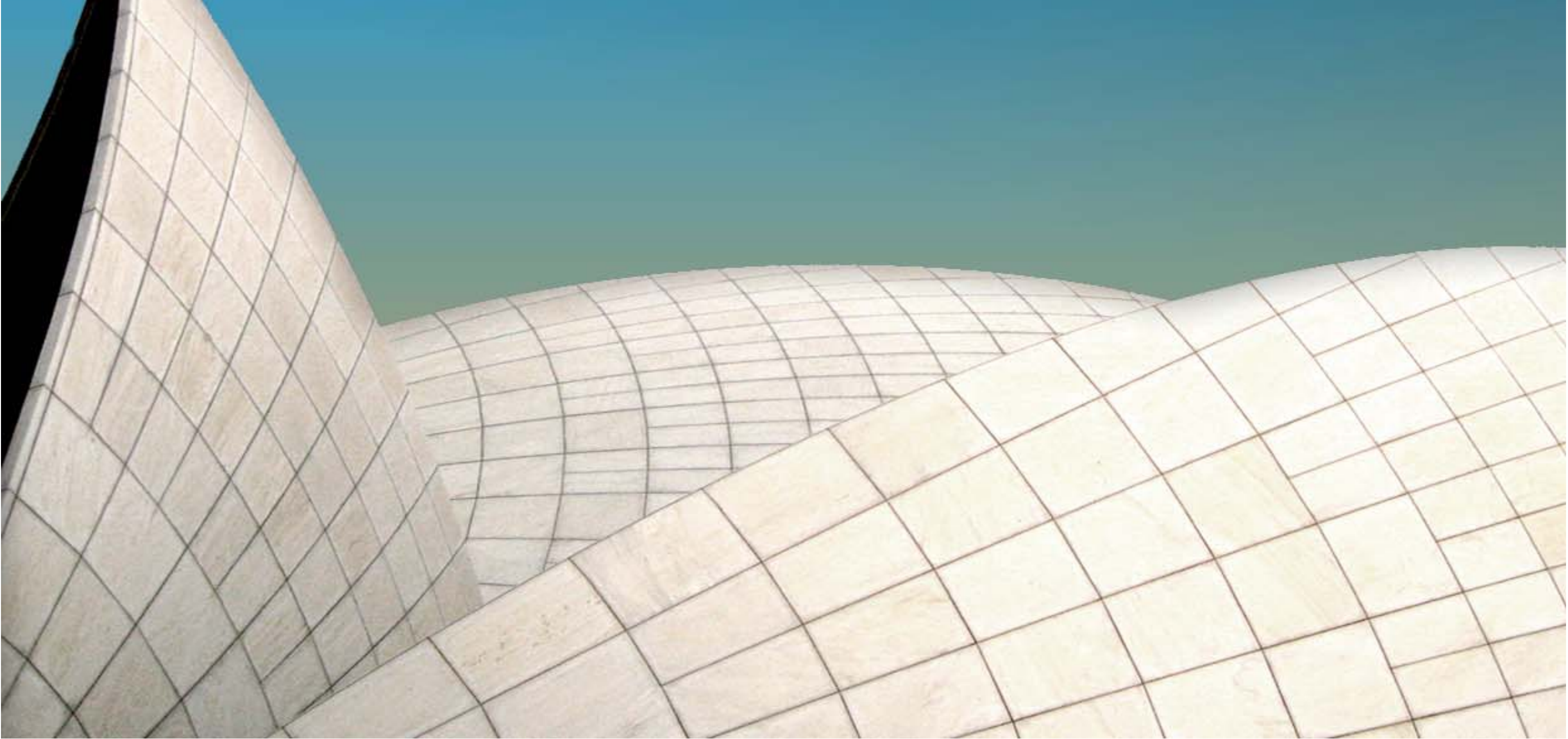
iCERi 2009

International Conference of Education, Research and Innovation

CONFERENCE PROCEEDINGS

Madrid (Spain)
16th-18th of November, 2009
www.iceri2009.org

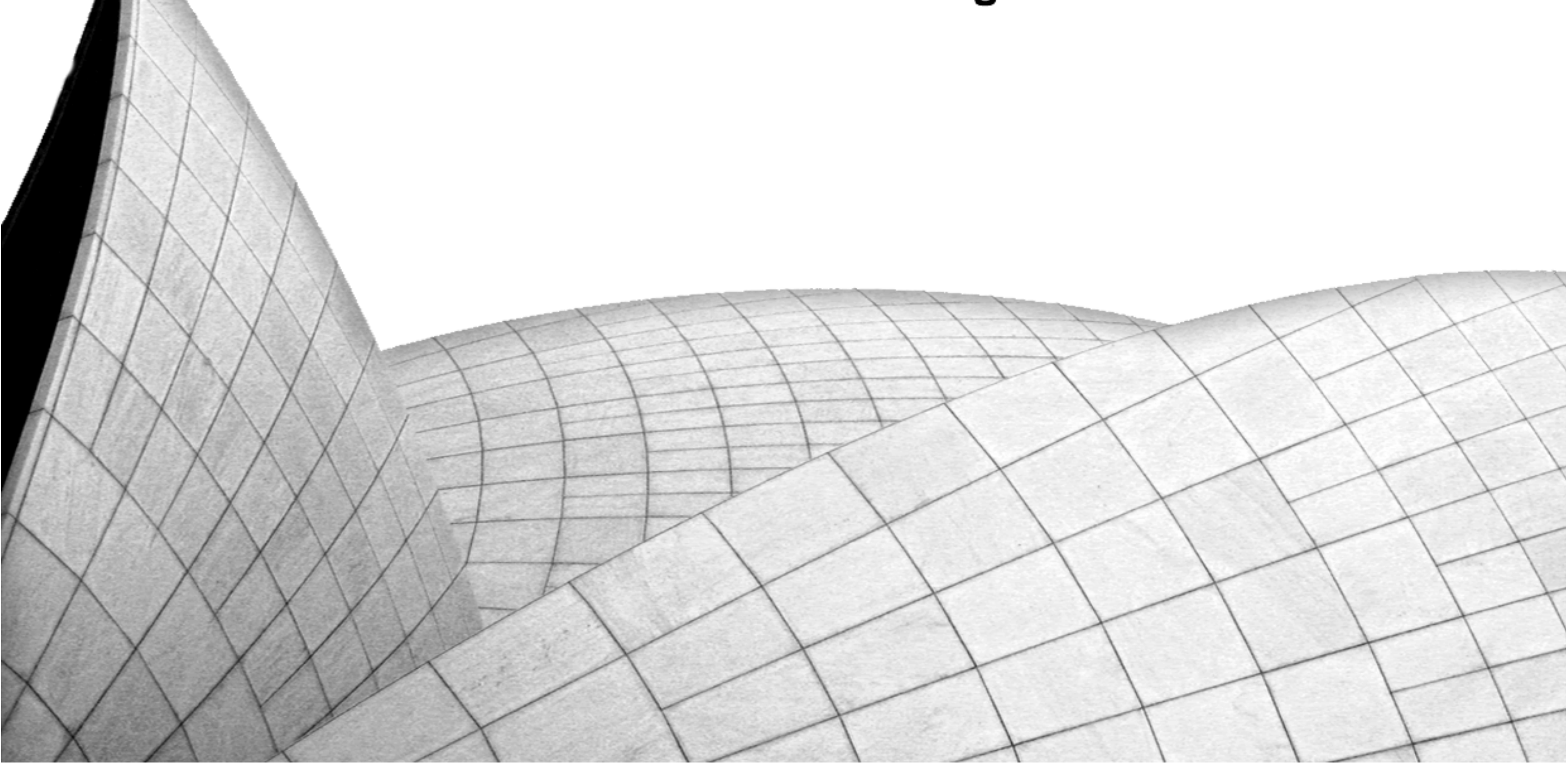


iCERi 2009

International Conference of Education, Research and Innovation

CONFERENCE PROCEEDINGS

Madrid (Spain)
16th-18th of November, 2009
www.iceri2009.org



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

ICERI2009 Proceedings CD
Edited by
L. Gómez Chova, D. Martí Belenguer, I. Candel Torres
International Association of Technology, Education and Development
IATED, Valencia, Spain

ISBN: 978-84-613-2955-7
Depósito Legal: V-3905-2009

Book cover designed by
J.L. Bernat Tomás

Session: Poster Session1. E-learning and Virtual Innovations

A FRAMEWORK FOR DEVELOPING E-CONTENT FOR ONLINE LEARNING
R.A. Yahaya, M.K. Abas

AN APPROACH FOR MULTIMODAL MOBILE LEARNING USING TEXT-TO-SPEECH TECHNOLOGY
O. Zbozhna

APPLICATION OF PROTOCOLS IN DENTISTRY WITH THE AIM OF A SOFTWARE SIMULATION BASED LEARNING
J. Oteo Calatayud, A. Oteo Zaccagnini, I. Charlen Diez, M.D. Oteo Calatayud, C. Oteo Calatayud

