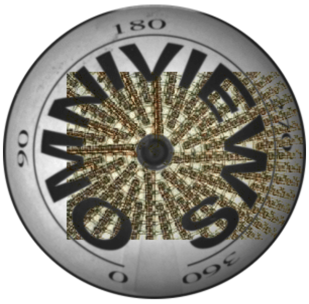
OMNIVIEWS

Omni-directional Visual System

IST-1999-29017



Vision Algorithms for OMNIVIEWS Cameras

Niall Winters, José Gaspar, Alexandre Bernardino, José Santos-Victor
Instituto Superior Técnico
Lisboa, Portugal
{niall,jag,alex,jasv}@isr.ist.utl.pt

September 2001

Contents

1 Introduction

This report describes the work done by Instituto Superior Técnico (IST) in the context of the IST-1999-29017 Project OMNIVIEWS from Sept. 1st 2000 - Sept. 1st 2001.

The work described in this report concerns the use of Omniviews cameras (detailed in [9]), for the tasks summarized below:

- **Visual Navigation** - In terms of visual navigation we have used tested the Omniviews camera for application in mobile robot navigation. The navigation approach combines a topological level and a mode of Visual Path Following.
- **Teleoperation and 3D Reconstruction** - The ability to generate images with specific geometric properties can help not only autonomous navigation but also the definition of powerful interfaces for teleoperation. We provide some early examples in this direction as well as a methodology for obtaining 3D models from (one or more) omnidirectional images which can be used for specifying goals or trajectories for a robot to follow.
- **Visual Tracking** - Techniques for tracking general targets under a rich class of image transformations is proposed and tested with Omniviews images.

In the framework of the Omniviews Project the fundamental idea is that of designing the mirror profile and the layout of the sensor in order to obtain geometric and computational benefits in the resulting image / processing.

Due to the rotational symmetry of the omnidirectional images, a natural choice is that of using an image sensor with polar structure. As a result panoramic images can be directly read out from the sensor with uniform angular resolution and without requiring any additional processing/image warping.

As the sensor radial pixel distribution is for the time being fixed (logarithm decay of resolution towards the periphery), the mirror profile provides the necessary degrees of freedom to modify the image geometry. As detailed in [9], we have considered two designs:

- **Constant vertical resolution** This viewing geometry facilitates tracking by reducing the amount of distortion that an image target undergoes when an object is moving in 3D. Finally in visual navigation it helps by providing a larger degree of image landmarks invariance w.r.t the viewing geometry.
- **Constant horizontal resolution** - As the ground plane is imaged under a scaled Euclidean transformation, it greatly facilitates the measurement of distances and angles directly from the image, as well as an easier track of points lying on the pavement with a large impact on navigation algorithms.

We have also designed a so-called *Mixed Mirror*, where the outer part of the image sensor is used to obtain a constant vertical resolution image, while the inner part is devoted to yield a constant horizontal resolution image. In this case, both the differential constraints on the mirror shape resulting from the two design goals are combined together in a single profile.

2 Visual Navigation

As previously stated, the fundamental idea behind the OMNIVIEWS project is to obtain geometric and computation benefits from the design of a specific mirror profile/sensor layout. This section shall outline how the task of visual navigation benefits from the use of our Omniviews camera design.

We propose two main navigation modalities: *Topological Navigation* and *Visual Path Following*. Topological Navigation is used for traveling long distances and does not require knowledge of the exact position of the robot but rather, a *qualitative* position on a topological map. The navigation process combines appearance based methods and visual servoing upon some environmental features in a bird's-eye view of the ground plane. Details can be found in [5].

Visual Path Following is required for local, very precise navigation, for e.g. door traversal, docking. The robot is controlled to follow a pre-specified path accurately, by tracking visual landmarks in the scene.

Our Omniviews camera was placed on a SCOUT mobile robot platform, as shown in Figure 1. An a priori set of images was captured every 40 cm, along a simple corridor environment at IST/ISR. Each image was captured with constant vertical resolution and was 110×252 pixels in size.

Figure 1: The omnidirectional camera mounted on a SCOUT mobile robot platform.

2.1 Topological navigation

A topological map is used to describe the robot's global environment. A mission could be specified as: "*go to the third office on the left-hand side of the second corridor*". Navigation is achieved by comparing current omnidirectional image to previously acquired views of the corridor (landmarks). Given that a large number of images are required to map the environment, the a priori dataset is reduced to a low-dimensional eigenspace by the use of Principal Component Analysis.

