**Noname manuscript No.**
(will be inserted by the editor)

# Combining Invariant Features and the ALV Homing Method for Autonomous Robot Navigation Based on Panoramas

**Arnau Ramisa · Alex Goldhoorn · David Aldavert · Ricardo Toledo · Ramon Lopez de Mantaras**

Received: date / Accepted: date

**Abstract** Biologically inspired homing methods, such as the Average Landmark Vector, are an interesting solution for local navigation due to its simplicity. However, usually they require a modification of the environment by placing artificial landmarks in order to work reliably. In this paper we combine the Average Landmark Vector with invariant feature points automatically detected in panoramic images to overcome this limitation. The proposed approach has been evaluated first in simulation and, as promising results are found, also in two data sets of panoramas from real world environments.

**Keywords** visual homing · biologically inspired methods · local features · robot navigation

**PACS** 68 Computer Science

**Mathematics Subject Classification (2000)** 68T40 · 68T45

## 1 Introduction

The interest in visual navigation systems based on local methods is increasing in the field of mobile robotics: the path between two different location is specified as a succession of intermediate targets to be reached. Hence, the complex navigation problem

This work was partially supported by the FI grant from the Generalitat de Catalunya, the European Social Fund, the MID-CBR project grant TIN2006-15140- C03-01 and FEDER funds, the grant 2005-SGR-00093, the MIPRCV Consolider Imagennio 2010 and the Marco Polo fund from the University of Groningen.

A. Ramisa · A. Goldhoorn · R. L. de Mantaras
IIIA, Campus UAB, 08193, Bellaterra, Spain
Tel.: +34 93 580 9570
Fax: +34 93 580 9661
E-mail: aramisa@iiia.csic.es, mantaras@iiia.csic.es

D. Aldavert · R. Toledo
CVC, Campus UAB, 08193, Bellaterra, Spain
Tel.: +34 93 581 1828
Fax: +34 93 581 1670
E-mail: aldavert@cvc.uab.cat, ricard@cvc.uab.cat

is divided into smaller tractable sub-problems that can be readily solved by simple algorithms with low computational requeriments. Besides, in topological environment representations like in Ramisa et al (2009) a global metric representation is not available. Therefore, the most efficient way of traveling between different locations consists in using a local navigation method. This approach is convenient for navigation in large scale environments, since it is not necessary to estimate a global, geometrically correct, metric map of the environment. In such scenario, the navigation between two different places can be defined as a succession of homing steps.

In robot homing research, many methods solve the perception problem by using trivial to detect artificial landmarks (Möller 2000; Lambrinos et al 2000; Busquets et al 2003; Usher et al 2003). Although these methods often give good results on the experiments, its deployment in practice is limited, as they require setting up the environment beforehand.

Another technique commonly used in local homing is named image warping, and consists in computing a distance between the destination panoramic images and a simulation of the current image that would be introduced by a certain movement of the robot (Möller et al 2009; Vardy 2005). This type of methods provide good results in general, but are more demanding in terms of memory and computation, and are very sensitive to projective distortions introduced by camera movement. Furthermore, using all the image instead of a landmark-based representation makes this type of approaches more sensitive to environment changes.

Methods like that of Goedemé et al (2007), Pons et al (2007) or López-Nicolis et al (2010) use local feature matching techniques to estimate the parameters to navigate to the home position. Although very good results are obtained with these techniques, they require storing all the local descriptor for the origin and destination positions, as well as a computationally expensive step to establish the correspondences between the panoramic images. In contrast, in this work we focus on methods that do not require a feature matching stage. In our work the goal is to create a simple homing method that can be used without having to rely on artificial landmarks or an expensive matching stage. For this we propose the combination of the Average Landmark Vector (ALV) homing technique with visual invariant feature detectors, like the ones described by Mikolajczyk et al (2005), in panoramic images. These local detectors are robust to noise in the image, and its local nature minimizes the effect of viewpoint changes. To the best of our knowledge no other work has addressed the combination of the ALV homing method with invariant feature points such as the MSER or the DoG.

The Average Landmark Vector or ALV Lambrinos et al (1998, 2000) is a biologically inspired approach to navigation that assumes that the animal stores an average landmark vector instead of a snapshot. Landmarks can be (simple) features like edges. The direction to the destination is the difference of the ALV at the destination and the ALV at the current location. Advantages of this model are its simplicity, that only the average landmark vector at the destination location needs to be stored, and that no expensive matching between local descriptors needs to be performed.

The main contribution of this paper is an ALV-based homing method that uses as input information visual invariant features instead of artificial landmarks, and can therefore be used directly in unprepared environments.

In order to evaluate the proposed method, we conducted a series of experiments with the ALV homing method combined with invariant visual feature detectors. First, experiments were done in a simulated environment (Goldhoorn et al 2007a,b) and because the results were promising, experiments were also done with a real robot

(Goldhoorn 2008) in several rooms of an office-type environment. We also conducted experiments with our baseline method (ALV with artificial landmarks) in order to quantify the loss in performance introduced by using the selected invariant visual features as natural landmarks. The obtained results show that the proposed method is suitable for homing in the evaluated environment, and that its performance is only slightly inferior to that of artificial landmarks.

This paper is divided in the following sections: first, related work is discussed, after this we present our method along with some background on the techniques we use, which include the panorama acquisition technique, the local features and the Average Landmark Vector homing method; next, the proposed method is explained; then the experiments performed to evaluate the method are presented followed by a discussion of the results and finally the conclusions and future work are described.

## 2 Related Work

There is significant research in robotic navigation using methods based on animal navigation techniques. For instance, Carwright and Collet (1983) studied how the honeybees learned and used landmarks to navigate. From this research they created the *snapshot model*. The idea of this model is to calculate the home vector, which is the vector pointing to the home position. A panoramic image of the target location is created and stored by the animal. Then, when the insect wants to go back to the stored position it uses a matching mechanism to compare the current retinal image to the stored panorama. Another example is the work on robotic navigation of (Lambrinos et al 1998, 2000) where the ALV was initially proposed, that took inspiration from the different navigation techniques of the ant species *Cataglyphis* described by Wehner (1987). These techniques have the advantage of being computationally cheap.

So far, in most works that studied the ALV homing method, artificial landmarks have been used. For example Lambrinos et al (2000) used as landmarks four black vertical cylinders, and in (Möller 2000) experiments were done inside of a white box with several wide black vertical stripes on the walls. Möller et al (2001) did extensive experiments in a desert type outdoor scenario with four black cylinders as landmarks. In this same work an experiment was attempted in an indoor scenario. Natural landmarks where found by vertically averaging a certain area of the image and finding edges (i.e. intensity jumps) in the unidimensional graylevel profile.

Hafner and Möller (2001) investigated if a Multi-Layer Perceptron with backpropagation and a Perceptron with Delta Rule were able to learn a homing strategy both in simulation and in real world experiments. For the real-world experiments panoramic images acquired by the robot camera were reduced to a single line by vertically averaging (similarly to what Möller et al (2001) did), thus the input of the neural networks is a unidimensional image. Both neural networks successfully learned a homing strategy with the same characteristics as ALV.

Usher et al (2003) used a version of ALV augmented with depth information to guide a car-like vehicle in an outdoor experiment. Landmarks were salient color blobs and the depth information was acquired directly from the distance of the landmark to the center of the omnidirectional image (no unwrapping is performed) using a flat-world assumption. The authors performed real-world experiments using red traffic cones (witch hat model) as landmarks.

Vardy (2005) did an extensive study for a variety of biologically plausible visual homing methods in his PhD thesis, both for local and associative methods, in a real office environment. Among the methods evaluated in his work, there is the one proposed in Hafner and Möller (2001), referred to as *Center of Mass ALV*. In the experiments it performed similarly to other local homing methods, although it was found that an extra learning phase was necessary to determine which area of the panoramic image should be used to generate the unidimensional image in certain environments.

Labrosse (2007) proposes an image warping method to compute a homing vector from a pair of raw two-dimensional images, which avoids relying on types of landmarks that may be nonexistent or sub-optimal for a given environment. Nevertheless, the performance of this method depends on the parameters of the warping procedure, whose optimal value is also environment-dependent. In this work, a compromise value for these parameters was obtained as the average between the optimal value for a large number of image pairs separated by a 25 cm displacement. The constant orientation problem is addressed by calculating column shifts between the consecutive panoramas, bounding the accumulated estimation error at 5° in short trajectories. Finally, in order to gain robustness to large occlusions between the current and the destination image, the route is divided in multiple, manageable, sub-problems.

Another homing by image warping method has recently been proposed in (Möller 2009; Möller et al 2009). These methods build on the work of Franz et al (2008) and improve if by using two-dimensional instead of one-dimensional images. They showed experimentally (in their own datasets and also in standard, pre-existing ones) that their 2D method worked better, especially with multiple scale panes, and that the additional computational time needed was small.

Chaudhari (2010) uses the ALV algorithm with single 180° FOV camera, from which he extracts predefined landmarks using color. Since in order to navigate using the ALV all the landmarks have to be detected, a ring of sonar sensors is used to maintain a map of the landmarks. Experiments were done both in simulation and with a Pioneer robot.

## 3 Proposed Method

In this section we describe the different concepts and techniques that we have used: First the panorama acquisition technique, next the local feature detectors, an finally the foundations of the ALV homing technique.