APPLYING B-LEARNING METHODOLOGY IN THE "BASIC PSYCHOLOGICAL PROCESSES" SUBJECT ON THE UNIVERSITY OF ALICANTE EDUCATIONAL PSYCHOLOGY DEGREE COURSE
R. Ferrer-Cascales, A. Reig-Ferrer, J. Herranz-Bellido, N. Albaladejo-Blazquez, M.D. Fernandez-Pascual, E. Vallejo-Muñoz

AUDIOBOO: A NEW MEDIUM FOR UBIQUITOUS MICROTEACHING
S. Shilston, J. Lund

BLENDED-LEARNING APPROACH IN TEACHER EDUCATION: THE CASE STUDY OF SCHOLAR PEDAGOGY
P. Queirós, P. Silva

CLINICAL CONNECTED CARE PROJECT – CLINICAL COMMUNICATION SYSTEM FOR EYE CARE
D. Ren, L. Messner

COMPUTER NETWORKING TEACHING EXPERIENCES USING COTS ROUTERS AND VIRTUAL ENVIRONMENTS: THE UC3M LABORATORY
C. Bernardos, A. García Martínez, C. Durán

DESIGNING AND BUILDING A SUSTAINABLE VIRTUAL LEARNING COMMUNITY
E. Robak, E. Moore

DEVELOPING THE FIVE MINDS OF THE FUTURE ON IONIA ISLAND IN SECOND LIFE
A. Lim, S. Wong, V. Lee, S.M. Amin, F. Mohd Noor, A. Tan

E-LEARNING EXPERIENCES ON THE AGRO-FOOD TRAINING AND TECHNOLOGY TRANSFER IN ANDALUSIA. SPAIN
P. Galindo, G. Baena, A. Pedrera, R. Guijarro, F. Ortiz, L. Pérez

E-SUBMISSION OF ASSIGNMENTS: A CASE STUDY
M.K. Abas, R.A. Yahaya, B. Abdul Rahman

EXPERIENCES WITH A BLENDED LEARNING SCHEME BASED ON THE VIRTUAL TECHNOLOGY LAB
A. Picard, K.P. Kaemper, K. Klaeger, C. Kremb, A. Schuetze