Figure 2 (left) shows an unknown image acquired at runtime and Figure 2 (right) its estimated topological position. Topological Navigation can also be undertaken using the most discriminatory information from the a priori dataset, using a method we developed, termed *Information Sampling.* In this way, the amount of information needed to map the environment, is significantly reduced. For more details on the information sampling method, see [2] and for its application to navigation, see [3].

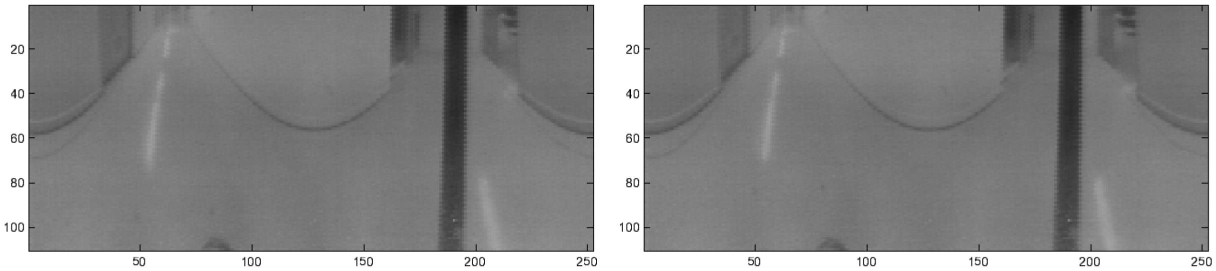


Figure 2: Left: An unknown image captured at runtime. Right: Its closest match.

We note here that in order to successfully navigate along the topological graph, we still have to define a suitable vision-based behavior for corridor following (*links* in the map). In different environments, one can always use simple knowledge about the scene geometry to define other behaviors. We exploit the fact that most corridors have parallel guidelines to control the robot heading direction, aiming to keep the robot centered in the corridor. These guidelines are extracted from bird's-eye views of the corridor. We note here that by using the mixed mirror design, we can obtain the images needed for both topological navigation and visual path following, directly from a single mirror profile.

2.2 Visual Path Following

For local, precise navigation tasks, we rely on *Visual Path Following* [1] for e.g. door traversal, docking and navigating in cluttered environments.

We use the omnidirectional images to track environmental features, estimate the robot's position/orientation and drive the robot along a pre-specified trajectory.

As features we use edge segments (Fig.3). Edge segments are represented by 15 to 30 sampled points, that are tracked by searching the image perpendicularly to the edge segments. The search criterion is based upon the evaluation of the image gradient and the distance to the original edge position. Edge segments are obtained through a robust fitting procedure and the new corner points are determined by their intersection.

Currently, the user initializes the relevant features to track. To detect the loss of tracking during operation, the process is continuously self-evaluated by the robot, based on gradient intensities obtained within specified areas around the landmark edges. If these gradients decrease significantly compared to those expected, a recovery mechanism is launched.

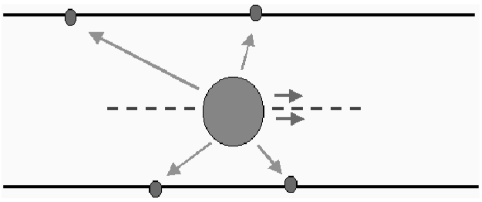


Figure 3: Visual Path Following: the robot follows a previously specified path, the dashed line, relying on the visual tracking of the vertical segments that are represented in the image by the small filled circles.

3 Teleoperation and 3D reconstruction

In order to be able to teleoperate the mobile robot, an intuitive user interface is in the process of being developed. The main idea behind this approach is to present the user with a simple 3D model of the environment. The user may then pick locations, for example a certain office, from within the model thus defining the robot's destination.

We have developed an algorithm which can reconstruct environmental scene from a *single* omnidirectional image [4].

3.1 Interactive 3D Reconstruction

We now describe the method used to obtain 3D models given the following information: a perspective image (obtained from an omnidirectional camera), a camera orientation obtained from vanishing points and some limited user input [4].

