THE EFFICIENT E-3D VISUAL SERVOING

Geraldo Silveira [*,†,‡], Ezio Malis [†], and Patrick Rives [†]

[†] INRIA Sophia-Antipolis – Project ARobAS, 2004 Route des Lucioles, BP 93, 06902 Sophia-Antipolis Cedex, France,

FirstName.LastName@sophia.inria.fr

[‡] CenPRA Res. Center – DRVC Lab., Rod. Dom Pedro I, km 143,6, Amarais, CEP 13069-901, Campinas/SP, Brazil,

Geraldo.Silveira@cenpra.gov.br

Abstract

A vision-based control technique is proposed to automatically drive a robot to a given desired pose which has never been reached beforehand. Hence, the corresponding desired image is not available. Furthermore, since we deal with unknown scenes, standard pose reconstruction algorithms cannot be applied. To efficiently solve this problem, we represent the scene as a collection of planes. A robust detector is employed to explicitly identify planes, since they may leave the image during extended navigation tasks. These planes are then exploited by an efficient direct method for pose recovery, leading to fast and accurate estimates. The framework is validated with synthetic and real imagery.

Index Terms

Vision-based control, visual servoing, vision-based navigation, pose reconstruction, plane detection, unknown environment.

[*] Corresponding author. E-mail: Geraldo.Silveira@sophia.inria.fr

Nomenclature

γ	Control gain
$\boldsymbol{\omega}$	Rotational velocity
$\boldsymbol{\pi}$	Normal vector of a plane
σ	Singular value
θ	Angle of rotation
$\boldsymbol{\upsilon}$	Translational velocity
$\boldsymbol{\xi}$	Vector of pose coordinates
d	Euclidean distance from the center of projection to a plane
$\mathbf{e}$	Vector of control error
$\mathcal{F}$	Cartesian frame (coordinate system)
$\mathbf{G}$	Projective homography
$\mathbf{H}$	Euclidean homography
$\mathcal{H}$	Convex hull
$\mathcal{I}$	The image space
$\mathbf{K}$	Intrinsic camera parameters
$\mathbf{m}$	Homogeneous normalized pixel coordinates
$\mathbf{n}$	Unit normal vector of a plane
$\mathcal{O}$	Center of projection
$\mathbf{P}$	Homogeneous coordinates of a 3D point
$\mathbf{p}$	Homogeneous pixel coordinates
$\mathbf{R}$	Rotation matrix
$\mathbf{T}$	Change-of-frames homogeneous matrix for coordinate transformation
$\mathbf{t}$	Translation vector
$\mathcal{T}$	Reference template (image region)

$\mathbf{u}$	Unit axis of rotation
$\mathbf{v}$	Camera velocity (control input)
$\mathbf{W}$	Change-of-frames homogeneous matrix for velocity transformation
$\mathbf{w}$	Warping function

1. Introduction

How to adequately exploit visual information for controlling dynamic systems in closed loop has been widely investigated during the last two decades. Indeed, various vision-based controllers are readily available in the literature (Chaumette and Hutchinson 2006). In all cases however, the control objective of visual-servoing systems consists in driving the robot to a desired (reference) pose by using appropriate visual information. The vast majority of techniques to date focuses on the teach-by-showing approach, e.g. the 2D visual servoing technique presented in (Weiss and Anderson 1987). In this case the reference signals are relative to the given reference image, which is captured by placing the robot at the desired pose. Those systems are generally designed such that the initial pose is considered to be in a neighborhood of the desired one. Recently, a relevant research topic within the vision-based control community has concerned with visual servoing under large initial displacements. These studies have focused on how to maintain in the field-of-view a sufficient amount of the same information found in the reference image. Some techniques e.g. (Mezouar and Chaumette 2002), (Silveira and Malis 2007a) endeavor to plan a suitable image path so as to respect the visibility constraints.

1.1. Control Objectives

The present work[1] focuses on automatically driving a camera-mounted robot to a given desired Cartesian pose relatively to a given reference frame. Since everything is relative, the reference frame is also defined by the user. That is, the desired pose can be specified relatively to a particular camera

[1] This article was presented in part at the IEEE International Conference on Robotics and Automation, Orlando, FL, 2006; and in part at the IEEE/RSJ International Conference on Intelligent Robots and Systems, China, 2006.

frame (e.g. the first frame), or even to a particular known object by attaching a frame to this latter. For example, the robot may be commanded to visually move in a particular direction with respect to its current pose. Hence, standard 3D visual servoing strategies e.g. (Wilson et al. 1996), (Thuilot et al. 2002) fall into this class of methods. However, these latter strategies require the prior knowledge of the object's metric model.

Very importantly, this work deals with unknown scenes/objects. In this case, standard model-based techniques for pose recovery cannot be applied. Furthermore, we consider navigation (or positioning) tasks where the given desired pose has never been reached by the robot beforehand. Thus, the corresponding desired image is neither available nor can be rendered. This fact makes impossible to use 'metric model'-free visual servoing techniques which are based on the teach-by-showing approach. For example, it is not possible to use either the technique proposed in (Basri et al. 1999), where the Essential matrix that links the current and desired images is exploited by assuming non-planar scenes, or the general technique proposed in (Silveira and Malis 2007a), where neither scene assumptions nor decompositions are performed.