GRID TECHNOLOGIES IN CHERNOGOLOVKA FOR SCIENCE AND EDUCATION
M.V. Grigoryeva

HARNESSING THE POWER OF EMERGING CLOUD COMPUTING TO DEVELOP AND FOSTER A COLLABORATIVE SCHOOL COMMUNITY, NGEE ANN 2.0, IN THE NEW DIGITAL AGE
A. Tan, S. Wong, T. Lai, S. Ong, S. Limarsono

INDIVIDUALLY TAILORED-E- LEARNING COMBINING REMEDIAL AND SYNCHRONOUS E-LEARNING FOR LEARNING DISABLED STUDENTS
D. Barat, H. Avni Schon

INTERDISCIPLINARY LEARNING SESSIONS IN UNDERGRADUATED NURSING STUDENTS
M.J. Garcia Cabanillas, P. Bas Sarmiento, J.L. Bianchi Llave, M.A. Martelo Baro, M. Fernandez Gutierrez, C. Castro Yuste, O. Paloma Castro, J.M. Romero Sanchez, L.J. Moreno Corral

INTERNET-BASED CREATIVE WRITING LEARNING: LITERACY AND SOCIAL FORMATION FOR PUBLIC SECONDARY SCHOOL STUDENTS
B. Mota, S. Rodrigues

MAIN PATH TO LEARNING USING INTERACTIVE ADJUSTABLE SOFTWARE
E.R. Rodrigues, J.M. Moreno

MODERN KNOWLEDGE TRANSFER AT UNIVERSITIES
J. Stofkova, K. Stofkova

ONLINE ASSESSMENT AS A TEACHING STRATEGY IN CHEMISTRY AND PHYSICAL CHEMISTRY
G. Montalvo García, M. T. Rodríguez Laguna, R. Palmeiro Uriach, J. Gallego Castro, D. Marin Noarbe

ONLINE FORUM DISCUSSIONS IN TEACHER EDUCATION: HOW CAN THEY HELP TO MONITORING LEARNING OUTCOMES?
P. Queirós, P. Silva

PRACTICAL TASKS TO ELIMINATE JUNIOR HIGH SCHOOL STUDENTS' DILEMMA ON THEIR MOST FREQUENT ERRORS
J. Dorri, R. Vahdani sanavi, P. Yazdani

PRECLINICAL LEARNING OF INDIRECT INLAYS TECHNIQUES ON POSTERIOR TEETH WITH COMPOSITE RESINS FOR POSTGRADUATE DENTAL EDUCATION
J. Oteo Calatayud, A. Oteo Zaccagnini, M.J. Calvo Box, C. Oteo Calatayud

SECOND LIFE – A TECHNOLOGICAL SIMULATION FOR BRINGING GLOBAL PERSPECTIVES OF SOCIAL JUSTICE AND HUMAN RIGHTS TO EVERY CLASSROOM
M. Maisano, E. Rothstein

SOFTWARE FOR THE LEARNING OF APPEARANCE-BASED METHODS IN COMPUTER VISION AND ROBOTICS
L. Paya, L. Fernandez, O. Reinoso, L.M. Jimenez, A. Gil, D. Ubeda

TEACHING POETRY: AN INVESTIGATION OF STAKE-HOLDERS PERSPECTIVE IN ACADEMIC SETTINGS
R. Vahdani sanavi, L. Samady Rendy, J. Dorri

TEACHING TOOLS IN BIOLOGY OR THE MIXTURE BETWEEN TRADITION AND INNOVATION
R. Martín Orti, P. Marín García, A. García Moreno, J. González Soriano

THE PERCEPTION OF STUDENTS WITH VISUAL IMPAIRMENT ABOUT IMPACT OF E-LEARNING ON THEIR EDUCATION
U. Muhammad, R. Bashir

ULTRA HIGH RESOLUTION VIDEO-WALL: DESIGN, IMPLEMENTATION AND FEATURES
A.Y. Menshutin

UNDERGRADUATED NURSING STUDENTS' PERCEPTION ABOUT CLINICAL SESSIONS-BASED LEARNING
C. Castro Yuste, J.M. Romero Sanchez, J.C. Paramio Cuevas, O. Paloma Castro, M.J. Rodriguez Cornejo, C. Carnicer Fuentes, M.J. Garcia Cabanillas, C. O'Ferrall Gonzalez, C. Gavira Fernandez, L.J. Moreno Corral

USE OF DIGITAL LEARNING OBJECTS (DLOS) FOR SUPPORTING ATTENDANCE CLASSES
R. Balart Gimeno, D. Garcia Sanoguera, L. Sánchez Nácher, O. Fenollar Gimeno

USE OF MOLECULAR DIFFUSION ASSISTANT (MDA) SOFTWARE FOR TEACHING PURPOSES
D. Garcia Sanoguera, R. Balart Gimeno, L. Sánchez Nácher, O. Fenollar Gimeno

USE OF VIRSSPA VIRTUAL REALITY TOOL FOR PLANNING DIEP FLAPS IN BREAST RECONSTRUCTION
P. Gacto Sánchez, D. Sicilia Castro, T. Gomez Cía, A. Lagares, M.T. Collell, C. Suarez-Mejías, C. Parra, P. Infante Cossío, J.M. De La Higuera

USING A REMOTE LABORATORY FOR TEACHING LABORATORY PRACTICES IN ENGINEERING COURSES
O. Reinoso, L.M. Jimenez, M. Julia, L. Paya, J.M. Marin, M. Ballesta

SOFTWARE FOR THE LEARNING OF APPEARANCE-BASED METHODS IN COMPUTER VISION AND ROBOTICS

L. Payá, L. Fernández, O. Reinoso, L.M. Jiménez, A. Gil, D. Úbeda

Departamento de Ingeniería de Sistemas Industriales.
Miguel Hernández University. Avda. de la Universidad s/n.
03202, Elche (Alicante), Spain
lpaya@umh.es, l.fernandez@umh.es

Abstract

When a robot has to carry out a task in an environment, a map is usually needed so that the robot can estimate its position and orientation and navigate to the target points. This map can be build using the images taken by a vision system. In this kind of applications, the appearance-based approach has attracted the interest of researchers recently due to its simplicity and robustness. In these methods, the scenes are stored without any feature extraction, and recognition is achieved based on the matching of the whole images. Usually, omnidirectional vision systems are used due to their low cost and the richness of the information they provide.

