured a set of omnidirectional images on a pre-defined *40x40* cm grid in several indoor environments. Fig. 4 shows a bird's eye view of the grid used to take the images and several examples of panoramic images.

(a) (b) (c)

Figure 3. (a) Pioneer P3-AT robot, (b) catadioptric system and (c) omnidirectional and panoramic image.

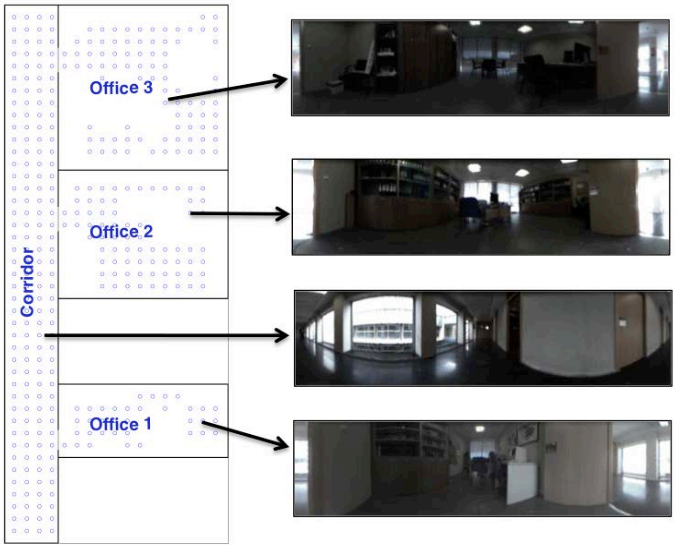



Figure 4. Bird eye's view of the grid where the images were taken, with several examples of panoramic images.

## 3.2 Localization and orientation of the robot

To test the validity of the previous maps, the robot has captured several test images in some half-way points among those stored in the map. We have captured several sets of test images. The first one, at the same time we took the training set and the rest of them in different times of the day (with varying illumination conditions) and, as we deal with a real working environment, with changes in the position of some objects. The objective of this section is to compute the position and the orientation of the robot when it took the test images. It must be carried out just comparing the visual information with the information in the maps. This process is carried out in the following way, depending on the descriptor used:

### *3.2.1 PCA-based techniques.*

The PCA map is made up of the matrix $V \in R^{KxM}$, which contains the *K* main eigenvectors and the projections of the training images $\vec{p}^{\,j} \in R^{Kx1}; j = 1 \ldots N$ (one per position).

To compute the location where the robot took each test image, we have to project the test image onto the eigenspace. Once the projection $\vec{p}^{\,i} \in R^{K_1 x1}$ is obtained, it must be compared with all the vectors $\vec{p}^{\,j}$ stored in the map. The criterion used is the Euclidean distance. The image of the database that presents minimum value of the distance is the corresponding one.

$$d_{ij} = \sqrt{\sum_{l=0}^{k_1} \left(p^{il} - p^{jl}\right)^2}; \quad j = 1 \dots N$$

### 3.2.2 Fourier-based techniques.

To compute the position and orientation of the robot for each test image, we compute the Fourier Signature and then, we compute the Euclidean distance of the power spectrum of the test image with the spectra stored in the map. The best match is taken as the current position of the robot.

On the other hand, the orientation is computed with the shift theorem. We obtain a different angle per row so we have to compute the average angle.

### 3.2.3 HOG-based techniques.

When a new image arrives, first, its histogram of oriented gradient is computed using cells with the same size as when the map was built. So, the time needed to find the pose of the robot varies depending on both vertical and horizontal cells. To find the position of the robot, the horizontal cells information is used, whereas to compute the phase it is necessary to use the information in the vertical cells. In both cases, the information is found by calculating the Euclidean distance between the histogram of the new image and the stored ones in the map.

### 3.2.4 GIST-based techniques.

The location is estimated by calculating the minimum distance between the horizontal cells' descriptor of the database and the current image.

# 4 DESCRIPTION OF THE TOOL

In this section, we present a detailed description of the interface operations. Our objective while designing it was to simplify the understanding of global appearance descriptors in mapping and localization tasks. In this sense, all the parameters of the different techniques are configurable. The distribution of images in the database, the position of the test images and the distance of the test image descriptor are represented graphically, and we also include localization results and computational cost information of both map building and localization process so that the user can make a deep comparison between the techniques.