On one hand, if no other sensory device than a single camera is used, the translational part of the task is defined up to a scale factor (Rives 2000). That is, only the specified direction of translation is ensured to be tracked with high accuracy. The fact of being under controlled motion provides only with an estimate of the actual amount of translation performed by the robot. The accuracy of the attained amount of translation is clearly dependent on the quality of this estimate. On the other hand, the desired orientation can be fully specified and tracked with high accuracy in all cases.

1.2. Overview of the Method

In order to efficiently perform our visual servoing task, an important issue concerns the modeling of the scene. Although higher-order approximations could be adopted, we exploit the well-known fact that representing the scene as composed of planes the estimation algorithms are improved in terms of accuracy, stability, and rate of convergence (Szeliski and Torr 1998). For these reasons, the unknown (and possibly large-scale) scene is modeled in this work as a collection of planar regions. The number of planes to be considered in the algorithm can be viewed as a compromise between accuracy and computational load. Given that our scheme can deal with large-scale scenes, the planes may leave the field-of-view as the robot moves toward its (possibly very) distant goal. Therefore, visibility constraints do not apply at all to our framework. In fact, the (unavailable) corresponding desired image may not have anything in common with the initial one, but the desired Cartesian path can still be tracked precisely.

Specifically, the proposed efficient E-3D visual servoing approach[2] (see Figure 1) mainly relies on two key techniques: on a novel approach to optimally identify multiple new planes in the image as the robot moves, so that the known planes may leave the field-of-view; and, by exploiting these planes, on a direct method for pose reconstruction. Once the optimal current camera pose is recovered, our control objective can be pursued. While scene planarity especially favors computational efficiency, the latter direct method contributes to achieve high levels of accuracy.

Direct methods (Irani and Anandan 1999) exploit all pixel intensities so as to recover the desired information, differently to feature-based methods. This latter requires two intermediate steps. First, a sufficiently large set of features is extracted. Afterward, correspondences are established based on descriptors together with a robust matching procedure. Although feature-based methods may afford larger motions of the object in the image, it inevitably introduces errors which are never corrected. Since we consider real-time vision-based control, we can suppose that the frame rate is sufficiently high so as to observe small displacements of the object in the field-of-view. Furthermore, the robustness to illumination changes is somewhat limited within feature extraction and matching procedures. On the other hand, the robustness to arbitrary lighting variations can be effectively incorporated within direct methods (Silveira and Malis 2007b). Therefore, by using all possible image information and avoiding the difficulties of feature-based methods, the accuracy of direct pose reconstruction proce-

[2] E-3D is an acronym for Extended-3D.

dures is significantly improved. Results from vision-based navigation tasks are shown to confirm these statements.

1.3. Other Related Works

Given that no other sensory device than a single camera is used, the control problem at hand is closely related to an active monocular SLAM problem.[3] Although the mapping does not necessarily have to be reconstructed to find the pose (by using an appropriate tensor e.g. Essential matrix), precision may be rapidly lost within monocular frameworks if they are not simultaneously performed. This happens because important structural constraints, e.g. scene rigidity, are not effectively exploited in a long run. As a remark, the use of multiple cameras for pose recovery e.g. binocular (Comport et al. 2007) or trinocular systems (Saeedi et al. 2006) represents a different type of problem, as far as the baselines are sufficiently large with respect to the scene depths. In this case and under this baseline condition, visual odometry can indeed be sufficiently accurate despite not explicitly recovering the structure. It constitutes a different type of problem due to this important prior knowledge concerning the baselines.

Nevertheless, the proposed approach is also different from existing monocular SLAM techniques. Firstly, the vast majority of existing methods do not control the robot. For example, the scheme proposed in (Molton et al. 2004), besides not controlling the camera, assumes that small image patches are observations of planar regions. In addition, the normal vector of these patches is initially assigned to a "best guess" orientation. Here, we explicitly use the Planar Region Detector proposed in (Silveira et al. 2006a), which is robust to large camera calibration errors. Furthermore, the normal vector is determined by a closed-form solution (Silveira et al. 2006b), which is presented in this article. The necessary and sufficient conditions to allow for identifying new planes that enter the image are also provided. Experimental results in different scenarios demonstrate the robustness characteristics of the method.

[3] SLAM is an acronym for Simultaneous Localization And Mapping.

1.4. Paper Organization

The remainder of this work is arranged as follows. Section II reviews some basic theoretical aspects, as well as it introduces the proposed visual servoing scheme. The vision aspects involved in the strategy is presented in the Section III, while the control aspects are developed in Section IV. The results are then shown and discussed in the Section V. Finally, the conclusions are presented in the Section VI, and some references are given for further details.