In this work we present a tool we have developed to be used in computer vision and robotics subjects. This tool provides the students a graphical interface to understand all the concepts involved in appearance-based methods in robotics, working with real data, in a very intuitive way. This software has demonstrated to be very useful for the students to fully understand the appearance-based methods and other basic computer vision concepts.

Keywords - Computer vision, mobile robots, educational software, higher education.

1 INTRODUCTION. APPEARANCE-BASED METHODS

When a mobile robot has to carry out a task in a large environment, it has to take decisions about its localization and about the trajectory to follow to move from its current position to the target point. To solve this problem a map or an internal representation of the environment is needed. This map should contain enough information to allow the robot to compute its current position and the necessary control action to lead it to its destination following, maybe, a chosen trajectory. This map can be build using some images captured by the robot along the environment. One of the major trends in robotics is the study of how to make this representation of the environment from this visual information. These concepts are approached in mobile robots subjects, and are treated deeper in doctorate studies.

Nowadays, there are two main approaches to carry out this map construction. The first family of methods, namely SLAM (Simultaneous Localization And Mapping), tries to build a global map of the environment while simultaneously determining the location of the robot. Usually, these approaches rely on the extraction of several landmarks or characteristic points of the environment either natural or artificial, as [1] does. These models are conceptually quite complex and require high-level mathematical knowledge to be developed.

The second family of methods tries to solve the problem without necessity of creating complex models of the environment. They are based on the comparison of the general visual information of the images, without necessity of extracting any feature. It is just needed a teaching step, where the environment is learned, and a navigation step, where the robot is able to localize itself and go to the target point just comparing its current sensory information with the data stored in the database. These appearance-based approaches are especially useful for complicated scenes in unstructured environments where appropriate models for recognition are difficult to create. The main disadvantage of the appearance-based techniques is that they require huge amounts of memory to store the necessary information of the environment and they suppose high computational cost to make the necessary comparisons during the autonomous navigation so, the key points of these approaches reside on the type and quantity of information to store and in how to make the comparison between the current view and the stored information to reduce the computation time. As an example, [2] and [3] address a method consisting on the direct comparison of low-resolution images and [4] adds other kinds of appearance information.

ISBN:978-84-613-2955-7

In a subject in our doctorate studies, students learn how these methods work theoretically. However, they must be provided with a practical environment where they can test the algorithms they develop. With this aim, we have developed a software tool that allows the practice of all the concepts involved in appearance-based methods through a simple graphical interface and in a very intuitive way. The remainder of the paper is structured as follows: section 2 presents the fundamentals of the appearance-based methods in robotics. In section 3, the experimental that students must carry out is outlined. Section 4 describes the tool we have created to facilitate the experiments. At last, in section 5, we give some implementation details and after this we present the conclusions of the work.

2 APPEARANCE-BASED METHODS IN ROBOTICS

In this paper, we use the appearance-based methods with the objective that a robot builds a map of an environment, using the visual information from the images the robot captures from the environment. Once the map is constructed, the robot must be able to compute its position and orientation within the map.

2.1 Steps in appearance-based methods

The approach consists of two phases:

- Map building. The robot goes through the environment to map and takes some images from several points of view. Images are high dimensional data so, a compression phase is usually needed to extract the most relevant information from each image. Several algorithms can be used with this aim, such as PCA (Principal Components Analysis) or DFT (Discrete Fourier Transform). After this phase, the map, consisting of a data vector for every location, is built. Also, at this phase, a filtering of the data may be carried out to avoid dependency on the illumination conditions and changes in the environment.
- Localization. When the robot has to carry out a task in the environment, it has to compute its location in the map. To do it, the robot has to acquire a new image, compress it and compare it with the data stored in the map. As a result, the location and orientation of the robot must be computed. If no compression was carried out in the previous phase, the computational cost of the operations to make the localization would be extremely high (due to the fact that appearance-based methods work with the information of the whole images, without extracting characteristic points).

In the tool we present, several environments have been included. The student can build the database using the images from these databases, through different approaches and with some filtering techniques. After this, the student can test several algorithms to compute the position and orientation of the robot. With this aim, several sets of test images have been included.

2.2 Appearance-based approaches in robotics

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 was 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 two of the main compression techniques: PCA (Principal Component Analysis) [5], [6] and Fourier transform-based techniques [6], [7]. In the next paragraphs, we give a brief description of these methods.

A. PCA-based techniques

When we have a set of *N* images with *M* pixels each, $\vec{x}^j \in \Re^{Mx1}; j = 1 \ldots N$, each image can be transformed in a feature vector (also named 'projection' of the image) $\vec{p}^j \in \Re^{Kx1}; j = 1 \ldots N$, being *K* the PCA features containing the most relevant information of the image, $K \le N$ [8]. The PCA transformation can be computed from SVD of the covariance matrix *C* of the data matrix, *X* that contains all the training images arranged in columns (with the mean subtracted). If *V* is the matrix containing the K principal eigenvectors and P is the reduced data of size K x N, the dimensionality reduction is done by $P = V^T \cdot X$, where the columns of P are the projections of the training images, $\vec{p}^j$.