### 3.1 Panorama

The panorama acquisition technique used in this work consists in stitching images taken with a conventional camera rotating around a fixed point of view until the full 360° have been covered. The images should be projected onto a smooth surface such as a cylinder to avoid discontinuities or inhomogeneous sampling. A cylindrical representation offers some advantages. In the first place it can be created relatively easily, and also, in contrast with other plenoptic representations such as a sphere, can be unrolled and stored in an efficient way as a conventional rectangular image (McMillan and Bishop 1995). Using this panorama acquisition technique, it is important to have a fixed optical center to avoid introducing motion parallax. However, small translations can be tolerated when the objects are far enough from the camera. We used the same approach for panorama construction that was used in (Ramisa 2006). First the coordinates have to be transformed from the Cartesian system of the images to a cylindrical coordinate system:

$$\theta = \tan^{-1}\left(\frac{x}{f}\right), \quad v = \frac{y}{\sqrt{x^2 + f^2}} \tag{1}$$

Where ($x$,$y$) is the pixel position in the image, $f$ the focal distance (in pixels), $\theta$ the angular position and $v$ the height on the cylinder. The radius of the cylinder is equal to the focal length of the camera to optimize the aspect ratio of the image (Shum and Szeliski 1997). The next step is to stitch the images, but for this the displacement vectors $\Delta t = (t_x, t_y)$ have to be calculated for each succeeding image pair. In theory $t_x$ can be deduced from the panning angle and $t_y = 0$, however in reality this is not true due to camera twist and not perfect panning.

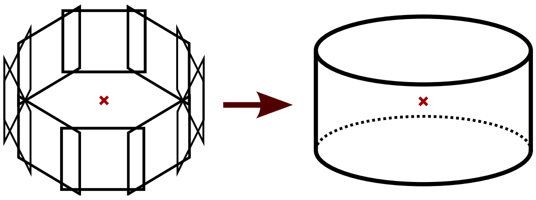


**Fig. 1** The projection from the image sequence to a cylinder.

Local features (see section 3.2) can be used to estimate the translation between two images. The advantage of using local features instead of the more conventional iterative maximization of the normalized correlation (McMillan and Bishop 1995; Szeliski and Shum 1997) is its lower computational complexity (provided that the local features will have to be computed anyway) and higher robustness to several image transformations such as illumination changes, noise and zoom. However, in the case of few texture in the image it is not possible to use the feature-based approach and the iterative method is used. When the translations have been calculated, the images can be stitched to produce the whole panorama.

An example of such a panorama created by stitching is shown in Figure 2. As can be seen there are still small distortions due to not perfect shifting of the images. The difference in intensity is because of the automatic camera gain. To avoid artifacts created in the stitching process, the features from the original images are used. Features from overlapping regions are only added to the constellation once. Another way to acquire panoramas is by using an omnidirectional camera. There are two approaches to do this: by using a fish-eye lens and by using a conventional camera pointed to a hyperbolic mirror above it. These methods have some clear advantages such as the speed of creation and that no images have to be stitched, and therefore no artifacts will be introduced. A disadvantage is the lower resolution of the acquired images.

Finally, another alternative to acquire panoramas is using a camera ring of synchronized cameras, which offers a high speed of acquisition without sacrificing the high resolution. A disadvantage of this method is the high price of the whole system.

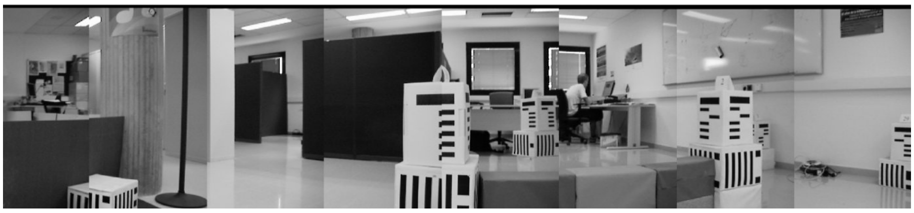


**Fig. 2** Part of a panorama image created by stitching several images together. The image is made in the robot laboratory.

## 3.2 Local Visual Features

Local visual features can be points or regions of an image which correspond to a local extrema function over it. The main interest of these features is that are detectable under several transformations and illumination changes. This robustness makes them very suitable for the purpose of matching and recognition. Moreover, representations made with such local features are robust to partial occlusions and background clutter. Extracting features from an image reduces the dimensionality of the information and adds robustness against noise, aliasing and acquisition conditions.

The feature region detectors Maximally Stable Extremal Regions (MSER) and Differences of Gaussians (DoG) are used in this work to test the homing method because they are fast to compute and yet robust. Here follows a brief description of these detectors.

### *3.2.1 Differences of Gaussians*

The Scale-Invariant Feature Transform (SIFT) algorithm proposed in (Lowe 1999, 2004) is based on a biologically inspired model of complex neurons in the primary visual cortex proposed by Edelman et al (1997). These neurons are activated by a gradient in a particular orientation if it appears within a small range of positions in the retina. Although the SIFT algorithm includes both an interest region detector and a descriptor, we are only interested in the detector part for this work: the extrema of the differences of Gaussians. The standalone version of this detector is known in the computer vision literature as Differences of Gaussians or simply *DoG*. Differences of Gaussians $D$ are produced subtracting every two neighbour levels of the scale-space of the image, separated by a factor $k$:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y), \tag{2}$$

where $G(x, y, \sigma)$ is a Gaussian kernel with standard deviation $\sigma$ and $I(x, y)$ is the input image. Figure 3 shows an efficient approach to construct $D$. To avoid detecting multiple times the same feature at different scales, they are only detected at their characteristic scale (Lindeberg and Gårding 1997).

Local extrema of $D$ are detected by comparing each sample point to its eight neighbours in the current image and the nine neighbours in the above and below scales of the DoG. The point is selected only if it is the maximum or minimum in its neighbourhood. Finally unstable feature points are rejected. These correspond, for example, to feature points localised along an edge or feature points with low contrast.

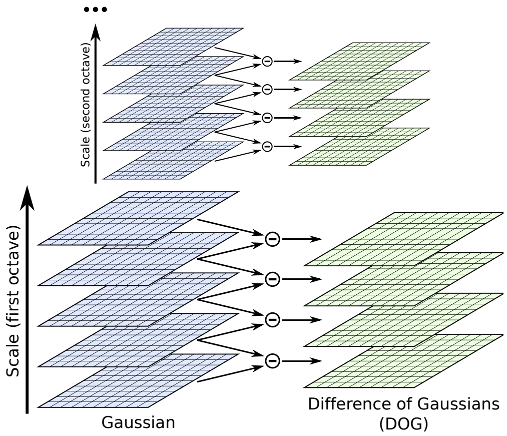


**Fig. 3** At the left the initial image is incrementally convolved with Gaussians. The adjacent image scales are subtracted to produce the DoGs, which are shown at the right. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process repeated.

#### *3.2.2 MSER*

The Maximally Stable Extremal Regions (MSER) proposed by Matas et al (2002) can be defined informally as image regions in which the pixels have an intensity value much higher or lower than neighboring pixels. Although apparently very simple, MSER feature points are very stable to change of viewpoint (they are perspective-invariant) and to affine illumination changes. Furthermore, the algorithm proposed by Matas et al (2002) to compute the MSER feature points has a near linear complexity. The algorithm works as follows: First the pixels are sorted by intensity, then the pixels are placed in the image (in decreasing or increasing order) and the list of connected components and their areas are maintained using an efficient union find algorithm. Each connected component is stored as a function of intensity. By doing intensity thresholds we find the parts of the function where no changes in the area of connected components occur, i.e. they are not merged with others. These parts are the maximally stable extremal regions. Murphy-Chutorian and Trivedi (2006) propose an even more efficient version of the algorithm to compute MSER feature points using a N-Tree Disjoint-Set Forests structure.

The MSER detector was tested by Mikolajczyk et al (2005) and found to be one of the best in their repeatability experiments. A notable advantage of this method over DoG is that the regions found are much more robust and faster to compute. On the negative side, MSER feature points are usually scarce, which makes this type of feature points unsuitable (at least when used without complementary features) for applications such as object localization and recognition (Vinyals et al 2007).

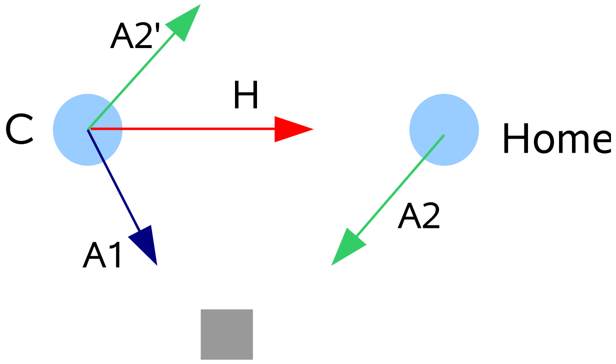


**Fig. 4** The calculation of the home vector. Both ALVs ($A_1$ and $A_2$) point to the average feature position, which is drawn as a gray block. The home vector ($H$) is calculated by subtracting the ALV at the destination location ($A_2$) from the ALV at the current location ($A_1$). This subtraction is shown, by the addition of the reverse vector, $A_2'$, to $A_1$. The robots are aligned in this example.