Let $p = [u\ v\ 1]^T$ be the projection of a 3D point $[C\ C'\ C''\ 1]^T$ that we want to reconstruct. Then, if we consider a normalized camera, we have the following:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \lambda\,[r_1\ r_2\ r_3\ 0] \begin{bmatrix} C \\ C' \\ C'' \\ 1 \end{bmatrix} \tag{1}$$

$$= \lambda\,(Cr_1 + C'r_2 + C''r_3)$$

where r_1, r_2, r_3 are vanishing points. As is usual, we choose 0 as the origin of the coordinates for reconstruction. Next, we define lines towards vanishing points:

$$l_i = r_i \times [u\ v\ 1]^T, \quad i = 1 \ldots 3. \tag{2}$$

Then, using the cross and internal products property that $(r \times p)^T.p = 0$, we obtain:

$$l_i^T.r_1 C + l_i^T.r_2 C' + l_i^T.r_3 C'' = 0 \tag{3}$$

which is a linear system in the coordinates of the 3D point. This can be rewritten as:

$$\begin{bmatrix} 0 & l_1^T.r_2 & l_1^T.r_3 \\ l_2^T.r_1 & 0 & l_2^T.r_3 \\ l_3^T.r_1 & l_3^T.r_2 & 0 \end{bmatrix} \begin{bmatrix} C \\ C' \\ C'' \end{bmatrix} = 0. \tag{4}$$

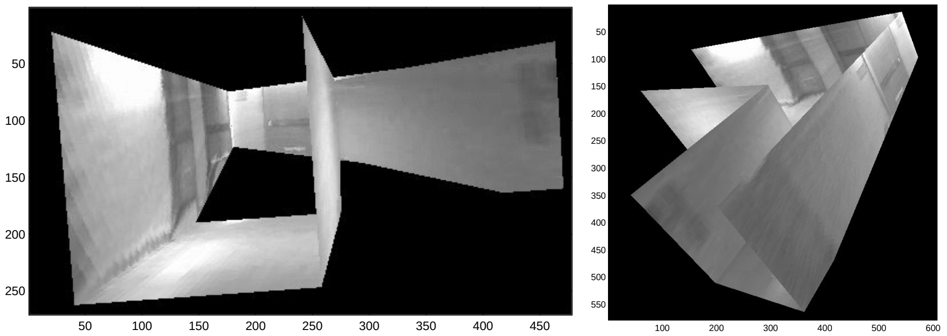


Figure 4: Views of the reconstructed 3D model.

Generalising this system to N points we again obtain a linear system:

$$A.\mathcal{C} = 0 \tag{5}$$

where $\mathcal{C}$ contains the $3N$ tridimensional coordinates that we wish to locate and A is block diagonal, where each block has the form shown in equation (4). Thus, A is of size $3N \times 3N$. Since only two equations from the set defined by equation (4) are independent, the co-rank of A is equal to the number of points N. As expected, this shows that there is an unknown scale factor for each point.

Now, adding some limited user input, in the form of co-planarity or co-linearity point constraints, a number of 3D coordinates become equal and thus the number of columns of A may be reduced. As a simple example, if we have 5 points we have 5×3 free coordinates, i.e the number of columns of A. Now, if we impose the constraint that points P_1, P_2, P_3 are co-planar, with constant z value and points P_4, P_5 are co-linear, with a constant (x, y) value, then the total number of free coordinates is reduced from the initial 15 to 11. Thus, the coordinates P_{2z}, P_{3z}, P_{5x}, P_{5y} are dropped from the linear system defined by equation (5).

Given sufficient user input, the co-rank of A becomes 1. In this case, the solution of the system will give a reconstruction with no ambiguity other than that - well known - of scale.

Figure 4 shows the 3D models obtained by the algorithm but using a standard omnidirectional camera. Current work focuses upon obtaining models using the Omniviews camera.

4 Visual Tracking

The ability to track visual features in the environment is one of the basic perceptual capabilities in both natural and robotic systems. The interaction with dynamic environments require a constant monitoring of the changing position of moving objects and the perception of self-motion is greatly enhanced by visual perception. The use of wide field–of–view visual sensors (e.g. panoramic), further improve the reaction speed of the system, since an extended part of the environment is monitored without the need for mechanical control of the sensor. Also the navigation capabilities of the system greatly benefit from the translation-rotation ambiguity reduction of omnidirectional systems.

In [6], we presented a system purposefully designed for high precision tracking of planar surfaces. With conventional cameras, the 3D rigid motion of a plane can be described by 2D transformation (homography – 8 degrees of freedom) in the image plane. The availability of software and hardware optimized routines to perform such transformations, allows real-time performance of the tracking system. However, usual omnidirectional systems distort the viewing geometry in such a way that the motion of planar surfaces can no longer be described by such simple transformations. But, with a panoramic representation, the imaging geometry recovers a quasi-cartesian description, allowing to approximate the image observed deformations by 2D homographies, with the additional advantage of the observed surfaces "rarely" leaving the field of view[1].