2. Modeling

2.1. Notations

Throughout the article, otherwise explicitly stated, scalars are denoted either in italics or lowercase Greek letters, e.g. v or λ, vectors in lowercase bold fonts, e.g. $\mathbf{v}$, whereas matrices are represented in uppercase bold fonts, e.g. $\mathbf{V}$. Groups are written in uppercase double-struck (i.e. blackboard bold) fonts, e.g. the n-dimensional group of real numbers $\mathbb{R}^n$, whereas $\{\mathbf{v}_i\}_{i=1}^{n}$ corresponds to the set $\{v_1, v_2, \ldots, v_n\}$. Besides, $\left(\mathbf{V}^{-1}\right)^\top = \left(\mathbf{V}^\top\right)^{-1}$ is abbreviated by $\mathbf{V}^{-\top}$ and $\mathbf{0}$ denotes a matrix of zeros of appropriate dimensions. We also follow the standard notations $\widehat{\mathbf{v}}$, $\mathbf{v}^\top$ and $\|\mathbf{v}\|$ to respectively represent an estimate, the transpose and the Euclidean norm of a variable $\mathbf{v}$.

Let $\mathcal{F}$ be the camera frame whose origin $\mathcal{O}$ coincides with its center of projection. Suppose that $\mathcal{F}$ is displaced with respect to a second frame $\mathcal{F}'$ in the Euclidean space by $\mathbf{R} \in \mathbb{SO}(3)$ and $\mathbf{t} = [\, t_x \;\, t_y \;\, t_z \,]^\top \in \mathbb{R}^3$, respectively the rotation matrix and the translation vector. Consider the angle-axis representation of the rotation matrix. By using the matrix exponential $\mathbf{R} = \exp([\mathbf{r}]_\times)$, where $\mathbf{r} = \mathbf{u}\theta$ is the vector containing the angle of rotation θ, and the axis of rotation $\mathbf{u} \in \mathbb{R}^3 : \|\mathbf{u}\| = 1$. The notation $[\mathbf{r}]_\times$ represents the skew symmetric matrix associated to vector $\mathbf{r}$. Hence, the camera pose can be defined by a 6-vector $\boldsymbol{\xi} = \left[\, \mathbf{t}^\top \;\, \mathbf{r}^\top \,\right]^\top$, containing the global coordinates of an open subset of $\mathbb{R}^3 \times \mathbb{SO}(3)$.

2.2. Camera Model

Consider the pinhole camera model. In this case, a 3D point with homogeneous coordinates $\mathbf{P}_i = \left[\begin{array}{cccc} X_i & Y_i & Z_i & 1 \end{array}\right]^\top$ defined with respect to frame $\mathcal{F}$ is projected onto the image space $\mathcal{I} \subset \mathbb{R}^2$ as a point with pixel homogeneous coordinates $\mathbf{p}_i \in \mathbb{P}^2$ through

$$\mathbf{p}_i = \left[\begin{array}{ccc} u_i & v_i & 1 \end{array}\right]^\top \propto \mathbf{K} \left[\begin{array}{cc} \mathbf{I}_3 & \mathbf{0} \end{array}\right] \mathbf{P}_i, \tag{1}$$

where $\mathbf{K} \in \mathbb{R}^{3\times 3}$ is an upper triangular matrix that gathers the camera intrinsic parameters

$$\mathbf{K} = \begin{bmatrix} \alpha_u & s & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

with focal lengths $\alpha_u, \alpha_v > 0$ in pixel dimensions, principal point $\mathbf{p}_0 = \left[\begin{array}{ccc} u_0 & v_0 & 1 \end{array}\right]^\top$ in pixels, and skew s. Correspondingly, the same point $\mathbf{P}_i \in \mathbb{P}^3$ is projected onto the image space $\mathcal{I}' \subset \mathbb{R}^2$ associated to $\mathcal{F}'$ as

$$\mathbf{p}'_i = \left[\begin{array}{ccc} u'_i & v'_i & 1 \end{array}\right]^\top \propto \mathbf{K} \left[\begin{array}{cc} \mathbf{R} & \mathbf{t} \end{array}\right] \mathbf{P}_i, \tag{3}$$

under the assumption that $\mathbf{K}' = \mathbf{K}$. Then, from the general rigid-body equation of motion together with Eqs. (1) and (3), it is possible to obtain the geometric relation that links the projection of $\mathbf{P}_i$ onto both images:

$$\mathbf{p}'_i \propto \mathbf{K}\,\mathbf{R}\,\mathbf{K}^{-1}\mathbf{p}_i + \frac{1}{Z_i}\mathbf{K}\,\mathbf{t}. \tag{4}$$

2.3. Plane-based Two-view Geometry

Consider the normal vector description of a plane $\boldsymbol{\pi} = \left[\begin{array}{cc} \mathbf{n}^\top & -d \end{array}\right]^\top \in \mathbb{R}^4 : \|\mathbf{n}\| = 1, d > 0$. Let $\boldsymbol{\pi}$ be defined with respect to frame $\mathcal{F}$. If a 3D point $\mathbf{P}_i$, with inhomogeneous coordinates $\underline{\mathbf{P}}_i$, lies on such planar surface then

$$\mathbf{n}^\top \underline{\mathbf{P}}_i = \mathbf{n}^\top Z_i\,\mathbf{K}^{-1}\mathbf{p}_i = d, \tag{5}$$

and hence

$$\frac{1}{Z_i} = \frac{\mathbf{n}^\top \mathbf{K}^{-1} \mathbf{p}_i}{d}. \tag{6}$$