### 3.3 Average Landmark Vector (ALV)

In this section we describe the biologically inspired homing technique Average Landmark Vector by Lambrinos et al (1998, 2000). The ALV is defined as the average of the landmark (or feature) position vectors:

$$ALV(F, \overrightarrow{x}) = \frac{1}{n}\sum_{i=0}^{n} \overrightarrow{f_i} \tag{3}$$

Where $F = \{\overrightarrow{f_1}, \overrightarrow{f_2}, \ldots, \overrightarrow{f_n}\}$ is the collection of features that define the signature taken at the current position $\overrightarrow{x}$, and $f_i$ are the coordinates of the $i^{\text{th}}$ landmark position vector. In this equation $F$ contains the global feature positions to explain and proof the homing technique. This is the robot centered version, but it is made world centred by subtracting the current position $\overrightarrow{x}$ of the robot in the world to easily proof that the homing technique works:

$$ALV(F, \overrightarrow{x}) = \frac{1}{n}\sum_{i=0}^{i=n} \overrightarrow{f_i} - \overrightarrow{x} \tag{4}$$

The home vector is defined as follows:

$$homing(F, \overrightarrow{x}, \overrightarrow{d}) = ALV(F, \overrightarrow{x}) - ALV(F, \overrightarrow{d}) \tag{5}$$

Where $\overrightarrow{x}$ is the current location of the robot and $\overrightarrow{d}$ the destination. When the $ALV$ functions are substituted by Eqn. 5 then $\overrightarrow{d} - \overrightarrow{x}$ remains, which is exactly the home vector. Figure 4 shows an example of the calculation of the home vector. To simplify the image only the average landmark (the gray square) is shown. In this example it is also assumed that the depth of the landmarks is known. The ALVs are calculated for the current ($C$) and the $Home$ position, these are $A_1$ and $A_2$ respectively. The home vector ($H$) is calculated by subtracting the ALV at the destination position ($A_2$) from the ALV at the current position ($A_1$). This results in the home vector $H$ which points to the destination location.

One important prerequisite of the ALV is that it is necessary to have the panoramic images aligned to an external compass reference before computing the homing direction. The Sahara ant *Cataglyphis*, for example, uses the polarization patterns of the blue sky to obtain the compass direction (Wehner 1994).

ALV homing does not work when the ALV at the current location and at the goal location are the same (after correction for orientation differences), because this results in a zero vector. An exceptional theoretical case in which this could happen is when the ALV point, the current location and the goal location are aligned, in practice however this is very unlikely. To let the robot move anyway in such situations a random vector could be used to move the robot a small distance, and then continue the homing procedure.

As a way to solve the constant orientation prerequisite, in our work all test panoramic images have been acquired with the robot facing a constant direction as is common practice in similar works (Möller et al 2001; Hafner and Möller 2001). In order to apply the ALV method in a navigation experiment a magnetic compass, or another system to acquire the global orientation, is required to align the panoramas.

## 4 Experiments Performed and Results Obtained

### 4.1 Simulation

To evaluate how well the ALV homing method works with our type of visual features, a series of simulation experiments were performed first. Here we report the most important findings of these experiments. A more detailed explanation and discussion of the simulation experiments can be found in (Goldhoorn et al 2007b; Goldhoorn 2008).

The experiments were done in a simulated environment (see Fig. 5) with different distributions of feature points. The environment is a room composed of a flat floor, in which the robot moves, and up to four visible walls made of simulated feature points and projected to a virtual camera located in the robot position, that closely simulates the field of view of the panoramic images acquired with our real robot. A simulated robot run was said to be successful if it found the destination point within the following three limitations:

1. The robot is not allowed to use more than 2000 steps (iterations)
2. The projection of the world should not be empty more than five times in a row (in that case either the previous home vector or a vector with random orientation and length was used)
3. The robot should travel at most a distance ten times the Euclidean distance between the start and destination position.

Although the feature points used are robust to most image variations, there are almost always changes due to noise in the localization or occlusions. Adding Gaussian noise to the positions of the feature points with a standard deviation of 0.001 m or less resulted in a 90% successful runs, and a standard deviation of 0.05 m or more resulted in only 5% or less of successful runs.

Occlusions were simulated by removing randomly chosen feature points before every projection. Removing 50% of the feature points resulted in a mean success rate of 85%. The method was also robust to adding randomly placed feature points, which can be thought of as reappearing previously occluded objects.

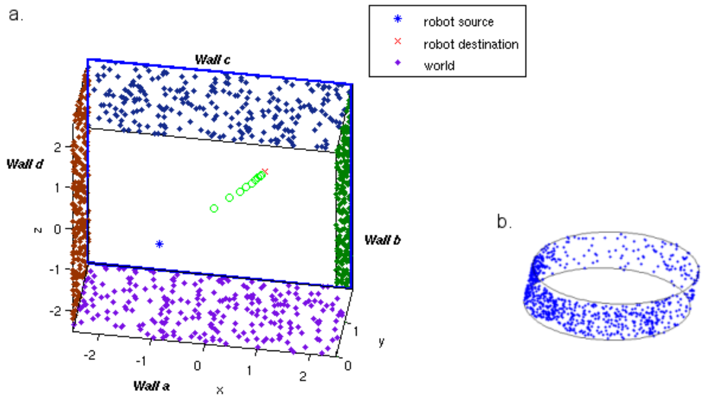


**Fig. 5** a) The simulated environment with uniformly randomly spread feature points. b) Panoramic projection of the world used as input for the robot homing system.

Having more reliable feature points present in the world increases the performance of the robot (higher success rate, less iterations and a smaller difference with the Euclidean distance). For the simulation, the range for the number of feature points is between 500 and 1000 for a success rate of 100%. Although having only 20 feature points in the world still resulted in 50% to 80% successful runs. However it has to be taken into account that these runs were without any noise and without any other disturbances.

Because no depth is used, the ALV method implies an equal distance assumption of the landmarks. Franz et al (1998) also mentions the isotropic feature distribution, which can explain why results in a world with only one wall were worse than in the other configurations.

From these experiments we concluded that visual feature points are suitable for visual homing with the ALV and, consequently, the next step was to try this method on a real robot.

## 4.2 IIIA Panoramas Database

This section shows the experiments conducted with the IIIA panoramas database[1]. First the experimental setup is explained, and then the results are presented and discussed.

### *4.2.1 Experimental Setup*

As it is common in related literature, in these experiments several panorama were acquired at a grid of known points in the rooms. The orientation of the robot was kept constant for each panorama so no alignment step is necessary between them.

[1] The IIIA Panoramas database can be downloaded from `http://www.iiia.csic.es/~aramisa/datasets/iiia_alv.html`

Three types of landmarks/feature points were used: 1) DoG feature points; 2) MSER feature points; and, only in the robot laboratory, 3) artificial landmarks. The experiments were done in three rooms of different sizes: the *robot laboratory*, the *square room* and the *corridor*. A scaled map of the rooms can be seen in Figures 7, 9 and 10.

The locations where the panoramas were created are marked as circles with its identifying number and a line starting at the center of the circle and pointing to the direction of the estimated home vector. The home location is shown as a red circle without line and is also indicated in the figures captions. The biggest objects in the rooms, such as desks, are also shown in the maps to give a rough idea of the environment. Finally, the squares in Figure 7 show the landmarks positions and its ID number.

Like in the simulation, only the direction of a feature is known and not its distance, therefore the home vector will not contain distance information either. The home angle calculated by the homing method is compared to the ground truth home angle which is calculated by geometry.

$$\theta_{\mathrm{diff}}(h_{\mathrm{h}}, h_{\mathrm{c}}) = \min\left(|h_{\mathrm{c}} - h_{\mathrm{h}}|; 360 - |h_{\mathrm{c}} - h_{\mathrm{h}}|\right) \tag{6}$$

All angles are in degrees and counter-clockwise; $h_{\mathrm{c}}$ is the correct homing direction calculated by using the positions (geometry), and $h_{\mathrm{h}}$ is computed by the homing method. To find out how well the method works for each room and each type of feature, all the panorama positions per data set are used. For each data set (the *square room*, the *robot laboratory* and the *corridor*) all the locations where a panorama was created are used to calculate the home vector to each of the other locations. From the error calculated with Eqn. 6 for each possible panorama pairings in one room, the mean, median, standard deviation and a score are calculated. The score is calculated by using the proportion of the maximum error and ranges between 0 and 1 where 1 is best. Namely:

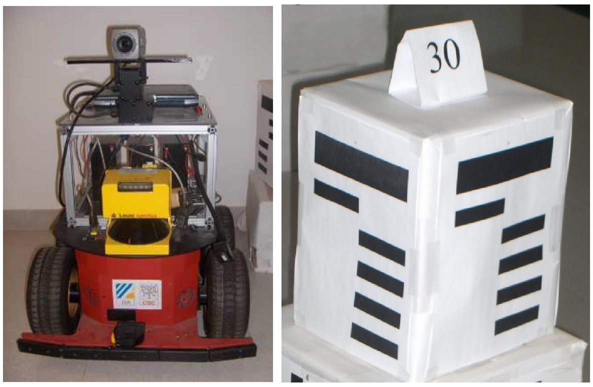
$$s = 1 - \frac{\sum_{i=1}^{n} \sum_{j=1; i \neq j}^{n} \theta_{\mathrm{diff}}(h_{\mathrm{h}}(P_i, P_j), h_{\mathrm{c}}(P_i, P_j))}{180n(n-1)} \tag{7}$$

where $n$ is the number of panoramas in the set and $P$ the set of panoramas. The numerator is the sum of the difference of the home angle estimated by the ALV homing method and by geometry. This error measure is calculated for each panorama pair, which in total are $n(n-1)$ pairs. The sum of errors is divided by that factor to get an average and, to normalize the score between 0 and 1, it is also divided by 180°, the maximum possible error.