Thus, the use of a panoramic sensor presents the advantages of both a simple geometrical image formation model and a complete horizontal field of view.

4.1 The Algorithm

The tracking algorithm principle can be summarized in the following lines (for further details see [7]):

- Initialization
    1. Grab the first image.
    2. Select a image region to track (*template*).
    3. Warp the image region according to some likely 2D planar transformations and collect them in sets of *deformed template databases.*
    4. Initialize the *current transformation* to the identity.
- Loop
    4 Grab a new image.
    5 Warp the current image according to the *current transformation*
    6 Compute the *residual transformation* by matching the warped image with the *deformed template databases* (we use a damped least squares regression method).
    7 Update the *current transformation* by composition with the *residual transformation.*
    8 Go to 4.

4.1.1 Remarks

The following steps of the algorithm are worth noticing:

Step 2 – The choice of transformations to use in the construction of the *deformed template databases* depends on each particular application. It should match, as much as possible, the distribution of the real transformations occuring in the application.

Step 4 – The computation of the residual transformation can be made in many ways. Usually we perform a few iterations on the *deformed template databases*, starting by the databases with more constrained transformations and ending with the less constrained ones.

---

[1] if we consider land mobile robots, most of the image motion occurs in horizontal directions.

4.2 The Image Warper

One of the required computational procedures of the previous algorithm is the ability to perform fast geometrical planar transformations on the images. We use the *Intel Image Processing Library* [8] to perform the warping. This library contains optimized code with MMX instructions and performs linear or bicubic interpolation to avoid aliasing in the generated images. The computation time required to warp each full image is about 10 msec on a Pentium 400 Mhz computer, but usually the selected regions to track are much smaller.

4.3 Virtual Pan and Tilt

The *Image Warper* performs any kind of planar transformation, including translations, rotations, scale changes, etc. Restricting the transformations to simple translations (both horizontal or vertical) we can emulate the pan and tilt degrees of freedom of a virtual camera looking to a region of interest in the panorama.

4.4 Results

We present some frames of a simulated tracking sequence illustrating translations and rotations (see Fig. 5). The initial region is represented by solid lines. The small circles represent the output of the algorithm and are supposed to track as close as possible the corners and the center of the selected region as it changes along time. Notice that the algorithm copes with the typical angular wrapping of panoramic sensors and the high precision of the tracker even with significantly blurred images.

5 Conclusions

We have shown how the images obtained from the Omniviews camera can be utilized in a variety of applications. In particular we have addressed robot navigation, teleoperation/robot interfaces and visual tracking.

In tests done with the current CMOS sensor technology, we have shown that these tasks can be accomplished in spite of the limited number of pixels, and benefit from the sensor-mirror geometry. Future work will include the usage of better quality images and the application of more demanding vision algorithms or applications.

References

[1] Gaspar, José; Santos-Victor, José. “Visual Path Following with a Catadioptric Panoramic Camera,” *7th International Symposium on Intelligent Robotic Systems* - SIRS1999 - Coimbra, Portugal, July 2001.

[2] Winters, Niall; Santos-Victor, José, “Information Sampling for Optimal Image Data Selection,” *9th International Symposium on Intelligent Robotic Systems* - SIRS2001 - Toulouse, France, July 2001.

[3] Winters, Niall; Santos-Victor, José, “Visual Attention based Robot Navigation using Information Sampling,” *2001 IEEE/RSJ International Conference on Intelligent Robots and Systems* - IROS2001 - Hawaii, USA.

[4] Gaspar, José; Grossmann, Etienne and Santos-Victor, José; “Interactive Reconstruction from an Omnidirectional Image,” *9th International Symposium on Intelligent Robotic Systems* - SIRS2001 - Toulouse, France, July 2001.