## 4.1 User's Manual

Our Graphic User Interface has been created with the tool GUIDE of MATLAB [10]. Brief details of the implementation are included in section 5. For that reason, this application must run under MATLAB. So then, first we have to launch MATLAB, add the folder where the GUI is, and call the interface by writing its name in the command window.

In Fig. 5 we can see the appearance of the application during an experiment. In the left upper quadrant of the window, we have the menu to select the descriptor and its variables. Under this menu, in the middle left side, we can select the test image to carry out a localization experiment when the map is created. Once we have compared the test image descriptor with the map, the test image and the map image selected as most similar are showed below. In the right side of the application, a map with the distribution of the database images is shown. When doing a localization experiment, we place the test image in the same plot. After the localization, it also appears a representation of the similarity in image's distance with each map element. Under both maps, it is shown the computational cost of the map building and the test image localization respectively. The next subsections describe the options and the way to use the application.

### 4.1.1 Map Building

First of all, we have to create the map where localizing the test images. Our database is made up of 363 colour panoramic images distributed in a 40x40cm grid along 4 different areas. These areas correspond to an indoor environment. Specifically, there are three offices and the corridor that connects them. In the first step, the student has to choose the technique to apply in order to create the map. There are 5 possibilities. Depending on the descriptor selected, the variables concerning the location and orientation parameters change. Fig. 6 shows the configuration of the location and orientation parameters for all the descriptors.

The value of every variable can be chosen using pop-up menus. The range of values is predefined. If any parameter is not changed, the application uses the default values that appear indicated. After selecting the desired parameters, we can generate the map.

Once the map is created, a panel with an indicative message is shown below the image distribution map. This panel includes information about the time and memory the map requires.