### *4.2.2 Robot*

A Pioneer 2AT robot (Figure 6.a) is used with a pan tilt unit (Directed Perception PTU 46-70) mounted on it and on top of this PTU, a Sony DFW-VL500 camera with a resolution of 640 × 480 pixels. The robot is controlled from an on-board laptop (Acer Travelmate C110).

The experiments were done in the three mentioned different areas of the IIIA-CSIC research center. The room in which most experiments were done is the robotics laboratory. The panorama in Fig. 2 shows this room as seen from the robot and in Figure 7, a map can be seen.



(a) (b)

**Fig. 6** (a) The Pioneer 2AT robot as used in the experiments. A pan tilt unit is mounted on the robot with a camera on top. (b) An example of a landmark in the *robotics laboratory*.

### *4.2.3 Landmarks*

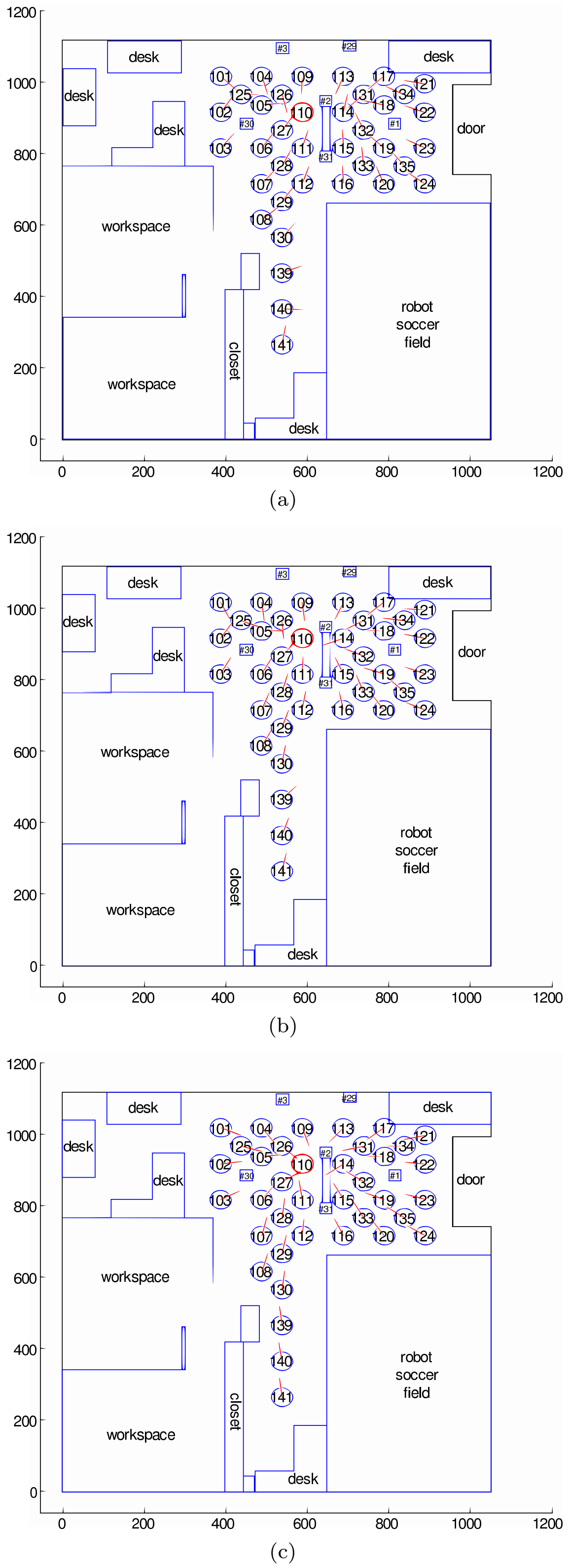
In order to compare our proposed approach to an artificial landmark based one, extra experiments were done using six artificial landmarks in the *robotics laboratory* (see Figure 6.b) available from previous experiments (Busquets 2003; Busquets et al 2003).

The landmarks contain a bar code from which an ID number can be extracted and, since the size of the bars is known, the distance to the landmark can be calculated. In order to make the artificial landmark approach comparable to the feature based one, neither the landmark number (for matching) nor the distance information was used in our experiments.

### *4.2.4 Results*

When calculating the home vector between two points, for example $a$ and $b$, the home vector from $a$ to $b$ will obviously always point in opposite direction of the home vector from $b$ to $a$. This means that these are dependent values and therefore only one of them was used in the analysis. Next we discuss the results for the three different areas.

*Robotics laboratory:* Most panoramas, 38 in total, were acquired in the *robotics laboratory*, a room of 10.5 m × 11.2 m. Only the half of the room is really used for this experiment because the other part is filled with working places and the robot soccer field as can be seen in Figure 7. The behaviour of the proposed method in this environment was satisfactory: Home vectors with an error equal or lower than 10° (direct approach to target) were obtained in 22.6% of the cases for the DoG detector, 32.7% for the MSER detector and 64.3% for landmarks. An error lower or equal to 90° (suitable for a zig-zag approach to destination) was obtained in 89.3% of the cases when the DoG detector was used, 92.6% for the MSER detector and 99.6% when the landmarks were used. Table 1 shows the results for each type of detector used. The homing errors for the three methods are all significantly different ($p < 0.001$) according to the rank sum test, and the $t$-test after bootstrapping ($n = 1000$). From this can be concluded



(a)

(b)

(c)

**Fig. 7** Homing to panorama 110 in the *robotics laboratory* using DoG feature points (a), MSER feature points (b) and the landmarks (c). All measures are in cm.

| | DoG | MSER | Landmarks |
|---|---|---|---|
| Mean error | 35.60° | 27.84° | 14.88° |
| Median error | 22.85° | 16.03° | 10.17° |
| Standard deviation | 36.67° | 35.51° | 14.86° |
| Score (s) | 0.8022 | 0.8454 | 0.9173 |
| Best home | 117 | 117 | 110 |

**Table 1** The homing error using the panoramas from the *robotics laboratory*. The *best home* field shows the number of the panorama (see Figure 7 for the numbers in the *robotics laboratory*), which when chosen as home, resulted in the lowest average error.

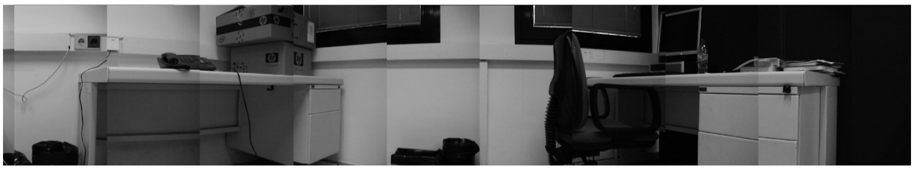


**Fig. 8** Part of the panorama 137 from the *square room*.

| | DoG | MSER |
|---|---|---|
| Mean error | 13.78° | 9.65° |
| Median error | 12.00° | 12.03° |
| Standard deviation | 11.31° | 7.84° |
| Score (s) | 0.9234 | 0.9464 |
| Best home | 138 | 138 |

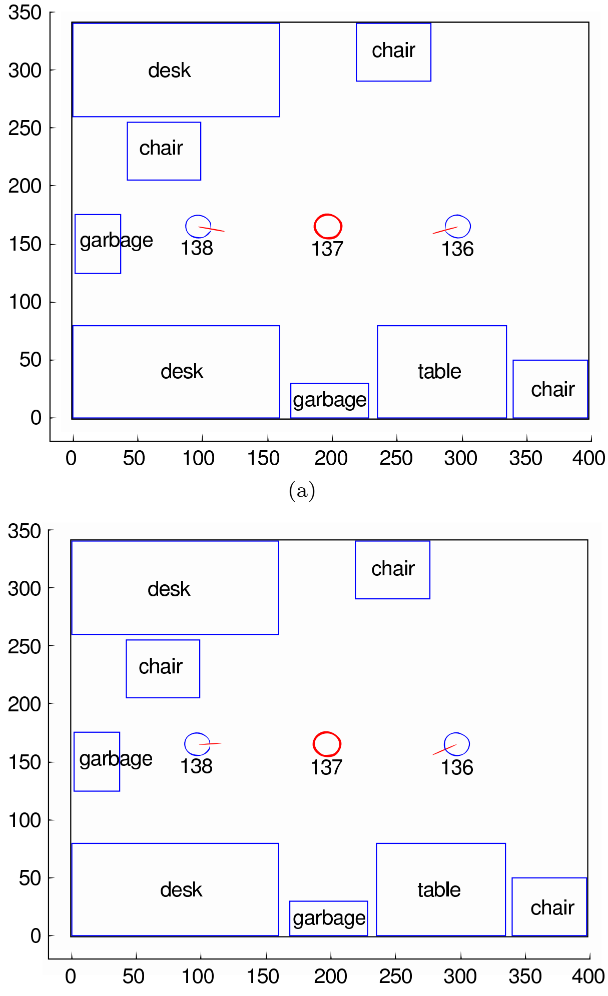
**Table 2** The error of the homing method using the panoramas which were made in the *square room*.

that the homing method worked best with the artificial landmarks, as expected, and worst with the DoG detector.