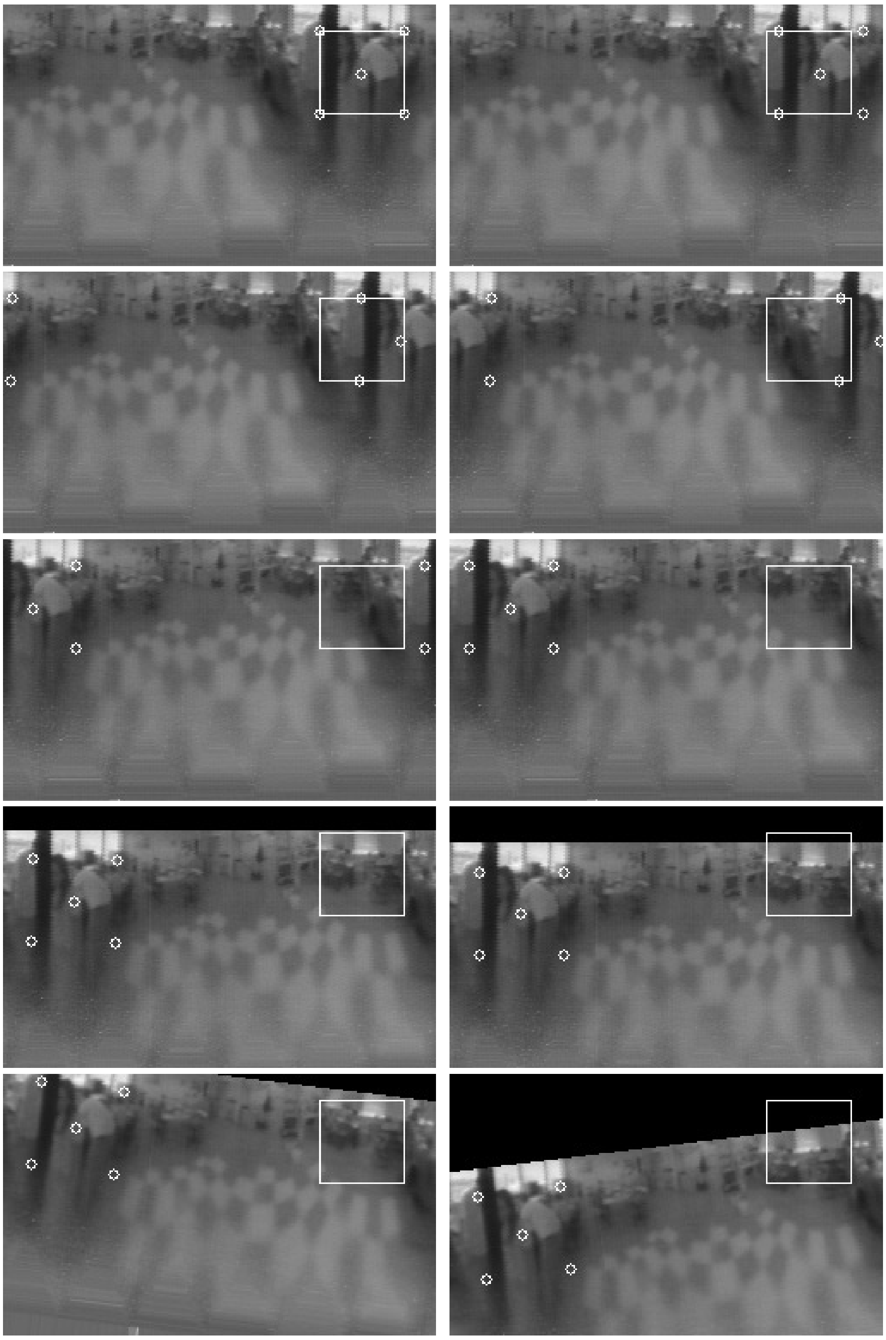


Figure 5: Tracking sequence. The solid square represents the original location of the selected region to track. The small circles correspond to the center and the corners of the estimated location of the tracked area.

[5] Gaspar, José; Winters, Niall and Santos-Victor, José. “Vision-based Navigation and Environmental Representations with an Omnidirectional Camera,” *IEEE Transaction on Robotics and Automation,* Vol 16, 6, December 2000.

[6] Sjoerd van der Zwaan, Alexandre Bernardino and José Santos-Victor. Vision Based Station Keeping and Docking for an Aerial Blimp. In *Proc. IROS'2000*, Takamatsu, Japan, October 2000.

[7] A. Bernardino, J. Santos-Victor and Giulio Sandini. Tracking planar structures with log-polar images. In *Proc. SIRS2000*, Reading, UK, July 2000.

[8] Intel Corporation. Intel image processing library. Online publication: http://developer.intel.com.

[9] Deccò, Claudia; Gaspar, José; Winters, Niall and Santos-Victor, José. “Omniviews mirror design and software tools”, Omniviews deliverable DI-2.