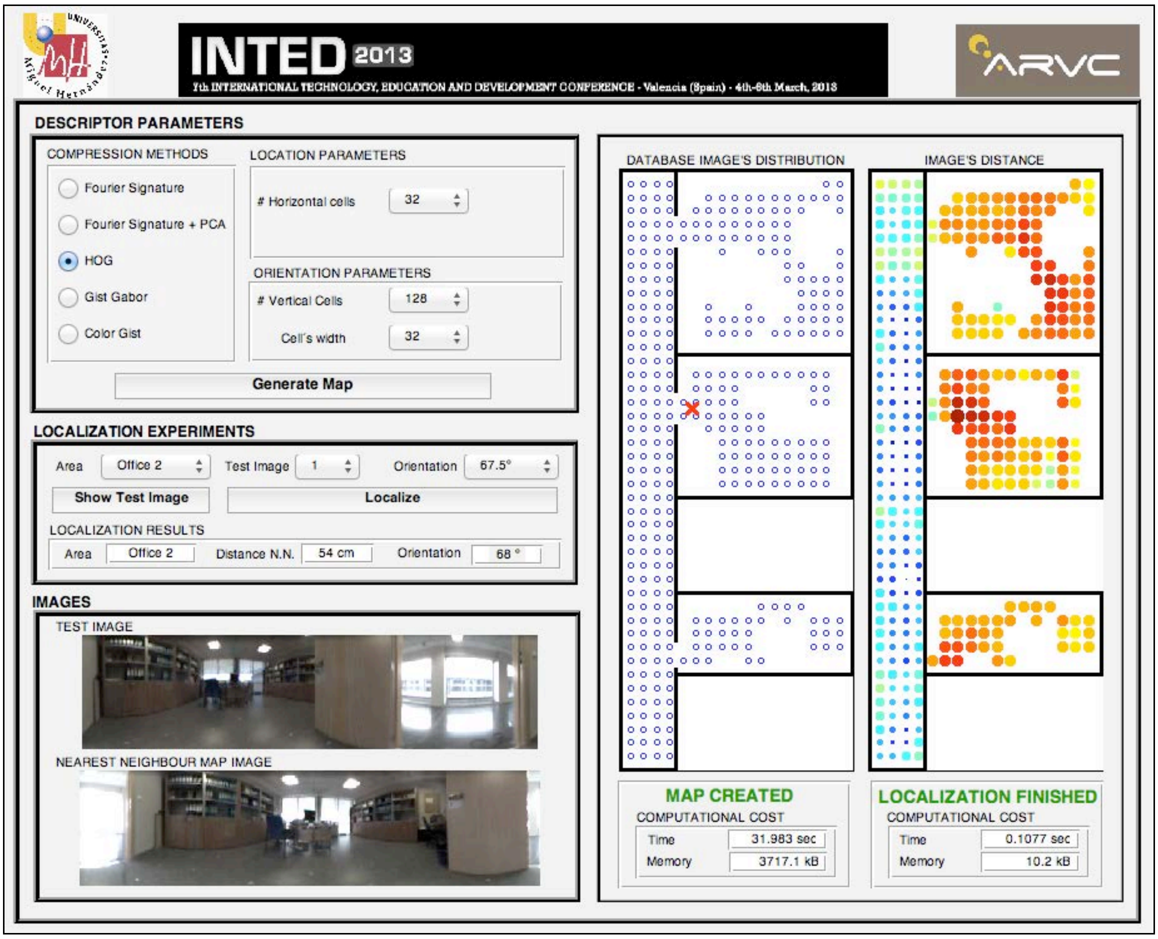



Figure 5. Capture of the Graphical User Interface application during an experiment.

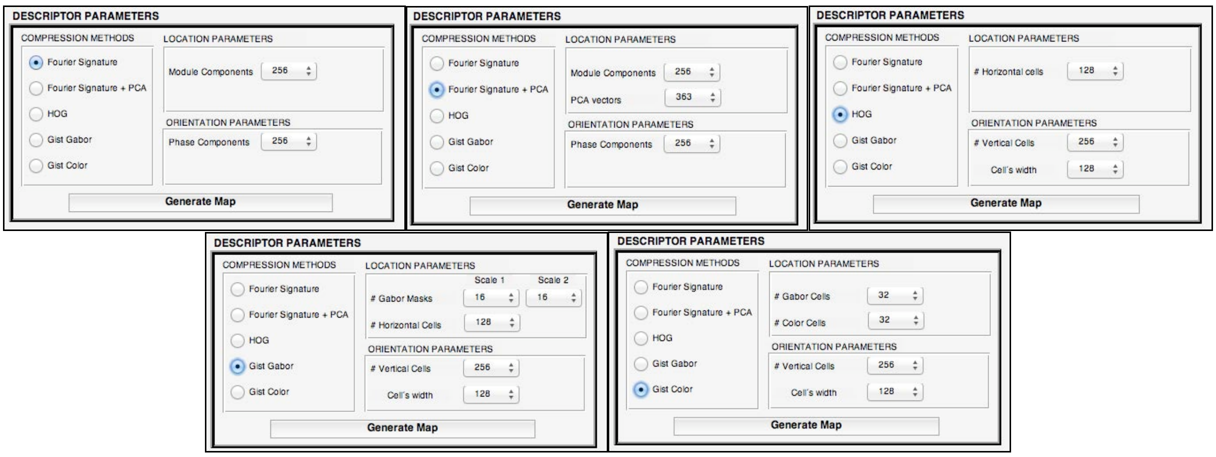



Figure 6. Location and orientation parameters for each descriptor.

#### 4.1.2 Localization of the robot

The next phase of the experiment lies in localizing test images in the map created previously. The compression method and parameters used are the same as in the map building. Until the map is successfully created, the localize button is disabled.

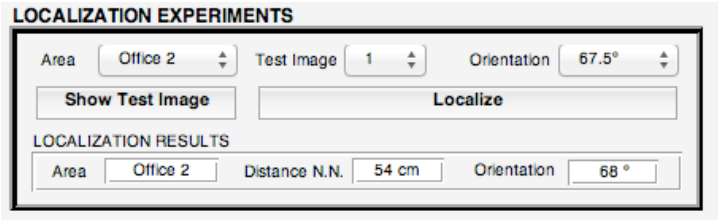



Figure 7. Localization experiments menu.

The aim is to let the student check the computational cost and precision of each global appearance descriptor by doing the descriptor retrieving with the map.