*Square room:* In Figure 8 a panorama from the square room (actually its size is 4.0 m × 3.4 m) can be seen. Figure 9 shows the map of the room and the home vectors to panorama 137. As in the case of the robotics laboratory, MSER feature points achieved lower error rates than DoG feature points, but this is not significant (confirmed by the rank sum test and the $t$-test) and it must be noted that only three panoramas were created in this room. Table 2 shows the statistics of the homing method using both feature types.

*Corridor:* Although the simulation showed that the ALV homing method works better in square rooms, we wanted to find out what the impact of a very long and very narrow room in a real environment would have on the method. A corridor was chosen for that reason as last experiment environment. The part of the corridor in which the robot moved is 2.2 m wide and about 22.5 m long. In Figure 10 the map of the corridor with the homing vectors to panorama 203 can be seen, and in Figure 11 the panoramas acquired in the corridor.

As expected, the performance in this dataset was much lower than in the previous tests: An error of 90° or less was obtained only in 73.3% of the cases for both feature types, and an error of 10° or less was only obtained in one case (6.7%). Table 3 shows the average error of this data set; the differences between the results with DoG and



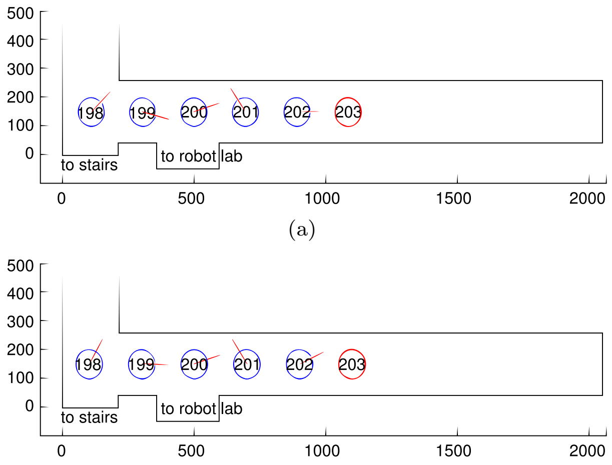
(a)

(b)

**Fig. 9** Homing to panorama 137 in the *square room* (a) using DoG points and (b) MSER points. All measures are in cm.

| | DoG | MSER |
|---|---|---|
| Mean error | 56.26° | 52.67° |
| Median error | 44.58° | 35.71° |
| Standard deviation | 43.64° | 44.90° |
| Score (s) | 0.6874 | 0.7074 |
| Best home | 203 | 200 |

**Table 3** The average error of the homing method in the *corridor* for the different feature types.



(a)

(b)

**Fig. 10** Homing to panorama 203 in the *corridor* using (a) DoG feature points and (b) MSER feature points. All measures are in cm.

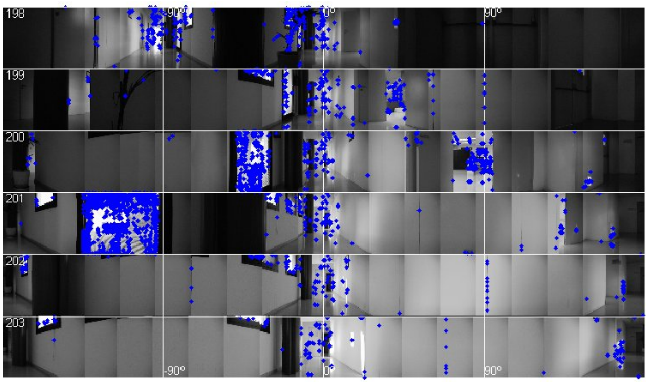


**Fig. 11** All the panoramas made in the corridor. The dots are MSER feature points.

MSER are not statistically significant. The panoramas acquired in the *corridor* (Figure 11) show that there are several *disturbing* factors on which numerous feature points were found but that were only visible from one or a few of the panoramas. For example, panorama 198 is the only panorama taken at a corridor intersection, and therefore the feature detectors find a large number of feature points than are not visible in the other panoramas. In panoramas 200 and 201 a door with blinds is visible, but since it is very close to the robot, its size rapidly changes, and with it the amount of feature points found. Finally, in panoramas 199 and 200 the robotics laboratory is visible through an open door which again has many feature points. The deviations in the home direction in Figures 10.a and 10.b show how the extra features *pull* the home vector in their direction. In Table 4 (see Annex), can be seen that the best *corridor* of the IIIA data sets is at rank 25, which is below the best of the data sets *robot lab* and *square room*.

As is clear the results at the different rooms, the ALV homing method works better in both the *square room* and the *robotics laboratory* than in the *corridor*. This difference is explained by the previously found conclusion, in the simulated experiment (Section 4.1), that the method works better in approximately square rooms that have an isotropic landmark distribution. This problem is well known in the homing community and an active area of research (Möller et al 2007).

*Upper and lower part:* Limiting the view of the robot to only the lower half of the panorama displays only the objects which are closer to the robot and therefore decrease the size of the visible world, making the perceived room more square.

In the *robotics laboratory*, using only the lower half of the panorama resulted in a lower error than using all feature points ($p < 0.001$ with the $t$-test and the rank sum test for both DoG and MSER). For the other rooms, no significant difference in performance was found. The use of only the upper half part of the panorama was also tested, but these results were significantly worse than using the whole panorama for the *robotics laboratory* ($p < 0.001$, $t$-test and rank sum test). Again, there was no significant difference in the *square room* and *corridor*.

## 4.3 Vardy's Panorama Database

A second database (Vardy 2005)[2] was used to evaluate the proposed method in images obtained with an omnidirectional camera. Vardy's image database consists of panoramic images acquired over a grid of equally separated points from the hall and the robotics laboratory of the Bielefeld University. He created six data sets of the laboratory and two of the hall, all under slightly different conditions, such as the amount of light and added objects. In the robotic laboratory the data set consisted of a 10 × 17 image grid with 30 cm separation between each image (horizontally and vertically); in the hall 10 × 21 images in a grid were created per data set with 50 cm separation between images. In contrast to the *IIIA database*, Vardy's database was acquired with an *ImagingSource DFK 4303* camera pointing towards an hyperbolic mirror, which directly acquires omnidirectional images, and therefore spares the panorama creation step. Nevertheless, it suffers from a much lower resolution. Figure 12 shows a panorama from the *hall1* data set. In our experiments, first all the feature points are extracted from the images. As can be seen in Figure 12, the image also contains non relevant parts which lay outside the mirror. To focus on the informative area of the image, the field of view is reduced to a limited number of degrees above and below the horizon, which is the line between the centre of the spherical mirror and the outer circle of the mirror. Only feature points which fall in this area are used for the homing method.

The vector of a feature has its origin in the image center (shown as the red dot in Figure 12) and points to the feature point. These vectors have to be normalized to 1 before calculating the ALV, because the length of the vectors only shows the distance in pixels on the image. After this, the ALVs and the home vector can be calculated as described in section 3.3.

[2] Vardy's *Panoramic Image Database* is available at `http://www.ti.uni-bielefeld.de/html/research/avardy/index.html`.

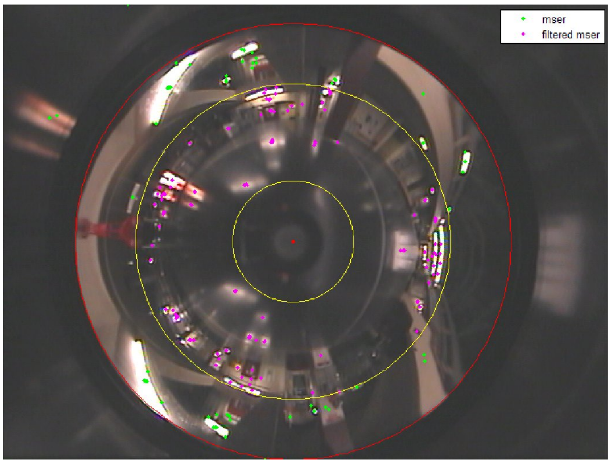


**Fig. 12** A panorama from Vardy's image database. The outer red circle shows the border of the parabolic mirror, the two inner yellow circles show the 20° line above and below the horizon. The points show the location of the MSER feature points; the filtered feature points are the ones between the yellow circles (best viewed in color).

### *4.3.1 Results*

As can be seen from Table 4 (see Annex), the scores for the Vardy data set vary from 0.85 to 0.3 and the home angle error from 28.2° ± 27.6 to 126.0° ± 43.3. The results are worse than the results with the previously discussed data sets, but it must be noticed that this data sets contain more samples.

For all feature types, a view angle of 15° (above and below the horizon) worked significantly better than a smaller view angle ($p < 0.001$, $t$-test and rank sum test) in the majority of the sets, and a view angle of 20° was best for all sets except for *doorlit* and *hall1* when MSER feature points were used, and *hall2* and *screen* when DoG points were used. For the DoG feature points case, using a view angle of 5° had the best results in the sets *hall2* ($p < 0.001$, rank sum test) and *screen* ($p < 0.05$, rank sum test).

