Omnidirectional Vision based Topological Navigation

Toon Goedemé[1]*, Marnix Nuttin[2], Tinne Tuytelaars[1], and Luc Van Gool[1,3]
[1]PSI–VISICS, K. U. Leuven, Kasteelpark 10, 3001 Leuven, Belgium
[2]PMA, K. U. Leuven, Celestijnenlaan 300, 3001 Leuven, Belgium
[3]BIWI, ETH, Sternwartstraße 7, 8092 Zürich, Switzerland

Abstract

This work presents a unique system for autonomous mobile robot navigation. The main sensor is an omnidirectional camera. The proposed system is capable to build automatically a topological map complex, natural environments. It can localise itself using that map on each moment, after startup (kidnapped robot) or using knowledge of former localisations. The topological nature of the map enables fast and simple path planning towards a specified goal. A visual servoing technique is implemented to steer the system along the computed path.

The key technology making this all possible is the novel *fast wide baseline feature matching*, which yields an efficient abstraction of the wealth of information offered by the visual sensor.

1 Introduction

We aim at a *vision-only* application, i.e. the entire application uses mainly one visual sensor. The advantage is that it is cost-efficient and easy to install on a mobile system. We chose to use an omnidirectional camera as visual sensor, because of its wide field of view and thus rich information content of the images acquired with. For the time being, we added a range sensing device for obstacle detection, such as a lidar or a set of ultrasound sensors.

Our method works with *natural* environments. That means that the environment does not has to be modified before navigation can be done in it. Indeed, adding artificial markers to every room in a house or to an entire city doesn't seem very feasible.

*Contact address: Toon.Goedeme@esat.kuleuven.be

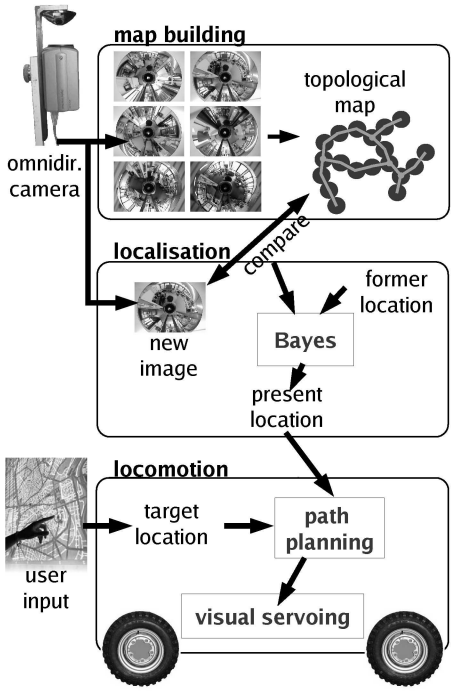

Figure 1: *Overview of the navigation method*

In opposition to classical navigation methods, we chose a *topological* representation of the environment, rather than a metrical one, because of its flexibility, wide usability and memory-efficiency.

1.1 Application

The targeted application of this research is the visual guidance of electric wheelchairs for severely disabled patients. These are not able to give detailed steering commands to navigate around in their homes and local city neighbourhoods. If it is possible for them to perform complicated navigational tasks by only giving simple commands, their autonomy can be greatly enhanced. For these patients such an increase of mobility and independence from other people is very welcome.

1.2 Method overview

An overview of the navigation method presented is given in fig. 1. The system can be subdivided in three parts: map building, localisation and locomotion.

The map building stage is a training procedure and has to be done only once

to train the system in a new environment. The mobile system is lead through all parts of the environment, while it takes images at a constant rate. Later, this large set of omnidirectional images is automatically analysed and converted in a topological map (see section 4) of the environment, which is stored in the system's memory. This map is used in the next parts of the algorithm.

The next stage is localisation (section 5). When the system is powered up somewhere in the environment, it takes a new image with its camera. This image is rapidly compared with all the images in the environment map, and an hypothesis is formed about the present location of the mobile robot. When later new images come in, this hypothesis is refined using Bayes' rule.

Wen the present location of the robot is known and a goal position is communicated by the user to the robot, a path can be planned towards that goal using the map (section 6). This map, which is defined as a sequence of training images, is then executed by means of a visual servoing algorithm (section 7).

The remainder of this text is organised as follows. The next section gives an overview of the related work on total navigation solutions and the parts of it. In section 3, our core image analysis and matching technique is explained, fast wide baseline matching. The sections thereafter describe the different parts of our approach. We conclude the paper with an overview of experimental results (section 8) and a conclusion (section 9).