By injecting Eq. (6) into Eq. (4), a projective mapping $\mathbf{G} : \mathbb{P}^2 \mapsto \mathbb{P}^2$ (also referred to as the projective homography) defined up to a non-zero scale factor is obtained:

$$\mathbf{p}_i' \propto \mathbf{G}\, \mathbf{p}_i \tag{7}$$

with

$$\mathbf{G} \propto \mathbf{K} \left(\mathbf{R} + d^{-1}\, \mathbf{t}\, \mathbf{n}^\top \right) \mathbf{K}^{-1}. \tag{8}$$

A warping operator $\mathbf{w} : \mathbb{P}^2 \mapsto \mathbb{P}^2$ can thus be defined:

$$\mathbf{p}_i \mapsto \mathbf{p}_i' = \mathbf{w}(\mathbf{G}, \mathbf{p}_i) \tag{9}$$

$$= \begin{bmatrix} \dfrac{g_{11}u + g_{12}v + g_{13}}{g_{31}u + g_{32}v + g_{33}} \\ \dfrac{g_{21}u + g_{22}v + g_{23}}{g_{31}u + g_{32}v + g_{33}} \\ 1 \end{bmatrix} \tag{10}$$

where $\{g_{ij}\}$ denotes the elements of the matrix $\mathbf{G}$. It can be noticed that $\mathbf{G}$ encompasses an Euclidean homography $\mathbf{H} \in \mathbb{R}^{3\times 3}$ for the case of internally calibrated cameras. That is, using normalized homogeneous coordinates

$$\mathbf{m}_i' = \mathbf{K}^{-1}\, \mathbf{p}_i' \qquad \text{and} \qquad \mathbf{m}_i = \mathbf{K}^{-1}\, \mathbf{p}_i, \tag{11}$$

and multiplying Eq. (7) by $\mathbf{K}^{-1}$ yields

$$\mathbf{m}_i' \propto \mathbf{H}\, \mathbf{m}_i \tag{12}$$

with

$$\mathbf{H} \propto \mathbf{R} + d^{-1}\, \mathbf{t}\, \mathbf{n}^\top. \tag{13}$$

Remark 2.1: We can observe that the same relations are obtained for corresponding points, independently whether the object is planar or not, if the camera undergoes a pure rotation motion (i.e. $\mathbf{t} = \mathbf{0}$). In this particular case, the structure of the object cannot be recovered.

2.4. Navigation Formulation

Visual servoing systems are usually designed such that the desired frame $\mathcal{F}^*$ to be attained by the camera is aligned with the absolute frame $\mathcal{F}_w = \mathcal{F}^*$. In this case, the aim is to promote adequate motions such that $\mathcal{F} \longrightarrow \mathcal{F}_w$. On effect, this leads then to set $\boldsymbol{\xi}^* = \mathbf{0}$ and the control objective to drive $\boldsymbol{\xi} \longrightarrow \mathbf{0}$ as $t \longrightarrow \infty$. However, the purpose of this work (see Figure 1) is to visually servo the robotic platform from a starting pose to an user-specified one, both with respect to a given absolute frame. For example, the absolute frame can be set to coincide with the initial frame i.e. $\mathcal{F}_w = \mathcal{F}_0$. Thus $\boldsymbol{\xi}_0 = \mathbf{0}$, and the control objective can be specified as

$$\boldsymbol{\xi} \longrightarrow \boldsymbol{\xi}^* \qquad \text{as} \qquad t \longrightarrow \infty. \tag{14}$$

In fact, after specifying the navigation task, a change of coordinate system back to the usual one can obviously be made.

In this work, we exploit the well-known fact that the representation of the scene as a collection of planar regions allows for implementing much more stable and accurate pose reconstruction algorithms (Szeliski and Torr 1998). Indeed, provided $\mathbf{K}$ and a set of planes $\{\boldsymbol{\pi}\}$, the control objective in Eq. (14) can be perfectly achieved by regulating a Cartesian-based error function $\mathbf{e}$ constructed from images. The control aspects are further discussed in Section IV. An overview of the proposed method to perform vision-based control tasks over (possibly large-scale) unknown scenes is presented in Algorithm 1, for some sufficiently small $\epsilon > 0$. With regard to the initialization (Line 1 of this algorithm), it can be performed in several ways. For instance, by providing a coarse metric estimate of a plane. In this case, the decomposition of the Euclidean homography will provide the required $\boldsymbol{\pi}_0$.