It is also clear from Table 4 (see Annex) that again the performance of the MSER detector is better than that of the DoG detector. This difference is significant for all data sets with a view of more than 5° above and below the horizon (using the $t$-test and rank sum test; $p < 0.001$). It can also be seen from the table that the best of the IIIA sets are all above the data sets of Vardy, however this is only significant for the *robotics laboratory*. Finally, although the rooms could be assumed to be quite similar between the IIIA and the Vardy data sets as they both are flat "office like" with several desks, chairs and computers, the environments have some significant differences (e.g. the landmarks present in the *robotics laboratory* are not present in Vardy's rooms) so results comparing the two databases have to be taken with a grain of salt.

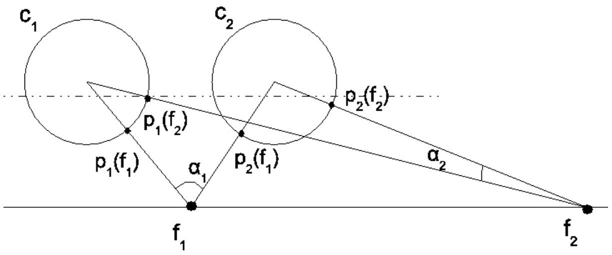


**Fig. 13** Where $c_i$ are the acquired panoramas, $f_j$ the features in the world, and $p_i(f_j)$ their projection in the images. The position where points are projected in different panoramas varies less (and therefore are less informative) if the points are far away. This is a problem for narrow and long corridors with most texture at the extremes.

## 4.4 Overall Discussion of Experimental Results

The ALV homing method combined with the proposed local feature detectors gave very positive results in the experiments with a real robot. The best results were obtained with the panoramas from the *square room*, while the results from the *corridor* were the worst, as expected. Already in the simulation it was found that the performance of the homing method is better in square rooms than in rooms with big differences in width and length, since the projections of the feature points onto a panorama are closer to each other the further away they are from the robot (see Figure 13).

An attempt to improve the results was done by trying to make the rooms, such as the corridor, more square by only using the lower half of the panorama, because then the closer objects are more prominent. This however had no significant improvement in the *corridor*, and neither in the *square room*. Only in the *robotics laboratory* there was a significant lower error ($p < 0.001$).

Looking at the difference in performance using DoG and MSER feature points it can be concluded that the use of MSER feature points significantly outperforms the use of DoG feature points. The artificial landmarks in the *robotics laboratory* were used to find out how well the method worked in comparison with invariant feature points. The results with the artificial landmarks were significantly better than using invariant feature points, but the error was only about 7° less than using MSER feature points (with only the lower half of the panorama).

Normally one should expect the homing method to work worse when the distance between the current location and the home is lower, but this relation could not be found. This might be because the room is too small or because objects occlude a big part of the field. Further work would be needed to find out if there is any relation between the distance and error.

The images of Vardy (2005) data sets were also used to test the ALV homing method. Although the different panorama acquisition system, in practice the performance of these sets was not much worse than the results of the IIIA ones. From these images also SIFT and MSER feature points were extracted and used to calculate the ALV. It was found that using almost the whole image (20° above and below the horizon) resulted in the best performance.

The scores (with 1 being best and 0 begin worst) of the IIIA data sets varied from 0.67 to 0.96, whereas the results of Vardy's data sets varied from 0.30 to 0.85 (see Table

4 in the Annex). Looking at the best parameters however, such as using the lower half of the panorama for the IIIA data sets and using a view angle of 20° above and below the horizon of Vardy's data, the scores of the IIIA data sets vary from 0.73 to 0.96 and the scores of Vardy's data sets from 0.67 to 0.85. This shows that the method performs almost as well in the different rooms and with the different types of panoramas.

Finally some comparison to other work can be made, however in most works other error measurements are used such as the distance at which it stops from home. In this work however no such experiments have been done yet. Hafner (2001) also did experiments in an office environment in a grid. After off-line learning the average error was smaller than 90° in 92% of the cases and smaller than 45° in more than 69%. This is comparable to the results in the *robotics laboratory* for the DoG feature points, and our results for using MSER feature points were even better.

The experiments by Franz et al (1998) were done in a 118 cm × 110 cm environment but the catchment area was relatively smaller than the catchment area of the IIIA data sets. Their algorithm performed robustly up to an average distance of 15 cm. They also mention experiments done in an office environment in which the algorithm performed robustly until about 2 m.

## 5 Conclusion and Future Work

In this work we propose a method for homing that, contrarily to previous works, relies only in natural landmarks detected using invariant visual feature detectors in panoramic images. This method is suitable, for example, for directing the robot from one of the nodes of a topological map to the next with the minimal cost (i.e. no matches have to be established between visual feature points of the images). Two types of invariant feature detectors were tested: the Difference of Gaussians extrema (DoG) of Lowe (2004) and the Maximally Stable Extremal Regions (MSER) of Matas et al (2002).

Although there are several methods to do homing, such as the 1D method of Hong et al (1991), warping (Franz et al 1998; Möller et al 2009) or snapshots (Lambrinos et al 2000), the *ALV homing* method (Lambrinos et al 1998, 2000) has been used mainly because of its simplicity and low computational complexity.

In order to evaluate the proposed method, initial experiments using a simulated environment were conducted and later it was tested in a real world scenario. The real world experiments were done with panoramas acquired in three different rooms at the IIIA research center using a conventional camera mounted in a pan-tilt unit, and with the panorama dataset proposed by Vardy (2005), acquired using an omnidirectional camera.

The ALV homing was found to be a good working method, however the method performed worse in rooms with very different width and length. This has been explained by the way the feature points are projected on the panorama and by the *equal distance assumption* (Franz et al 1998).

We found that the results in the IIIA data sets were slightly better than those of obtained with Vardy's data sets, but this difference is not significant, and in practice does not compensate the increased acquisition time of the rotating camera.

Regarding the feature types, in our experiments MSER significantly outperformed SIFT, and was only 7° worse than using the artificial landmarks in the *robotics laboratory*. This difference seems low enough to justify the applicability of the presented homing method, since it does not require setting up the environment by placing artificial landmarks.

Clearly, the next step is the evaluation of the method in complex navigation experiments combined with a topological localization method like the one proposed by Ramisa et al (2009). For the homing method to work in real-time the panoramas should be created faster, so the use of a camera and parabolic mirror is a good option. Furthermore, conducting real navigation experiments will require the robot to have a means to obtain its global orientation. A digital magnetic compass connected to the robot can be used for this task; however, Hafner (2001) mentioned that a magnetic compass does not work very well inside buildings, therefore she used camera information to compensate for that. Extra experiments should be done to verify the stability of the compass orientation. Other options to recover orientation from the visual sensors include that of Lambrinos et al (2000), who used a polarized-light compass which worked good, but needed sunlight from all directions above it and glass windows depolarize the light, therefore it cannot be used inside buildings. Vardy (2005) proposed to use the coherence of flow fields as an indicator of correct orientation, and finally Zeil et al (2003) suggested to use the difference between images to align them by using one-dimensional gradient descent.

Another possible improvement could be using a machine learning method to discard spurious feature points, for example by tracking feature points in a training sequence and modeling those with a low repeatability rate. These feature points are a source of noise for the homing method, and discarding them could improve significantly the results.

## References

Busquets D (2003) A multiagent approach to qualitative navigation in robotics. PhD dissertation, Universitat Politècnica de Catalunya

Busquets D, Sierra C, de Mantaras RL (2003) A multiagent approach to qualitative landmark-based navigation. Autonomous Robots 15(2):129–154

Carwright BA, Collet TS (1983) Landmark learning in bees: Experiments and models. Journal of Comparative Physiology 151:521–543

Chaudhari P (2010) Localization using Average Landmark Vector in the presence of clutter. In: Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on, IEEE, pp 1592–1595

Csurka G, Bray C, Dance C, Fan L (2004) Visual categorization with bags of keypoints. Workshop on Statistical Learning in Computer Vision, ECCV pp 1–22

Edelman S, Intrator N, Poggio T (1997) Complex cells and object recognition, unpublished manuscript, University of Cornell

Franz M, Stürzl W, Hübner W, Mallot H (2008) A Robot System for Biomimetic Navigation–From Snapshots to Metric Embeddings of View Graphs. Robotics and cognitive approaches to spatial mapping pp 297–314

Franz MO, Schölkopf B, Mallot HA, , Bülthoff HH (1998) Where did i take that snapshot? scene-based homing by image matching. Biological Cybernetics 79:191–202

Goedemé T, Nuttin M, Tuytelaars T, Van Gool L (2007) Omnidirectional vision based topological navigation. International Journal of Computer Vision 74(3):219–236

Goldhoorn A (2008) Solving ambiguity in global localization of autonomous robots. Master's thesis, University of Groningen

Goldhoorn A, Ramisa A, de Mantaras RL, Toledo R (2007a) Robot homing simulations using the average landmark vector method. Tech. Rep. RR-IIIA-2007-03, IIIA-CSIC, Bellaterra

Goldhoorn A, Ramisa A, de Mantaras RL, Toledo R (2007b) Using the average landmark vector method for robot homing. In: 19th International Conference of the ACIA, IOS Press, Frontiers in Artificial Intelligence and Applications, vol 163, pp 331–338

Hafner V, Möller R (2001) Learning of Visual Navigation Strategies. In: European Workshop on Learning Robots (EWLR), Prague