2 Related Work

Traditionally, other sensors are used for robot navigation, like GPS and laser scanners. Because GPS needs a direct line of sight to the satellites, it can not be used indoors or in narrow city centre streets we forsee in our application. Time-of-flight laser scanners are widely applicable, but are expensive and voluminous, except when the scanning field is restricted to a horizontal plane. The latter only yields a poor world representation, with the risk of not detecting essential obstacles such as table tops.

That is why we propose a vision based solution to navigation. Vision is, in comparison with these other sensors, much more informative. We observe that many biological species, in particular flying animals, use mainly their visual sensors for navigation. Moreover, we see that the majority of insects and arthropods benefit from a wide field of view, which sustains our omnidirectional camera choice.

Many researchers proposed different ways to represent the environment perceived by vision sensors. One approach is building dense 3D models out of the incoming visual data [1, 2]. Although, this method has some inherent disadvantages. It is computationally and memory demanding, and fails to model planar and less-textured parts of the environment such as walls. Especially in urban large

environments these disadvantages restrict the use severely.

One way out of the computational burden is making abstraction of the visual data. Instead of modelling a dense 3D model containing billions of voxels, a sparse 3D model is built containing only special features or *visual landmarks*. A simple solution to do this abstraction is adding artificial markers to strategically chosen places in the world. To make these features easily detectable with a normal camera, they are given special photometric appearances (for instance coloured patterns [3] or even 2D barcodes [4]). Using such artificial markers is perfectly possible for some applications, but often difficult. Navigation through an entire city or inside someone's house are examples of cases where pasting these markers everywhere around raises questions.

That is why, in this project we use *natural landmarks* that are available in most scenes. Hence, no special markers are required, as these landmarks can be extracted from the scene itself. Moreover, the extraction of these landmarks must be automatic and robust against changes in viewpoint and illumination to ensure the detection of these landmarks in as much circumstances as possible.

Many researchers proposed algorithms for natural landmarks. Mostly, local regions are defined around interest points in the images. The characterisation of these local regions with descriptor vectors enables the regions to be compared across images. Differences between approaches lie in the way in which interest points, local image regions, and descriptor vectors are extracted. Some well-known wide baseline matching features are the ones of Schmid and Mohr [5], Lowe *et al.* [6], Tuytelaars & Van Gool [7], Matas *et al.* [8], and Mikolajczyk & Schmid [9].

Although these methods are capable to find very qualitative correspondences, most of them are too slow to use in a real-time mobile robot algorithm. That is why we spent efforts to speed this up, as explained in section 3.

Examples of researchers solving the navigation problem with sparse 3D maps of natural landmarks are Se *et al.* [10] and Davison [11]. They position natural features in a metrical axis frame, which is as big as the entire mapped environment. Although less than the dense 3D variant, these methods are still computationally demanding for large environments since their complexity is quadratic in the number of features in the model. Also, for larger models the metric error accumulates, so that feature positions are drifting away.

One step further in the abstraction of environment information is the introduction of topological maps. Locally, places are described as a set of natural landmarks. These places form the nodes of the graph-like map, and are interconnected by traversable paths. Other researchers [12, 13] also chose for probabilistic topological maps because they scale better to real-world applications than metrical, deterministic representations, given the complexity of unstructured environments and its inherent uncertainty. Other advantages are the ease of path planning in such a map and the absence of drift.