Algorithm 1. The efficient E-3D visual servoing.

```
1: define plane $\boldsymbol{\pi}_0$ in the first image $\mathcal{I}_0$
2: while $\|\mathbf{e}\| > \epsilon$ do
3:   apply control law
4:   track known planes by simultaneously recovering pose
5:   if conditions in the Proposition 3.1 are verified then
6:     identify new planes that have entered the field-of-view
7:   end if
8: end while
```

If there exist more than one plane in different configurations within a rigid scene, one may enforce this rigidity constraint to obtain $\boldsymbol{\pi}_0$ without requiring any coarse estimate a priori. If no other sensory device than a single camera is used, then the desired translation is defined up to a scale factor. The procedures stated from Line 4 to 6 of the algorithm are detailed in the next section.

3. Planes Detection and Tracking

3.1. Pose Reconstruction from Multiple Planes

This subsection presents our efficient direct method for determining the pose of the camera with respect to a given reference frame. Consider that a set of planar objects $\{\boldsymbol{\pi}_j\}_{j=1}^{n}$ has been determined. Along with this metric model, the corresponding reference template as well as the camera pose from where they were first viewed are all stored in memory. How to detect these planar regions in the image (i.e. to obtain the reference template) and to obtain their metric model will be described in the next subsections.

We formulate this important subtask of pose recovery as an optimization problem. It consists in seeking the motion parameters that best align multiple reference templates $\{\mathcal{T}_j\}_{j=1}^{n}$ to the current image $\mathcal{I}'$ such that each pixel intensity is matched as closely as possible:

$$\{\mathbf{R},\mathbf{t}\} = \underset{\substack{\mathbf{R}\,\in\,\mathbb{SO}(3)\\ \mathbf{t}\,\in\,\mathbb{R}^3}}{\arg\min}\ \frac{1}{2}\sum_{j=1}^{n}\sum_{\mathbf{p}_i}\Big[\mathcal{I}'\Big(\mathbf{w}\big(\mathbf{G}_j(\mathbf{K},\mathbf{R}_j,\mathbf{t}_j,\boldsymbol{\pi}_j),\mathbf{p}_i\big)\Big) - \mathcal{T}_j(\mathbf{p}_i)\Big]^2, \tag{15}$$

using Eqs. (8) and (9). In Eq. (15) both $\mathcal{I}'(\mathbf{p}_i)$ and $\mathcal{T}(\mathbf{p}_i)$ denotes the intensity of the pixel $\mathbf{p}_i$, and each $\{\mathbf{R}_j,\mathbf{t}_j\}$ represents the relative displacement for a particular plane. This displacement is trivially obtained at each iteration of this alignment procedure by using the pose from where the plane was first viewed and $\{\mathbf{R}, \mathbf{t}\}$.

Therefore, we can also interpret this formulation as a model-based visual tracking problem parameterized in the $\mathbb{SO}(3) \times \mathbb{R}^3$, or simply model-based visual odometry. Using an efficient second-order approximation method (Benhimane and Malis 2006), this optimization procedure can be solved using only first-order derivatives. With this, higher convergence rate and avoidance of irrelevant local minima are both achieved. It is computationally efficient because the Hessians are never explicitly computed. Given that all planes are linked by the obtained camera motion, the rigidity of the scene is directly enforced. This enforcement, along with the fact that all possible information is exploited, significantly increase the accuracy of the pose estimates.

3.2. Detection of New Planes from Images

Since the known planes may eventually get out of the image during a long-term navigation, one must be able to continuously detect new planes that have entered the field-of-view. In this subsection, the method used to segment planar regions using a pair of images is presented. The reader is referred to (Silveira et al. 2006a) for more profound demonstrations and discussions.

The interest in finding planar regions in images is not new, and a number of different approaches is available in the literature. Many of existing methods relies on scene assumptions, e.g. presence of lines (Baillard and Zisserman 1999) or perpendicularity assumptions (Dick et al. 2000). Hence we cannot apply them since we deal with unknown scenes. Another class of existing methods endeavors to perform a preliminary step of 3D scene reconstruction e.g. (Okada et al. 2001). These methods usually require several images to converge and are in general too time-consuming to be applied to real-time systems, e.g. visual-servoing systems. In order to circumvent these shortcomings, the proposed algorithm is based on a computationally efficient voting procedure from the solution of a linear system. This linear system is derived as follows. Eq. (4) along with Eq. (6) allow for rewriting the equation that links the projection of the same 3D point onto $\mathcal{I}$ and $\mathcal{I}'$ as:

$$\mathbf{p}_i' \propto \mathbf{K}\,\mathbf{R}\,\mathbf{K}^{-1}\,\mathbf{p}_i \,+\, \mathbf{K}\,\mathbf{t}\,\mathbf{x}^\top\mathbf{p}_i, \tag{16}$$

where

$$\mathbf{x} = \mathbf{K}^{-\top}\frac{\mathbf{n}}{d}. \tag{17}$$

Pre-multiplying both members of Eq. (16) by $[\mathbf{p}_i']_\times$ and using $\mathbf{x}^\top\mathbf{p}_i = \mathbf{p}_i^\top\mathbf{x}$, the linear system is finally obtained:

$$\mathbf{A}_i\,\mathbf{x} \,=\, \mathbf{b}_i, \tag{18}$$

with

$$\begin{cases} \mathbf{A}_i = [\mathbf{p}_i']_\times\,\mathbf{K}\,\mathbf{t}\;\mathbf{p}_i^\top \\ \\ \;\mathbf{b}_i = -[\mathbf{p}_i']_\times\,\mathbf{K}\,\mathbf{R}\,\mathbf{K}^{-1}\,\mathbf{p}_i. \end{cases} \tag{19}$$