Hafner VV (2001) Adaptive homing: Robotic exploration tours. Adaptive Behavior 9(3/4):131–141

Hong J, Tan X, Pinette B, Weiss R, Riseman E (1991) Image-based homing. In: Proceedings of the 1991 IEEE International Conference on Robotics and Automation, pp 620 – 625

Labrosse F (2007) Short and long-range visual navigation using warped panoramic images. Robotics and Autonomous Systems 55(9):675–684

Lambrinos D, Möller R, Pfeifer R, Wehner R (1998) Landmark navigation without snapshots: the average landmark vector model. In: Elsner N, Wehner R (eds) Proc 26th Göttingen Neurobiology Conference, Thieme-Verlag

Lambrinos D, Möller R, Labhart T, Pfeifer R, Wehner R (2000) A mobile robot employing insect strategies for navigation. Robotics and Autonomous Systems 30(1-2):39–64

Lindeberg T, Gårding J (1997) Shape-adapted smoothing in estimation of 3-D shape cues from affine deformations of local 2-D brightness structure. Image Vision Comput 15(6):415–434

López-Nicolis G, Guerrero J, Sagüés C (2010) Multiple homographies with omnidirectional vision for robot homing. Robotics and Autonomous Systems 58(6):773–783

Lowe DG (1999) Object recognition from local scale-invariant features. In: ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2, IEEE Computer Society, Washington, DC, USA, pp 1150–1157

Lowe DG (2004) Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2):91–110

Matas J, Chum O, Martin U, Pajdla T (2002) Robust wide baseline stereo from maximally stable extremal regions. In: Rosin PL, Marshall D (eds) Proceedings of the British Machine Vision Conference, BMVA, London, UK, vol 1, pp 384–393

McMillan L, Bishop G (1995) Plenoptic modeling: an image-based rendering system. In: SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques, ACM Press, New York, NY, USA, pp 39–46

Mikolajczyk K, Tuytelaars T, Schmid C, Zisserman A, Matas J, Schaffalitzky F, Kadir T, Gool L (2005) A comparison of affine region detectors. International Journal of Computer Vision 65:43–72(30)

Möller R (2000) Insect visual homing strategies in a robot with analog processing. Biological Cybernetics, special issue: Navigation in Biological and Artificial Systems 83:231–243

Möller R (2009) Local visual homing by warping of two-dimensional images. Robotics and Autonomous Systems 57(1):87–101

Möller R, Lambrinos D, Roggendorf T, Pfeifer R, Wehner R (2001) Insect strategies of visual homing in mobile robots. In: Webb B, Consi TR (eds) Biorobotics. Methods and Applications, AAAI Press / MIT Press, pp 37–66

Möller R, Vardy A, Kreft S, Ruwisch S (2007) Visual homing in environments with anisotropic landmark distribution. Autonomous Robots 23(3):231–245

Möller R, Krzykawski M, Gerstmayr L (2009) Three 2D-warping schemes for visual robot navigation. Autonomous Robots pp 1–39

Murphy-Chutorian E, Trivedi M (2006) N-tree disjoint-set forests for maximally stable extremal regions. In: Proceedings of the British Machine Vision Conference

Pons J, Hübner W, Dahmen H, Mallot H (2007) Vision-based robot homing in dynamic environments. In: 13th IASTED International Conference on Robotics and Applications, K. Schilling, Ed, Citeseer, pp 293–298

Ramisa A (2006) Qualitative navigation using panoramas. Master's thesis, Universitat Autònoma de Barcelona

Ramisa A, Tapus A, Lopez de Mantaras R, Toledo R (2008) Mobile robot localization using panoramic vision and combinations of feature region detectors. In: Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on, pp 538–543

Ramisa A, Tapus A, Aldavert D, Toledo R, Lopez de Mantaras R (2009) Robust vision-based robot localization using combinations of local feature region detectors. Autonomous Robots 27(4):373–385

Shum H, Szeliski R (1997) Panoramic image mosaics. Tech. Rep. MSR-TR-97-23, Microsoft Research

Szeliski R, Shum HY (1997) Creating full view panoramic image mosaics and environment maps. In: SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, vol 31, pp 251–258

Usher K, Ridley P, Corke P (2003) Visual servoing of a car-like vehicle-an application of omnidirectional vision. In: Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on, vol 3

Valgren C, Lilienthal AJ (2008) Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments. In: Proc. IEEE Int. Conf. on Robotics and Automation, pp 1856–1861

Vardy A (2005) Biologically plausible methods for robot visual homing. PhD dissertation, Carleton University

Vinyals M, Ramisa A, Toledo R (2007) An Evaluation of an Object Recognition Schema Using Multiple Region Detectors. FRONTIERS IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS 163:213

Wehner R (1987) Spatial organization of foraging behavior in individually searching desert ants, cataglyphis (sahara desert) and ocymyrmex (namib desert). Experientia Supplementum 54:15–42

Wehner R (1994) The polarization-vision project: championing organismic biology. Progr Zool (Fortschr Zool) 39:103–143

Zeil J, Hofmann M, Chahl J (2003) Catchment areas of panoramic snapshots in outdoor scenes. JOSA-A 20(3):450–469

## Annex

| type | detector | mean | median | std. dev. | score | best home | n | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | square room | upper half | MSER | 6,83 | 4,10 | 5,33 | 0,9621 | 138 | 3 |
| 2 | square room | not filtered | MSER | 9,65 | 12,03 | 7,84 | 0,9464 | 138 | 3 |
| 3 | square room | not filtered | DoG | 13,78 | 12,00 | 11,31 | 0,9234 | 138 | 3 |
| 4 | robot lab | not filtered | Landmarks | 14,88 | 10,16 | 14,86 | 0,9173 | 110 | 38 |
| 5 | square room | lower half | DoG | 14,94 | 14,52 | 10,75 | 0,9170 | 138 | 3 |
| 6 | square room | lower half | MSER | 20,62 | 25,27 | 8,49 | 0,8855 | 138 | 3 |
| 7 | square room | upper half | DoG | 20,91 | 18,96 | 6,64 | 0,8838 | 138 | 3 |
| 8 | robot lab | lower half | MSER | 21,96 | 11,09 | 30,05 | 0,8780 | 117 | 38 |
| 9 | day | 20 | MSER | 26,18 | 18,73 | 27,62 | 0,854 | 17 | 170 |
| 10 | robot lab | lower half | DoG | 26,90 | 13,05 | 34,74 | 0,8506 | 117 | 38 |
| 11 | robot lab | not filtered | MSER | 27,84 | 16,03 | 35,51 | 0,8454 | 117 | 38 |
| 12 | screen | 20 | MSER | 28,64 | 18,42 | 31,04 | 0,840 | 95 | 170 |
| 13 | doorlit | 15 | MSER | 30,69 | 19,35 | 33,38 | 0,829 | 15 | 170 |
| 14 | arboreal | 20 | MSER | 34,89 | 23,39 | 35,49 | 0,806 | 50 | 170 |
| 15 | doorlit | 20 | MSER | 35,41 | 21,27 | 38,52 | 0,803 | 50 | 170 |
| 16 | robot lab | not filtered | DoG | 35,60 | 22,85 | 38,67 | 0,8022 | 117 | 38 |
| 17 | arboreal | 15 | MSER | 37,83 | 25,31 | 37,20 | 0,789 | 17 | 170 |
| 18 | day | 15 | MSER | 39,78 | 29,30 | 36,49 | 0,779 | 17 | 170 |
| 19 | hall1 | 15 | MSER | 42,61 | 31,55 | 38,32 | 0,763 | 159 | 200 |
| 20 | original | 20 | MSER | 43,18 | 32,23 | 38,49 | 0,760 | 50 | 170 |
| 21 | screen | 15 | MSER | 45,71 | 33,75 | 40,43 | 0,746 | 0 | 170 |
| 22 | hall1 | 10 | MSER | 45,81 | 35,12 | 39,05 | 0,745 | 41 | 200 |
| 23 | twilight | 20 | MSER | 46,21 | 34,90 | 39,72 | 0,743 | 50 | 170 |
| 24 | doorlit | 10 | MSER | 48,45 | 33,95 | 43,90 | 0,730 | 14 | 170 |
| 25 | hall | lower half | MSER | 48,83 | 39,02 | 41,63 | 0,7287 | 203 | 6 |
| 26 | arboreal | 10 | MSER | 49,97 | 35,14 | 44,80 | 0,722 | 153 | 170 |
| 27 | robot lab | upper half | MSER | 50,62 | 39,33 | 42,47 | 0,7188 | 117 | 38 |
| 28 | winlit | 20 | MSER | 52,39 | 39,58 | 44,07 | 0,708 | 50 | 170 |
| 29 | corridor | not filtered | MSER | 52,66 | 35,71 | 44,89 | 0,7074 | 200 | 6 |
| 30 | robot lab | upper half | DoG | 56,14 | 45,77 | 43,84 | 0,6881 | 117 | 38 |
| 31 | corridor | not filtered | DoG | 56,26 | 44,58 | 43,64 | 0,6874 | 203 | 6 |
| 32 | corridor | lower half | DoG | 56,45 | 49,69 | 42,39 | 0,6864 | 203 | 6 |