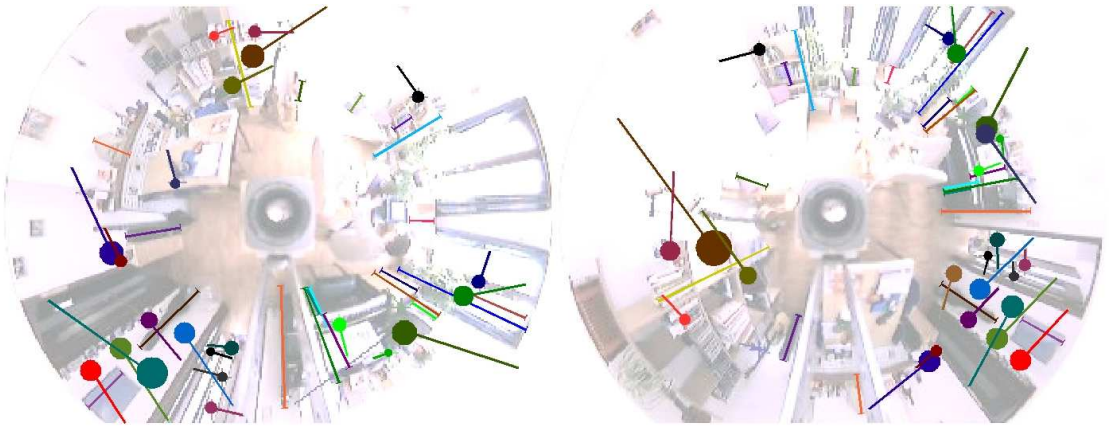

Figure 2: *A pair of omnidirectional images, superimposed with colour-coded corresponding column segments (radial lines) and SIFT features (circles with tail).*

3 Fast wide baseline matching

The novel technique we use for image comparison is *fast wide baseline matching*. This key technique enables extraction of natural landmarks and image comparison for our map building, localisation and visual servoing algorithms.

We use a combination of two different kinds of these wide baseline features, namely a rotation reduced and colour enhanced form of Lowe's *SIFT features* [6] (see also [17]), and the *invariant column segments* we developed [14]. These techniques extract local regions in each image, and describe these regions with measures that are invariant to image deformations and illumination changes. Across different images, similar regions can be found by comparing these descriptors. This makes it possible to find correspondences between images that are taken widely apart, or under different lighting conditions. The crux of the matter is that the extraction of these regions can be done beforehand on each image separately. Database images can be processed off-line, so that the image pixel data itself must not be present at the time of matching with another image.

Fig. 2 shows the matching results on a pair of omnidirectional images. As seen in these examples, the SIFT features and the column segments are complementary, which pleads for the combined use of the two. The computing time required to extract features in two 320×240 images and find correspondences between them is about 800 ms for the enhanced SIFT features and only 300 ms for the vertical column segments (on a 800 MHz laptop). Typically 30 to 50 correspondences are found.

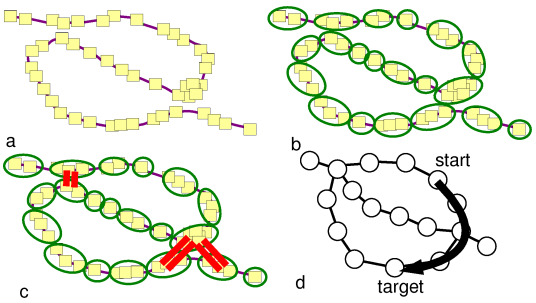

Figure 3: *Topological map building. (a) Squares denote places where training images were taken. (b) Place grouping of images. (c) Cross-place matching. (d) In the resulting topological map, a path is found between a certain start and target place.*

4 Map Building

One of the properties of the navigation approach here described is that it is able to construct automatically a topological world representation out of a sequence of training images. During a training tour through the entire environment, omni-directional images are taken at regular time intervals. The order of the training images is known.

To be of use in the following parts of the navigation method, this topological map must describe all places in the environment and the possible connections between these places. The topology of the world, being the maze of streets in a city or the structure of a house, must be reflected in the world model.

Figure 3 gives a small example to explain the algorithm used. An S-shaped path was travelled, yielding images depicted as small rectangles in the figure (a). We observe that the spatial density of the training images is not homogeneous, because the exploring speed is not constant. Therefore, *place clustering* is performed. To compare images, we developed a image dissimilarity measure which of two levels [19]. We first compare two images with a coarse but fast global technique. After that, a relatively slower comparison with more precision based on local features only has to be carried out on the survivors of the first stage. This combined dissimilarity measure is computed between consecutive images, and *places* with consant size are formed (ellipses in (b)). For each place, a prototype image is automatically chosen. A next phase in the algorithm performs an exhaustive comparison of all non-consecutive place prototypes to find locations that are visited more than once during the training session. The place prototype pairs with a dissimilarity under a predefined threshold get an extra connection between them (c). This yields the topological map depicted in figure (d).

5 Localisation

When the system has learnt a topological map of an environment, this map can be used for a variety of navigational tasks, firstly *localisation*. For each arbitrary new position in the known environment, the system can find out *where* it is. The output of this localisation algorithm is a *location*, which is—opposed to other methods like GPS—not expressed as a metric coordinate, but as a topological place.

Actually, two localisation modes exist. When starting up the system, there is no *a priori* information on the location. Every location is equally probable. This is called *global localisation*, alias the *kidnapped robot problem*. Traditionally, this is known to be a very hard problem in robot localisation. In contrary, if there is knowledge about a former localisation not too long ago, the locations in the proximity of that former location have a higher probability than others further away. This is called *location updating*.