Fig. 1 shows the structure of the database that contains several panoramic images, taking *K*=*3*. In this case, each image could be reduced to a point in the *3D* space. The three coordinates of each point contain the most relevant information of the training images. If $K = N$, the information in P is the same as in X (the reconstruction error is *0*) but the dimension of the projected vectors is smaller: each projection, $\vec{p}_j$, has just K components. Further information about this method can be found in [6].

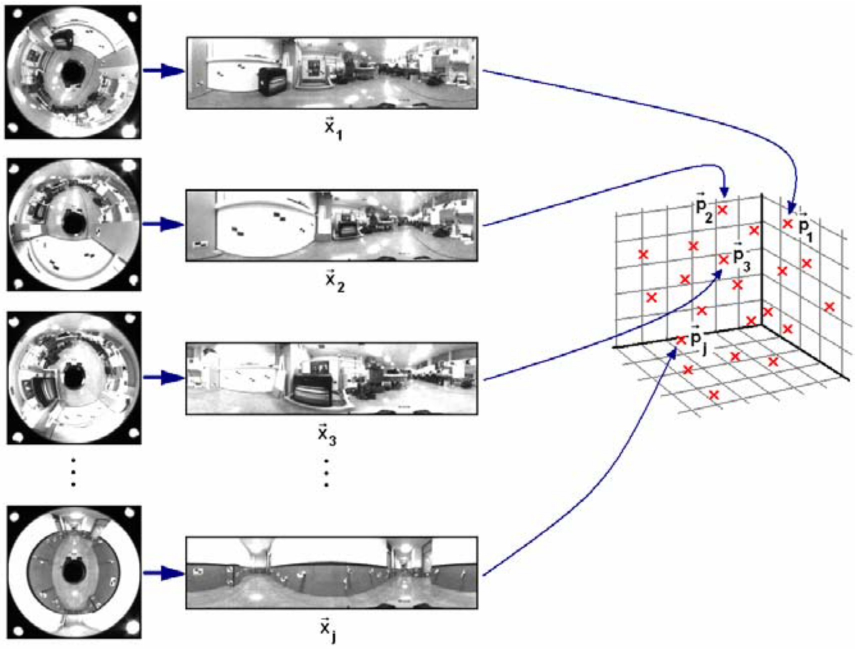


Fig. 1. The panoramic views captured along the environment can be dimensionally reduced by means of PCA.

B. Fourier-based techniques

If we work with panoramic images, we can use another Fourier-based compact representation that takes profit of the shift theorem applied to panoramic images [7]. It consists in expanding each row of the panoramic image $\{a_n\} = \{a_0, a_1, \ldots, a_{N_y-1}\}$ using the Discrete Fourier Transform into the sequence of complex numbers $\{A_n\} = \{A_0, A_1, \ldots, A_{N_y-1}\}$.

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 which are the same but shifted along the horizontal axis. Each row of the first image can be represented with the sequence $\{a_n\}$ and each row of the second image will be the sequence $\{a_{n-q}\}$, being q the amount of shift, that is proportional to the relative rotation between images. The rotational invariance is deducted from the shift theorem, which can be expressed as:

$$\Im\left[\{a_{n-q}\}\right] = A_k e^{-j\cdot\frac{2\pi qk}{N_y}}; \qquad k = 0, \ldots, N_y - 1$$

where $\Im\left[\{a_{n-q}\}\right]$ is the Fourier Transform of the shifted sequence, and A_k are the components of the Fourier Transform of the non-shifted sequence. According to this expression, the amplitude of the Fourier Transform of the shifted image is the same as the original transform and there is only a phase change, proportional to the amount of shift q. 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 [9]. Further information can be found in [6].

C. Image filtering

The appearance of an object in an image can vary strongly depending on the kind and level of illumination of the scene. When we work with the appearance of panoramic images, it is necessary to take into account the fact that appearance is influded both by the position and shape of the objects and the illumination conditions. It is therefore necessary to implement a mechanism that allows us to work independently of the lighting conditions of the environment.

The application of several kinds of filters to the images offers us invariance with respect to the illumination of the scene in object recognition tasks. The different methods can be grouped in two fields. The first one concerns the application of a bank of gradient (first derivative) or Laplacian (second derivative) filters. The second one consists in performing a homomorfic filtering of the image separating the luminance from the reflectance component.

In our application, the student can choose among several kinds of filters and may configure all their parameters to get the best performance.

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. 2a) robot that is equipped with a catadioptric system (fig, 2b), 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. 2c, with a size of *64x256* pixels.

(a) (b) (c)

Fig. 2. (a) Pioneer P3-AT robot, (b) catadioptric system and (c) omnidirectional and panoramic image.

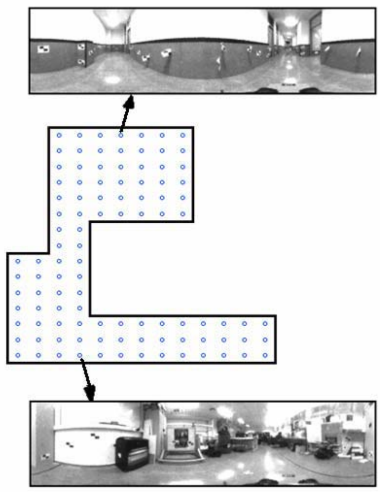


Fig.3. Bird eye’s view of the grid where the images were taken, with two examples of panoramic images.

To carry out the experiments, we have captured a set of omnidirectional images on a pre-defined *40x40* cm grid in an indoor environment (a laboratory). Fig. 3 shows a bird's eye view of the grid used to take the images and two 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:

A. 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 $\vec{x}^i \in R^{Mx1}$ onto the eigenspace, $\vec{p}^i = V^T \cdot \vec{x}^i \in R^{Kx1}$ and compare it 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.