The image test set is composed of 9 positions per area, with 16 orientations per position (every 22.5º), with a total of 546 different examples. In the interface, the student can select the area, the number of test image, and its orientation.

If the user presses the button ‘Show Test Image’, the application reads the test image and its position in the map. In the interface, there are some changes:

- In the plot with the database images distribution appears the position of the test image, indicated with a red x-mark. Before that change, the name of the areas appears printed over each one.
- The test image is shown in the bottom left.

If the user presses the button ‘Localize’, the application computes the descriptor of the image test, and estimates the Euclidean distance with all the map images. Next, it estimates the orientation of the robot by computing the phase lag of test image with the Nearest Neighbour of the map, i.e., the position of the map that presents the minimum Euclidean distance. These results are presented graphically in the interface by adding the following elements:

- In the panel of ‘Localization Experiments’ (fig. 7), it is indicated the area of the Nearest Neighbour location, the distance between that image and the test image, and the estimated orientation.
- Under the test image, we include the image of the Nearest Neighbour in the bottom left of the window. That way, we facilitate the comparison between both images in order to appreciate whether the location has been correct and its phase lag.
- On the right side of the application, we plot the test image distance with every image of the map. In each element position, a coloured circle is displayed. The colour and size of the circles represent the similarity between images. Bigger red dots indicate higher similarity, whereas smaller blue markers correspond with the less similar scenes.
- On the bottom right, we indicate in a new panel when the localization has finished, and the computational requirements of the process.

Although we would not show the test image before localizing, when we press the ‘Localize’ button, the same actions as when we press ‘Show Test Image’ are executed.

When a different compression method is selected, or any of the descriptor parameters modified, all the elements included during the localization routine disappear, and the button ‘Localize’ is disabled until the map is generated again.

## 5 IMPLEMENTATION DETAILS

In this section, we focus on the tools and main functions used and implemented in order to build our application. Due to the inherent mathematical operations which are necessary to estimate the different image descriptors, we need a numerical computing software program that can carry out the operations. For that reason, we choose MATLAB. This tool, apart from offering a wide range of matrix operation functions, also offers image file operations, plotting data actions, and has an interactive application to design graphics user interfaces.

## 5.1 MATLAB

MATLAB (Matrix Laboratory) [10] is a numerical computing program created by Cleve Moler in the late 1970's. MATLAB has his own programming high-level language. There are a large number of areas where MATLAB can be used, such as communications systems, computational biology, control systems, or image and video processing, existing toolboxes developed for each area. It is specially indicated when working with data arrays, as images.

The main advantages of using MATLAB are the mathematical functions the program offers, which allow among other possibilities the resolution of linear algebra problems, statistics analysis of data, graphic representation of functions and variables, or the Fourier transform of the data. It has been especially interesting its tool for building graphical interfaces applications.

## 5.2 GUI

As stated above, MATLAB has his own tool in order to design graphical user interfaces. It is called GUIDE (Graphical User Interface Development Environment). GUIDE is an interactive platform where GUIs can be created and customized, offering different ways of interface control, such as buttons, check boxes, sliders or pop-up menus.

When a new GUI project is created, two files appear automatically. The first has the extension *.fig*, and it corresponds with the graphic layout of the interface. The GUIDE Layout Editor permits the inclusion of new control elements to the interface, the design of the elements distribution and properties, or its appearance modification in an interactive way. The second file is a *.m* and it includes the commands that define the routines that the interface runs when the user interacts with the controls. These routines are called callbacks. This way, we can run previously defined functions when necessary and create plots to show the results in the interface.

## 5.3 Main operations

Although we have included several different functions in the code, a few of them are common in the great majority of functions. In this section, we highlight the most important MATLAB functions used.

Among them, the image files operations are the most executed. The instruction *imread.m* reads an image from the file indicated in the string we write as a parameter. We have to indicate the whole pathname. As a result, we obtain a MxN array with the values of the intensity of a grayscale image, or a MxNx3 if we are reading a colour image, separating the R,G,B channels. Our database is composed of colour images. Some of the global appearance techniques we work with need a grayscale image as an input. With *rgb2gray.m*, we convert a RGB image to grayscale format.