Then, triplet of corresponding interest points $\mathbf{p}_i' \leftrightarrow \mathbf{p}_i$ (e.g. provided by Harris detector together with a matching procedure) are managed in order to form linear systems whose solutions are used in a progressive Hough-like transform, and in order to respect the real-time constraints.

Voting procedures (e.g. the Hough Transform) are among the most important robust techniques in computer vision (Stewart 1999). As it will be experimentally shown in Section V, even if the set of camera parameters $\{\mathbf{K}, \mathbf{R}, \mathbf{t}\}$ are miscalibrated (i.e. only an estimate $\{\widehat{\mathbf{K}}, \widehat{\mathbf{R}}, \widehat{\mathbf{t}}\}$ is provided) and/or even if there exist mismatched corresponding points, it is still possible to cluster planar regions in the image. This robustness property is an attractive characteristic of the approach since it is able to tolerate large errors in its inputs. A major difference between the used voting technique and the standard Hough Transform is related to the performed mapping. Instead of voting the whole parameter space, the solution of the constructed linear system represents a single vote. Various advantages of this convergence mapping are discussed in (Silveira et al. 2006a), e.g. reduction of memory and computational complexities. Moreover, the strategy behind the progressive scheme is to avoid voting all possible combinations of three points. Thus, it contributes to further reduce the computational complexity since a plane is clustered as soon as the contents of the accumulator permits such a decision.

A plane (i.e. a template $\mathcal{T}$) is finally formed by means of the convex hull $\mathcal{H}$ of all clustered points:

$$\mathcal{H} \equiv \left\{ \sum_i \mu_i \, \mathbf{p}_i \, : \, \mu_i \geq 0, \; \forall i, \; \text{and} \; \sum_i \mu_i = 1 \right\}. \tag{20}$$

As we will show next, besides the explicit partitioning of planar regions, there is no "best guess" initialization regarding the normal vector of the planes, as in (Molton et al. 2004). This latter work assumes that small image patches are observations of planar regions and whose vector, after such an initialization, is refined based on a gradient descent technique. In the next subsection, a closed-form solution is presented to determine the parameters of the newly segmented planes.

3.3. Euclidean Characterization of the New Planes

To this point, a set of n new planes has been robustly partitioned in the image in terms of templates $\{\mathcal{T}_j\}_{j=1}^{n}$ (see Subsection III-B). Moreover, the relative pose $\{\mathbf{R}_j, \mathbf{t}_j\}$ between current $\mathcal{F}'$ and where they were first viewed is also provided by the running pose reconstruction algorithm (see Subsection III-A). In order to include the newly detected planes in this algorithm, we need to determine the Euclidean parameters $\boldsymbol{\pi}_j = \left[\begin{array}{cc} \mathbf{n}_j^\top & -d_j \end{array} \right]^\top \in \mathbb{R}^4$ for each plane, $j = 1, 2, \ldots, n$.

To this end, manipulating Eqs. (7) and (12) one obtains

$$\mathbf{H} = \alpha \, \mathbf{K}^{-1} \, \mathbf{G} \, \mathbf{K}, \tag{21}$$

where $\alpha \in \mathbb{R}$ represents a normalizing factor, and hence the following expression for the j-th plane:

$$\mathbf{t}_j \, \overline{\mathbf{n}}_j^\top = \alpha_j \, \mathbf{K}^{-1} \, \mathbf{G}_j \, \mathbf{K} - \mathbf{R}_j, \tag{22}$$

with

$$\overline{\mathbf{n}}_j = \frac{\mathbf{n}_j}{d_j} = \mathbf{n}_j \, \|\overline{\mathbf{n}}_j\|. \tag{23}$$

Pre-multiplying both members of Eq. (22) by $\mathbf{t}_j^\top$, a closed-form solution is obtained for determining the normal vector relative to where they were first viewed:

$$\overline{\mathbf{n}}_j = \big(\alpha_j\, \mathbf{K}^{-1}\, \mathbf{G}_j\, \mathbf{K} - \mathbf{R}_j\big)^\top \frac{\mathbf{t}_j}{\|\mathbf{t}_j\|^2}. \tag{24}$$