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

B. 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.

4 DESCRIPTION OF THE TOOL

In this section, a detailed description of the operation of the application is given. The general philosophy while designing this tool was the simplicity of use, trying to guide the user during all the process and taking into account the fact that the user may not be an expert in the field. The main goal was to provide the user with a set of methods to test the validity of appearance-based approaches and the necessary tools to put them into practice and to make a critical comparison of the performance of them in a real working environment.

4.1 User's manual

The application has been developed using MATLAB. Some details of implementation are outlined in section 5. The application must be launched from this program so the student has to run first MATLAB in his computer. When the user runs the tool, a graphical interface appears where the student can carry out all the necessary operations. Fig. 4 shows this graphical interface during an experiment. In broad outlines, this figure shows all the necessary options to create the map and to perform the localization process (in an interactive way), a bird's eye view of the map, the test image and its corresponding map image and a graphical representation of the localization process.

In the following subsections we describe the steps to be completed during a typical sequence of use of the tool and the options it offers to the student.

A. Creation of the map

The first step consists in building the map or database that contains information from several images taken along the environment. These panoramic images have been taken over a 40x40 cm grid. The total amount of images is 101. First, the student has to choose the type of compression to apply when creating the map. The possibilities are:

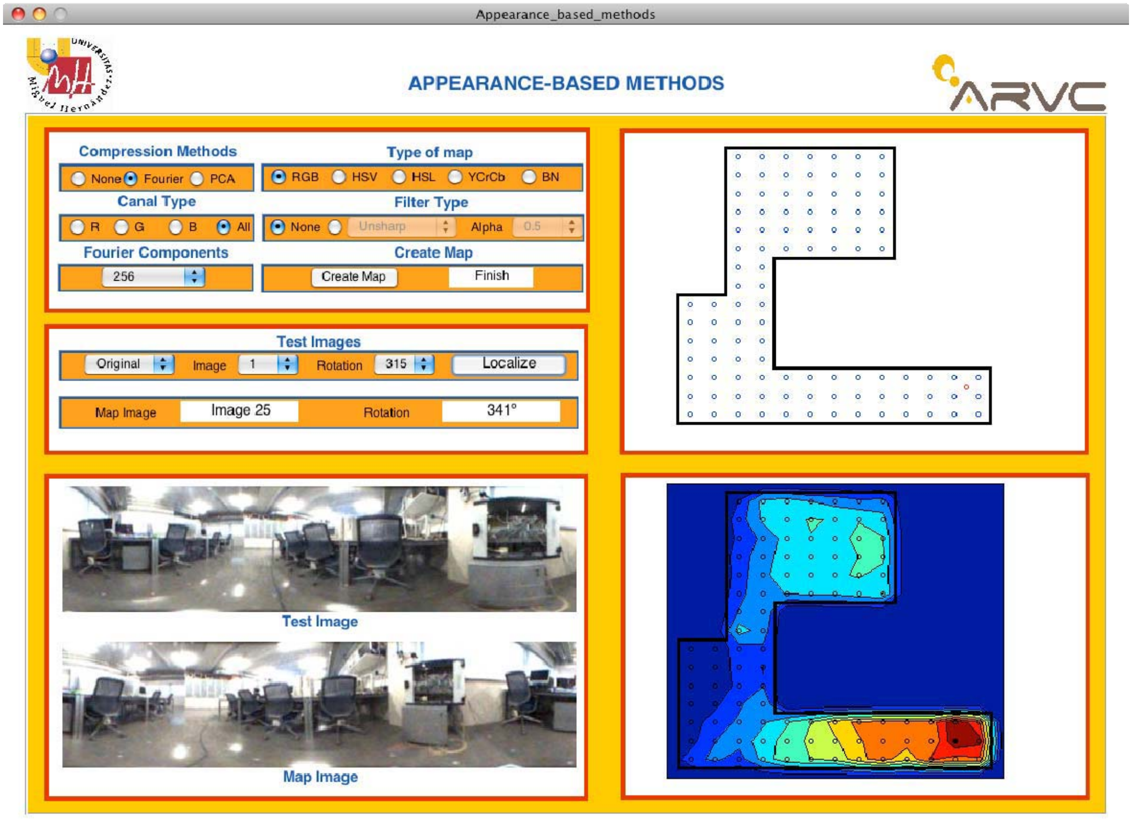


Fig. 4. Appearance of the graphical interface of the application.

- No compression. The images are stored as the have been captured, pixel by pixel, with no kind of compression.
- Fourier compression. The Fourier signature of each image is extracted. In this case, the most relevant information of each image is concentrated in the first components of each row. The student can decide how many columns of each Fourier signature are retained. The database is composed now of the Fourier signature of each image.
- PCA compression. The images are compressed using the Principal Components Analysis approach. Each image is transformed to a data vector with the most relevant information on the image. In this case, the student can choose the number of eigenvectors to work with. The dimension of the projection of the images and so, the amount of the information retained during the compression depends on the number of eigenvectors chosen. The database consists of the projection of each image and the matrix to make the change of basis.

The second option while creating the map consists in selecting the colour codification and channels to work with. The student can choose between the following five choices:

- Original RGB (Red, Green and Blue) colour map images.
- HSV colour map (Hue, Saturation and Value).
- HSL colour model (Hue, Saturation and Ligthness).
- YCrCb colour model (Luminance component and Blue-difference and Red-difference chroma components).
- Grey-level image. This is the model to use when the images have been captured in grey-level.

Apart from selecting the colour model to work with, the student can also decide whether to work with the information of all the channels or just with one of them, to test the robustness of every channel in localization operations.