In [19], we proposed a Bayesian system that is able to cope with both localisation modes. Instead of making a hard decision about the location, a probability value is given to each location at each time instant.

6 Path planning

With the method of the previous section, at each time instant the most probable location of the robot can be found, from which a path to a goal can be determined. How the user of the system, for instance the wheelchair patient, gives the instruction to go towards a certain goal is highly dependent on the situation. For every disabled person, for instance, an individual interface must be designed adapted to his/hers possibilities.

We assume a certain goal is expressed as a certain place of the topological map. From the present pose, computed by the localisation algorithm, a path can be easily found towards it using Dijkstra's algorithm [16]. This path is expressed as a series of topological places which have to be traversed.

7 Visual servoing

The algorithm described in this section makes the robot move along a path, computed by the previous section. Such a path is given as a sparse set of prototype images of places. The physical distance between two consecutive path images is variable (1 to 5 metres in our tests), but the visual distance is constant, and as such that there are enough local feature matches as needed by this algorithm.

As sketched in fig. 3 (d), following such a sparse visual path boils down to a succession of *visual homing* operations. First, the robot is driven towards the place

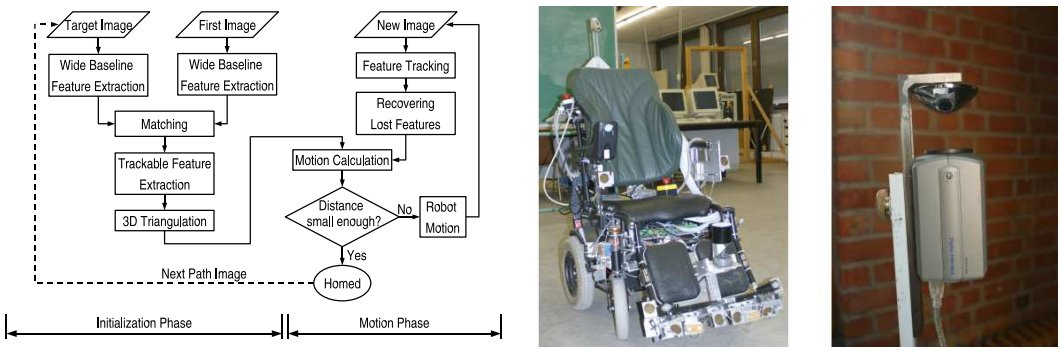

Figure 4: *Left: Flowchart of the proposed algorithm. Center: the robotic wheelchair platform. Right: the omnidirectional camera, composed by a colour camera and an hyperbolic mirror.*

where the first image on the path is taken. When arrived, it is driven towards the next path image, and so on. Because a smooth path is desired for the application, the motion must be continuous without stops at path image positions.

As explained in [20], we chose to tackle this problem by estimating locally the spatial structure of the wide baseline features using epipolar geometry. However, this decision poses no ambiguity with the non-metrical topological approach of our entire navigation method. Because the built-up sparse 3D maps are used only locally errors are kept local.

Fig. 4 offers an overview of the proposed method. Each of the *visual homing* operations is performed in two phases, an initialisation phase and an iterated motion phase.

8 Experiments

We have implemented the proposed algorithm, using our modified electric wheelchair "Sharioto". A picture of it is shown in the left of fig. 4. It is a standard electric wheelchair that has been equipped with an omnidirectional vision sensor (consisting of a Sony firewire color camera and a hyperbolic mirror, right in fig. 4). The image processing is performed on a 1 GHz laptop. As additional sensors for obstacle detection, 16 ultrasound sensors and a Lidar are added. A second laptop with a 840 MHz processor reads these sensors, receives visual homing vector commands, computes the necessary manoeuvres, and drives the motors via a CAN-bus.

Experimental results of the map building algorithm can be found in [18], results of the localisation and path planning in [19], and visual servoing results are detailed in [20].

9 Conclusion

This paper describes and demonstrates a novel approach for a mobile robot to navigate automonously in a large, natural complex environment. The only sensor is an omnidirectional colour camera. As environment representation, a topological map is chosen. This is more flexible and less memory demanding than metric 3D maps. Moreover, it does not show error build-up and enables fast path planning. As natural landmarks, we use two kinds of fast wide baseline features which we developed and adapted for this task. Because these features can be recognised even if the viewpoint is substantially different, few images are enough to describe a large environment.