In order to determine $\boldsymbol{\pi}_j$ using Eq. (24), we need to obtain the normalizing α_j and the projective homography $\mathbf{G}_j$. The former is obtained as follows. Given that $\mathrm{svd}(\mathbf{H}) = [\ \sigma_1\ \ \sigma_2\ \ \sigma_3\ ]^\top$ are the singular values of $\mathbf{H}$ in decreasing order, $\sigma_1 \geq \sigma_2 \geq \sigma_3 > 0$, and that such an homography can be normalized by the median singular value (Faugeras and Lustman 1988), it is possible to use the facts that $x = \mathrm{sgn}(x)\,|x|$, $\forall x \in \mathbb{R}$, $\det(\mathbf{H}) = \prod_{k=1}^{3} \lambda_k(\mathbf{H})$, and that σ_k are the square-roots of $\lambda(\mathbf{H}^\top\mathbf{H})$, so as to define

$$\alpha_j = \frac{\mathrm{sgn}\big(\det(\mathbf{H}_j)\big)}{\sigma_2(\mathbf{H}_j)}, \tag{25}$$

where $\mathrm{sgn}(\cdot)$ denotes the signum function. In regard to the needed projective homography $\mathbf{G}_j$ to compute Eq. (24), it can be optimally found as:

$$\mathbf{G}_j = \arg\min\ \frac{1}{2} \sum_{\mathbf{p}_i} \Big[\mathcal{I}'\big(\mathbf{w}(\mathbf{G}_j, \mathbf{p}_i)\big) - \mathcal{T}_j(\mathbf{p}_i)\Big]^2 \tag{26}$$

using Eqs. (7) and (9). This non-linear direct image alignment task can be initialized by a linear method involving all corresponding features $\mathbf{p}'_i \leftrightarrow \mathbf{p}_i$ inside the convex hull.

Proposition 3.1 (Normal Vector Determination): The necessary and sufficient geometric conditions for the normal vector determination expressed in Eq. (24) are such that:

- $\|\mathbf{t}_j\| > 0$;
- $|\det(\mathbf{G}_j)| > 0$.

Proof: The proof of the Proposition 3.1 comes directly from Eq. (24), together with the knowledge that $\mathbf{K} > 0$ and $\mathbf{R}_j \in \mathbb{SO}(3)$. Its first condition states that a sufficient amount of translation relative to the distance of the detected plane has to be carried out. Otherwise, as stated in Remark 2.1, its structure cannot be recovered. The last condition comes from the fact that $\alpha_j \neq 0$ also to avoid the trivial solution. From Eq. (25), given that $\sigma_k > 0$, $\forall k$, one must then have $|\det(\mathbf{H}_j)| > 0$. That is, using Eq. (21):

$$|\det(\mathbf{H}_j)| > 0 \tag{27}$$

$$|\alpha_j| \, |\det(\mathbf{K}^{-1}) \det(\mathbf{G}_j) \det(\mathbf{K})| > 0 \tag{28}$$

$$|\det(\mathbf{G}_j)| > 0. \tag{29}$$

This in fact may be used as a measure of degeneracy of the plane (in order to discard it, for instance) if each homography $\mathbf{G}_j$ is evaluated using Eq. (7). The plane is in a degenerate configuration when is projected in the image as a line. ■

4. Control Aspects

Consider a camera-mounted holonomic robot or an omnidirectional mobile robot. Let the control input be the velocity of the camera $\mathbf{v} = \begin{bmatrix} \boldsymbol{\upsilon}^\top & \boldsymbol{\omega}^\top \end{bmatrix}^\top \in \mathbb{R}^6$, respectively the translational and rotational velocities. As previously stated, the rigidity assumption of the scene is enforced so that the relative displacement of the camera is the same for all tracked planes. In addition, given that known planes can leave the field-of-view without destabilizing the system (since it is possible to identify new planes), the control error can be simply constructed from the knowledge of the current pose $\mathbf{T} = {}^c\mathbf{T}_0$ (see Section III-A for details on how it is recovered) and the desired ${}^0\mathbf{T}^*$. This error can obviously be expressed with respect to $\mathcal{F}^*$ to conform to the usual absolute frame:

$${}^*\mathbf{T}_c = \left({}^c\mathbf{T}_0 \, {}^0\mathbf{T}^*\right)^{-1} = \begin{bmatrix} {}^*\mathbf{R}_c & {}^*\mathbf{t}_c \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SO}(3) \times \mathbb{R}^3. \tag{30}$$

The control error vector is then defined as

$$\mathbf{e} = \left[\ \mathbf{e}_v^\top\ \ \mathbf{e}_\omega^\top\ \right]^\top = \left[\ {}^*\mathbf{t}_c^\top\ \ {}^*\mathbf{r}_c^\top\ \right]^\top \tag{31}$$

$$= \left[\ \mathbf{t}^\top\ \ \mathbf{u}^\top\theta\ \right]^\top \quad \in \mathbb{R}^6, \tag{32}$$

which, by dropping the indices from Eq. (31), respectively denotes the error in translation and in rotation with respect to the usual reference frame. We emphasize that this particular control error corresponds to a positioning task whose desired pose is specified relative to the initial robot pose ${}^0\mathbf{T}^*$. Another possible task would be, for instance, to drive the camera to a given desired pose relative to a particular known plane.