Two of the descriptors work with the unidimensional Fourier Transform. The function *fft.m* computes the FFT (Fast Fourier Transform) of an input vector. Other technique is based in the PCA projection of the data. The function *eig.m* produces the eigenvalues and the corresponding eigenvectors given a matrix.

To perform the HOG descriptor, we need to compute the gradient of the image. In that task, we use *conv2.m* to carry out the 2D convolution of the image with Gaussian masks. Color-Gist descriptor uses a similar function to filter the image with the different Gabor masks. That function is *imfilter.m*. The input elements are the image and the mask, and the output argument is the filtered image.

We have also developed our own functions in order to implement each global appearance technique; there are functions to create the map, to localize a test image in that map, and to compute the phase lag between images to estimate the orientation of the test image. Moreover, once we estimate the descriptor and the Euclidean distance between the test and the map images, we can represent graphically the results using *plot.m*.

Regarding to the graphical interface, there are two main instructions: *set* and *get*. These commands allow us to assign and obtain the values of the variables and elements of the interface, to store the parameters introduced by the users, and to show the different results in the interface.

## 6 CONCLUSIONS

This work presents a software tool we have designed to be used in a robotics and computer vision subject in a PhD degree. In this subject, the students put in practise some techniques for map building and localization of mobile robots using the appearance-based information extracted from the images captured by the camera the robot carries on it.

Thanks this tool, the students can fully understand the appearance-based approach in robotics mapping and can experience the localization process with real data.

- The students can test how the different compression methods work, which parameters have to be configured and how they influence the results of localization.
- The students are guided during all the process and the results are showed graphically.
- It is not necessary that students know in deep the mathematical developments to carry out all the computations. The student can centre his/her in the configuration and optimization of the algorithms. This way, we centre the objectives of the practical sessions in the robot localization field as students do not get lost with complex mathematical developments.
- As all the databases included have been captured in real environments under realistic lighting conditions, the tool allows the students to face a real problem and experience the troubles that would outcome in a real application.

This way, we provide students with a tool that allows them to freely test and improve the algorithms they learn in the classroom, with no timetable neither equipment restrictions.

## ACKNOWLEDGEMENTS

This work has been supported by the Spanish government through the project DPI2010-15308. ”Exploración Integrada de Entornos Mediante Robots Cooperativos para la Creación de Mapas 3D Visuales y Topológicos que Puedan ser Usados en Navegación con 6 Grados de Libertad”.

## REFERENCES

[1] Amorós, F., Paya, L., Reinoso O., Fernández, L., Marin, J. Mª. (2010). Visual Map Building and Localization with an Appearance-based Approach. 7th Internacional Conference on Informatics, in Control, Automation and Robotics, ICINCO 2010.

[2] Siegwart, R., & Nourbakhsh, I.R. (2004). Introduction to autonomous mobile robots. The MIT Press ISBN: 0-262-19502-X

[3] Krose, B., Bunschoten, R., Hagen, S., Terwijn, B., and Vlassis, N. (2004). Household robots look and learn: environment modeling and localization from an omnidirectional vision system. Robotics Automation Magazine, IEEE, 11(4):45 – 52.

[4] Menegatti, E., Maeda, T., and Ishiguro, H. (2004). Image based memory for robot navigation using properties of omnidirectional images. In Robotics and Autonomous Systems. Vol. 47, No. 4, pp. 251-276.

[5] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, San Diego, USA.

[6] Oliva, A. and Torralba, A. (2001). Modeling the shape of the scene: a holistic representation of the spatial envelope. In International Journal of Computer Vision, Vol. 42(3): 145-175.

[7] Friedman, A. (1979). Framing pictures: The role of knowledge in automatized encoding and memory for gist. In Journal of Experimental Psychology: General, 108:316-355.

[8] Torralba, A. (2003). Contextual priming for object detection. In International Journal of Computer Vision, Vol. 53(2), 169-191.

[9] Gabor, D. (1946). Theory of communication. In Electrical Engineers. Part III: Radio and Communication Engineering, Journal of the Institution of , vol.93, no.26, pp.429-457.

[10] http://www.mathworks.com/products/matlab/