Finally, in regards to the creation of the map, is possible to use some filters on images. The student can choose from a wide variety of filters, and he is allowed to configure, in all cases, the most important parameters. The available filters are show on fig. 5:

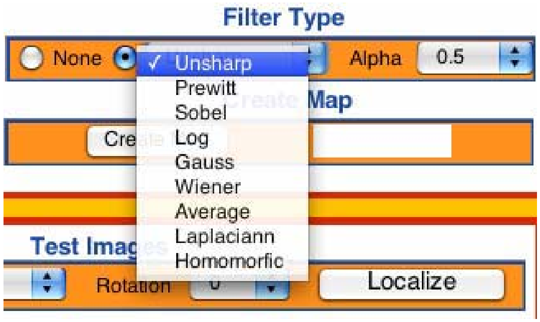


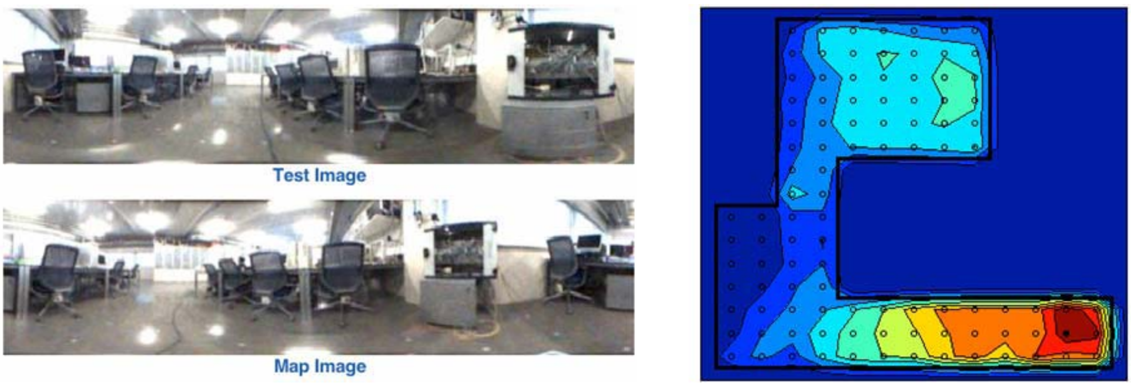
Fig. 5. Filtering techniques the user can choose while creating the map.

Once the map has been created, the amount of time elapsed is shown. This is an interesting data for the student to know the computational complexity with the selected parameters.

B. Localization and orientation of the robot

In the second phase of the experiment, the student has to test if the map created allows the computation of the position and orientation of the robot. With this aim, several test images are available. These test images have been captured by the robot in some half-way points among those of the grid stored in the map. When the student selects one of the test images, the position where it was captured is shown as a red point on the bird eye's view of the environment. Also, the student can select the rotation of this image before perform localization. This is the rotation the robot had when capturing the image and, thanks to this option, the student can test which of the compression methods is robust to changes in the orientation. Once all these parameters have been selected, the student can press the button 'Localize'. Once the process finishes, the interface shows the nearest image in the map, and the orientation computed for the test image with respect to that nearest image. Apart from this, the application also shows (fig. 6):

- On the bottom left of the interface two panoramic images are shown. The first one corresponds to the selected test image and the second is the corresponding image in the map.
- A gradient graphic of distances is shown on the interface. The student can easily see if the results are correct and, in this case, how accurate is the method he has selected and if there exists any confusion region in the environment. Those areas with colours close to red are those with the highest similarity and on the other hand, those close to the blue colours correspond to the less similarity.



(a) (b)

Fig. 6. (a) Test and corresponding map image and (b) similarity map.

The original set of test images has been taken with the same illumination conditions that when the training images were taken. However, other sets of test images are available in the interface to test the efficiency of the method to these changes. The next figure shows some of those test images. There are 17 test images available in each test set.

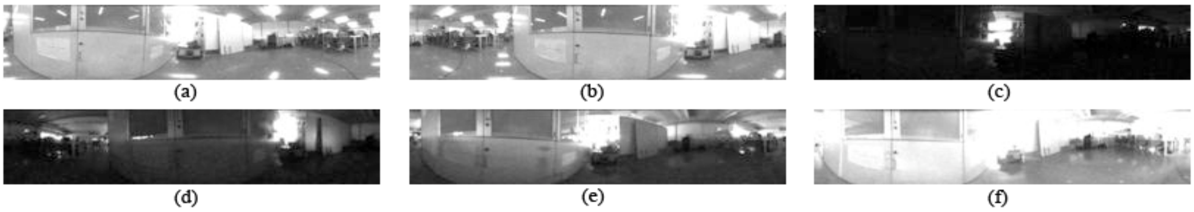


Figure 7: (a) Test 1 (9:00, artificial light), (b) Test 2 (9:00, artificial light, 90 degrees rotation), (c) Test 3 (18:00, no light), (d) Test 4 (11:00, natural light, 90 degrees rotation), (e) Test 5 (13:00, daylight) and (f) Test 6 (16:00, daylight).

5 IMPLEMENTATION DETAILS

To perform the application implementation, we needed a mathematical tool that allowed us to make complex mathematical operations in a short period of time and that allowed us to create a user interface that simplified the use of it. We decided to use MATLAB (MATrix LABoratory), a mathematical software that offers an integrated development environment with its own programming language (M-language).

5.1 MATLAB

MATLAB [10] was created in 1970 by Cleve Moler. It is a high-level language and interactive environment that allows us to perform computationally intensive tasks. MATLAB can be used in a wide range of applications such as signal and image processing, communications, control systems design, modelling, financial analysis and computational biology. It includes vector and matrix operations that are essential to solve scientific and engineering problems.

One of the most important features of MATLAB is the large number of mathematical functions it offers, such as those to solve linear algebra problems optimally, which enable us to perform Fourier analysis or image filtering. It also incorporates a large number of graphic functions for visualizing data, both two or three-dimensional.