Continued

| | **dataset** | **type** | **detector** | **mean** | **median** | **std. dev.** | **score** | **best home** | **n** |
|---|---|---|---|---|---|---|---|---|---|
| 33 | corridor | upper half | DoG | 57,08 | 38,19 | 45,65 | 0,6829 | 203 | 6 |
| 34 | twilight | 15 | MSER | 57,63 | 44,59 | 46,70 | 0,679 | 153 | 170 |
| 35 | hall1 | 20 | MSER | 58,49 | 48,71 | 44,53 | 0,675 | 99 | 200 |
| 36 | original | 15 | MSER | 58,53 | 45,11 | 47,56 | 0,674 | 17 | 170 |
| 37 | chairs | 20 | MSER | 58,92 | 45,23 | 47,55 | 0,672 | 84 | 170 |
| 38 | corridor | upper half | MSER | 59,15 | 42,56 | 46,02 | 0,6714 | 199 | 6 |
| 39 | hall2 | 20 | MSER | 59,51 | 49,20 | 43,09 | 0,669 | 18 | 200 |
| 40 | hall2 | 15 | MSER | 61,00 | 50,90 | 45,30 | 0,661 | 18 | 200 |
| 41 | day | 10 | MSER | 62,50 | 51,30 | 47,17 | 0,652 | 17 | 170 |
| 42 | hall1 | 5 | MSER | 62,69 | 53,82 | 44,78 | 0,651 | 39 | 200 |
| 43 | screen | 10 | MSER | 63,43 | 54,30 | 45,02 | 0,647 | 153 | 170 |
| 44 | screen | 5 | MSER | 66,71 | 52,61 | 50,03 | 0,629 | 102 | 170 |
| 45 | hall2 | 5 | MSER | 68,57 | 57,22 | 49,19 | 0,619 | 19 | 200 |
| 46 | original | 10 | MSER | 72,09 | 62,67 | 50,85 | 0,599 | 14 | 170 |
| 47 | winlit | 15 | MSER | 72,39 | 63,58 | 50,84 | 0,597 | 50 | 170 |
| 48 | day | 5 | MSER | 72,67 | 62,94 | 50,31 | 0,596 | 169 | 170 |
| 49 | hall2 | 10 | MSER | 72,72 | 62,01 | 50,68 | 0,596 | 19 | 200 |
| 50 | twilight | 10 | MSER | 73,55 | 65,58 | 51,23 | 0,591 | 18 | 170 |
| 51 | hall1 | 20 | DoG | 77,16 | 71,90 | 45,43 | 0,571 | 0 | 200 |
| 52 | winlit | 10 | MSER | 78,69 | 72,12 | 51,96 | 0,562 | 16 | 170 |
| 53 | chairs | 15 | MSER | 79,61 | 72,79 | 51,91 | 0,557 | 16 | 170 |
| 54 | doorlit | 5 | DoG | 80,15 | 75,59 | 52,15 | 0,554 | 169 | 170 |
| 55 | doorlit | 20 | DoG | 83,07 | 80,38 | 49,17 | 0,538 | 151 | 170 |
| 56 | doorlit | 10 | DoG | 83,79 | 81,13 | 50,27 | 0,534 | 169 | 170 |
| 57 | chairs | 10 | MSER | 84,42 | 82,22 | 50,74 | 0,531 | 14 | 170 |
| 58 | doorlit | 15 | DoG | 84,44 | 81,34 | 49,75 | 0,530 | 152 | 170 |
| 59 | chairs | 20 | DoG | 86,15 | 82,29 | 49,48 | 0,521 | 3 | 170 |
| 60 | screen | 5 | DoG | 86,87 | 85,63 | 51,82 | 0,517 | 135 | 170 |
| 61 | screen | 15 | DoG | 88,42 | 85,67 | 51,75 | 0,508 | 135 | 170 |
| 62 | screen | 10 | DoG | 88,76 | 88,36 | 52,02 | 0,506 | 135 | 170 |
| 63 | screen | 20 | DoG | 89,25 | 86,89 | 51,88 | 0,504 | 3 | 170 |
| 64 | hall1 | 15 | DoG | 89,27 | 85,64 | 45,78 | 0,504 | 0 | 200 |
| 65 | chairs | 15 | DoG | 90,33 | 87,09 | 51,16 | 0,498 | 4 | 170 |
| 66 | arboreal | 20 | DoG | 90,33 | 88,88 | 50,27 | 0,498 | 4 | 170 |
| 67 | original | 20 | DoG | 91,36 | 88,78 | 49,70 | 0,492 | 3 | 170 |
| 68 | twilight | 20 | DoG | 91,66 | 89,34 | 49,59 | 0,490 | 5 | 170 |
| 69 | day | 10 | DoG | 92,99 | 94,81 | 51,31 | 0,483 | 152 | 170 |
| 70 | day | 15 | DoG | 93,00 | 94,20 | 50,95 | 0,483 | 135 | 170 |
| 71 | day | 20 | DoG | 93,05 | 93,13 | 50,42 | 0,483 | 135 | 170 |
| 72 | day | 5 | DoG | 93,10 | 94,15 | 51,72 | 0,482 | 152 | 170 |
| 73 | chairs | 10 | DoG | 93,46 | 92,24 | 51,88 | 0,480 | 4 | 170 |
| 74 | chairs | 5 | DoG | 93,51 | 92,00 | 50,99 | 0,480 | 5 | 170 |
| 75 | arboreal | 15 | DoG | 95,20 | 95,05 | 51,65 | 0,471 | 4 | 170 |
| 76 | twilight | 15 | DoG | 96,44 | 96,35 | 50,29 | 0,464 | 5 | 170 |
| 77 | winlit | 5 | MSER | 97,11 | 103,58 | 54,36 | 0,460 | 136 | 170 |
| 78 | original | 15 | DoG | 97,93 | 97,66 | 49,81 | 0,455 | 4 | 170 |
| 79 | winlit | 20 | DoG | 98,86 | 99,65 | 44,11 | 0,450 | 0 | 170 |
| 80 | arboreal | 10 | DoG | 99,07 | 101,86 | 51,64 | 0,449 | 135 | 170 |
| 81 | arboreal | 5 | DoG | 100,55 | 104,77 | 51,59 | 0,441 | 135 | 170 |
| 82 | twilight | 10 | DoG | 101,25 | 102,97 | 50,17 | 0,437 | 6 | 170 |
| 83 | hall1 | 10 | DoG | 101,86 | 99,59 | 44,55 | 0,434 | 40 | 200 |

Continued

| | dataset | type | detector | mean | median | std. dev. | score | best home | n |
|---|---|---|---|---|---|---|---|---|---|
| 84 | original | 10 | DoG | 101,98 | 105,32 | 50,19 | 0,433 | 4 | 170 |
| 85 | twilight | 5 | DoG | 102,16 | 105,37 | 49,95 | 0,432 | 6 | 170 |
| 86 | doorlit | 5 | MSER | 102,79 | 110,41 | 51,90 | 0,429 | 14 | 170 |
| 87 | winlit | 5 | DoG | 103,01 | 109,13 | 46,78 | 0,427 | 134 | 170 |
| 88 | original | 5 | DoG | 103,19 | 108,47 | 50,36 | 0,426 | 50 | 170 |
| 89 | winlit | 15 | DoG | 103,28 | 105,54 | 45,33 | 0,426 | 34 | 170 |
| 90 | winlit | 10 | DoG | 104,93 | 109,14 | 46,08 | 0,417 | 135 | 170 |
| 91 | hall1 | 5 | DoG | 108,85 | 109,20 | 46,09 | 0,395 | 80 | 200 |
| 92 | arboreal | 5 | MSER | 112,73 | 122,76 | 48,51 | 0,373 | 14 | 170 |
| 93 | chairs | 5 | MSER | 116,47 | 126,06 | 46,16 | 0,352 | 14 | 170 |
| 94 | hall2 | 5 | DoG | 116,47 | 130,14 | 49,96 | 0,352 | 198 | 200 |
| 95 | original | 5 | MSER | 118,50 | 128,27 | 44,93 | 0,341 | 153 | 170 |
| 96 | hall2 | 20 | DoG | 122,09 | 132,21 | 44,38 | 0,321 | 20 | 200 |
| 97 | twilight | 5 | MSER | 122,21 | 133,26 | 44,17 | 0,321 | 136 | 170 |
| 98 | hall2 | 10 | DoG | 124,42 | 137,90 | 45,31 | 0,308 | 199 | 200 |
| 99 | hall2 | 15 | DoG | 125,99 | 137,57 | 43,33 | 0,300 | 61 | 200 |

**Table 4** This table shows the results of all real world experiments (IIIA in white and Vardy in light gray) sorted by score. For the IIIA data set, the type column shows which part of the panorama has been used: all feature points (*not filtered*), only the feature points at the *lower half* of the panorama or only at the *upper half* (see Section 4.2.4). For Vardy data set, the type column shows the number of degrees above and below the horizon of the image which were used. The detector column shows which feature detector has been used to perform homing: DoG, MSER or artificial *landmarks* which were only available in the *robot laboratory*. The next three columns: mean, median and std. dev. (standard deviation) show information about the direction error of the home vector in degrees. The calculation of the score is shown in Eqn. 7; 1 being best and 0 being worst. The *best home* column shows the ID of the location of the home where to the mean error is smallest. Finally the $n$ column shows the number of samples, i.e. different panoramas, for the data set.