Our experiments show that our system is able to build autonomously map of a natural environment it drives through. The localisation ability, with and without knowledge of previous locations, is demonstrated. With this map, very efficiently a path is computed towards each desired location. Our experiments with a real robotic wheelchair show the feasibility of executing such a path as a succession of visual servoing steps.

Future work includes the development of a omnidirectional vision-based obstacle detection, to eliminate the need of additional sensors and reducing the cost.

Acknowledgments

This work is partially supported by the Inter-University Attraction Poles, Office of the Prime Minister (IUAP-AMS), the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen), and the Fund for Scientific Research Flanders (FWO-Vlaanderen, Belgium).

References

[1] M. Pollefeys, L. Van Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, R. Koch, "Visual modeling with a hand-held camera," IJCV 59(3), 207-232, 2004.

[2] D. Nistér, O. Naroditsky, J. Bergen, "Visual Odometry," Conference on Computer Vision and Pattern Recognition, Washington, DC, 2004.

[3] T. Okuma, K. Sakaue, H. Takemura, and N. Yokoya, *Real-time camera parameter estimation from images for a Mixed Reality system*, Proc. ICPR, Barcelona, Spain, 2000.

[4] J. Rekimoto and Y. Ayatsuka, "CyberCode: Designing Augmented Reality Environments with Visual Tags," Designing Augmented Reality Environments (DARE 2000), 2000.

[5] C. Schmid, R. Mohr, C. Bauckhage, "Local Grey-value Invariants for Image Retrieval," PAMI, Vol. 19, no. 5, pp. 872-877, 1997.

[6] D. Lowe, "Object Recognition from Local Scale-Invariant Features," International Conference on Computer Vision, pp. 1150-1157, 1999.

[7] T. Tuytelaars, L. Van Gool, L. D'haene, and R. Koch, "Matching of Affinely Invariant Regions for Visual Servoing," Intl. Conf. on Robotics and Automation, pp. 1601-1606, 1999.

[8] J. Matas, O. Chum, M. Urban and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions," BMVC, Cardiff, Wales, pp. 384-396, 2002.

[9] K. Mikolajczyk and C. Schmid, "An affine invariant interest point detector," ECCV, vol. 1, 128–142, 2002.

[10] S. Se, D. Lowe, and J. Little, "Local and Global Localization for Mobile Robots using Visual Landmarks," In proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '01), Hawaii, USA, 2001.

[11] A. Davison, "Real-time simultaneous localisation and mapping with a single camera," Intl. Conf. on Computer Vision, Nice, 2003.

[12] A. Vale, M. Isabel Ribeiro, *Environment Mapping as a Topological Representation*, Proceedings of the 11th International Conference on Advanced Robotics - ICAR2003 Universidade de Coimbra, Portugal, June 30 - July 1- 3, 2003.

[13] I. Ulrich and I. Nourbakhsh, *Appearance-Based Place Recognition for Topological Localisation*, ICRA, San Francisco, CA, April 2000, pp. 1023-1029

[14] T. Goedemé, T. Tuytelaars, and L. Van Gool, "Fast Wide Baseline Matching with Constrained Camera Position," CVPR, Washington, DC, pp. 24-29, 2004.

[15] B. Kröse, J. Porta, A. van Breemen, K. Crucq, M. Nuttin, and E. Demeester, *Lino, the User-Interface Robot*, First European Symposium on Ambient Intelligence (EUSAI 2003), pp. 264-274, Veldhoven, The Netherlands, 2003.

[16] E. W. Dijkstra, *A note on two problems in connexion with graphs*, Numerische Mathematik, 1: 269-271, 1959.

[17] T. Goedemé, T. Tuytelaars, G. Vanacker, M. Nuttin and L. Van Gool, "Feature Based Omnidirectional Sparse Visual Path Following," IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2005, Edmonton, 2005.

[18] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Vision Based Intelligent Wheelchair Control: the role of vision and inertial sensing in topological navigation," INERVIS 2003, proc. of ICAR, Coimbra, 2003.

[19] T. Goedemé, M. Nuttin, T. Tuytelaars, and L. Van Gool, "Markerless Computer Vision Based Localization using Automatically Generated Topological Maps," European Navigation Conference GNSS, Rotterdam, 2004.

[20] T. Goedemé, T. Tuytelaars, G. Vanacker, M. Nuttin, and L. Van Gool, "Omnidirectional Sparse Visual Path Following with Occlusion-Robust Feature Tracking," OMNIVIS, proc. of ICCV, Beijing, 2005.