5.2 GUI

MATLAB allows the design of graphical user interfaces through the interactive tool GUIDE (Graphical User Interface Developement Environment). This tool allows us to include selection lists, push buttons, radio buttons, dropdowns and sliders, as well as MATLAB plots and ActiveX controls. Also, GUIs can be created programmatically using MATLAB functions.

When we create a graphical user interface using GUIDE in MATLAB, two files are created, *application.m* and *application.fig*. The first one refers to the program code, and it includes the functions and commands used by the application. The second one consists of the graphics and allows us to design the appearance of the user interface. When we add a push button in the graphics window, we must complete the code we need to be run in the share of *application.m*. Similarly we must act for each of the objects we have built into the interface. We can also request the status of the properties of objects (colour, text, etc) at each instant or conversely change these properties at any time. This way MATLAB offers a high degree of interactivity between the user and the GUI.

5.3 Main operations

In this section, we highlight the most interesting MATLAB mathematical functions we have used while developing the code for the application. One is the *fft.m* function that allows us to obtain the FFT (Fast Fourier Transform) of a sequence of numbers. Another important function within the library of MATLAB functions corresponds to *imread.m*, used to transform each image in the database in a *MxNx3* matrix. Thank to this feature we can work with the images as if they were matrices.

With regards to image processing, some functions have been used to modify the colour map images, such as *rgb2gray.m* to transform the image to black and white and *rgb2hsv.m* to transform the colour map image from RGB to HSV. We have also incorporated an external function to transform the colour map to HSL and YCbCr (*colorspace.m*).

To filter the images we use three functions: *fespecial.m* function creates the transfer function, and it is used by *imfilter.m* function, which returns the filtered image. The last of the filtering functions, *homofilter.m*, has been created by us as it has not been found in the library of MATLAB functions. It returns us a filtered image using a Homomorphic filter.

In addition we have incorporated a function within the *application.m* that allows us to obtain the rotation between two images transformed to frequency domain by FFT. The function created is *phase_differece.m* and returns us the relative rotation between two images in degrees. Also, we have added a function that allows the user to know the elapsed time when creating the map database. The function is *progressbar.m* and displays a progress bar.

Finally, two functions have been used to make possible the interaction between the graphical interface and the application code. The *get* command obtains the status of any of the properties of an object in the GUI, allowing us to know the data entered by the user in the interface. With the *set* command, the inverse operation can be carried out so, we can modify any property of a GUI object. Thus, with both commands, the interaction with the user is performed.

6 CONCLUSION

This work presents a software tool we have built to be used in a robotics and computer vision subject in the doctorate studies. With this tool, the students can fully understand the appearance-based approach in robotics mapping, with the next features.

- A database with panoramic images (both grey-scale and colour) of an environment is included. With these images, the student can build the map. Also, some test images are included so that the student can compute the location and orientation of the robot within the map when it captured these test images.
- The student can select different channels from the images to build the database (RGB, HSV, HLS, etc.).
- Some methods to compress the information of the images are implemented so that the student can test their performance and the cases each one works better (DFT and PCA).
- The student can decide the amount of information he wants to retain from each image.
- The tool is fully interactive. Once the map building and the localization steps are finished, several graphical representations of the data can be carried out to know the degree of accuracy of the method used.
- Some optional filters, fully configurable, have been included in the tool. Thanks to them, the student can make the map more robust against illumination variation and changes in the position of some objects of the environment.

This tool has demonstrated to be very useful for the students to fully understand the appearance-based methods and other basic computer vision concepts. The students distinguish the different compression methods and the different parameters to be configured so that they work correctly. They learn the different colour representations of an image, the use of omnidirectional vision and the accuracy in map building and localization. Also, they understand the problem of illumination variation and study some strategies to avoid it. Once the practical sessions have been completed, the student is capable to develop more complex algorithms to control the movements of a robot using this approach.

Acknowledgements

This work has been supported by the Spanish ‘Ministerio de Educación y Ciencia’ through the project DPI2007-61197. ‘Sistemas de percepción visual móvil y cooperativo como soporte para la realización de tareas con redes de robots’.

References

[1] Thrun, S. (2001). A probabilistic online mapping algorithm for teams of mobile robots. International Journal of Robotics Research, Vol. 20, No. 5, (2001), 335-363, ISSN 0278-3649

[2] Matsumoto, Y., Inaba, M., & Inoue, H. Visual navigation using view-sequenced route representation. Proc. IEEE Int. Conf. on Robotics and Automation, vol. 1, (1996), 83-88.

[3] Payá, L., Vicente, M.A., Navarro, L., Reinoso, O., Fernández, C., & Gil, A. Continuous navigation of a mobile robot with an appearance-based method. Proc. 2nd Int. Conf. on Informatics in Control, Automation and Robotics. vol. 1, (2005) 443-446.

[4] Zhou, C., Wei, T., & Tan, T. Mobile robot self-localization based on global visual appearance features. *Proc. IEEE Int. Conf. on Robotics and Automation*, vol. 1, (2003) 1271-1276.

[5] Oja, E. Subspace methods of pattern recognition; Research Studies Press and J. Wiley, 1983.

[6] Payá, L., Fernánez, L., Reinoso, O., Gil, A., & Úbeda, D. Appearance-based dense map creation. Proc. 6th Int. Conf. on Informatics in Control, Automation and Robotics. vol. 1, (2009).

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

[8] Kirby, M. Geometric data analysis. An empirical approach to dimensionality reduction and the study of patterns, Wiley Interscience, (2000) ISBN 0-4712-3929-1

[9] Jogan, M., & Leonardis, A. Robust Localization Using Eigenspace of Spinning-Images. In Proc. IEEE Workshop on Omnidirectional Vision, Hilton Head Island, USA, (2000) pp. 37-44, IEEE.

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