Then, the derivative of Eq. (31) yields

$$\dot{\mathbf{e}} = \mathbf{L}(\mathbf{e})\, {}^*\mathbf{v} \tag{33}$$

$$= \mathbf{L}(\mathbf{e})\, \mathbf{W}(\mathbf{e})\, \mathbf{v} \tag{34}$$

with the interaction matrix $\mathbf{L}(\mathbf{e})$ given by

$$\mathbf{L}(\mathbf{e}) = \begin{bmatrix} \mathbf{I}_3 & -[\mathbf{e}_v]_\times \\ \mathbf{0} & \mathbf{L}_\omega \end{bmatrix}. \tag{35}$$

The $\mathbf{L}_\omega$ is the interaction matrix related to the parametrization of the rotation:

$$\frac{d(\mathbf{u}\theta)}{dt} = \mathbf{L}_\omega\, \boldsymbol{\omega}. \tag{36}$$

By using the Rodrigues' formula for expressing the rotation matrix, it can be shown that

$$\mathbf{L}_\omega = \mathbf{I}_3 + \frac{\theta}{2}[\mathbf{u}]_\times + \left(1 - \frac{\mathrm{sinc}(\theta)}{\mathrm{sinc}^2(\frac{\theta}{2})}\right)[\mathbf{u}]_\times^2, \tag{37}$$

where the function $\mathrm{sinc}(\cdot)$ is the so-called sine cardinal (or sampling function) defined such that $\theta\,\mathrm{sinc}(\theta) =$

$\sin(\theta)$ and $\mathrm{sinc}(0) = 1$. Also, it can be noticed that

$$\det(\mathbf{L}_\omega) = \mathrm{sinc}^{-2}\left(\frac{\theta}{2}\right), \tag{38}$$

providing for the largest possible domain of rotations. The upper-block triangular matrix $\mathbf{W}(\mathbf{e}) \in \mathbb{R}^{6\times 6}$ in Eq. (33) represents the transformation

$$\mathbf{W}(\mathbf{e}) = \begin{bmatrix} \mathbf{I}_3 & [{}^*\mathbf{t}_c]_\times \\ \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} {}^*\mathbf{R}_c & \mathbf{0} \\ \mathbf{0} & {}^*\mathbf{R}_c \end{bmatrix} = \begin{bmatrix} {}^*\mathbf{R}_c & [{}^*\mathbf{t}_c]_\times {}^*\mathbf{R}_c \\ \mathbf{0} & {}^*\mathbf{R}_c \end{bmatrix}, \tag{39}$$

since the control input $\mathbf{v}$ is defined with respect to $\mathcal{F}_c$ whereas the control error Eq. (31) is expressed with respect to $\mathcal{F}^*$.

Concerning the control law, if an exponential decrease for the control error is imposed

$$\dot{\mathbf{e}} = -\gamma\, \mathbf{e}, \qquad \gamma > 0, \tag{40}$$

then its substitution into Eq. (33) using Eq. (31) yields

$$\mathbf{v} = -\gamma\, \mathbf{W}^{-1}(\mathbf{e})\, \mathbf{L}^{-1}(\mathbf{e})\, \mathbf{e} \tag{41}$$

$$= -\gamma \begin{bmatrix} {}^c\mathbf{R}^* & -{}^c\mathbf{R}^*\,[{}^*\mathbf{t}_c]_\times \\ \mathbf{0} & {}^c\mathbf{R}^* \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & [{}^*\mathbf{t}_c]_\times \mathbf{L}_\omega^{-1} \\ \mathbf{0} & \mathbf{L}_\omega^{-1} \end{bmatrix} \mathbf{e} \tag{42}$$

$$= -\gamma \begin{bmatrix} {}^c\mathbf{R}^* & \mathbf{0} \\ \mathbf{0} & {}^c\mathbf{R}^*\mathbf{L}_\omega^{-1} \end{bmatrix} \mathbf{e}. \tag{43}$$

Such an expression can be further simplified. Given that $[\mathbf{u}]_\times^k\, \mathbf{u} = \mathbf{0}$, $\forall k > 0$, one obtains that

$$\mathbf{L}_\omega^{-1}\, \mathbf{e}_\omega = \mathbf{e}_\omega, \qquad \forall \mathbf{e}_\omega \in \mathbb{R}^3, \tag{44}$$

since

$$\mathbf{L}_\omega^{-1} = \mathbf{I}_3 + \frac{\theta}{2}\mathrm{sinc}^2\!\left(\frac{\theta}{2}\right) [\mathbf{u}]_\times + \big(1 - \mathrm{sinc}(\theta)\big)\, [\mathbf{u}]_\times^2. \tag{45}$$

Thus, the control law Eq. (43) can be rewritten as

$$\mathbf{v} = -\gamma \begin{bmatrix} {}^{c}\mathbf{R}^{*} & \mathbf{0} \\ \mathbf{0} & {}^{c}\mathbf{R}^{*} \end{bmatrix} \mathbf{e}. \tag{46}$$

The control law Eq. (46), besides the full decoupling of translational and rotational motions (it has a block diagonal matrix), induces a straight-line path linking $\overrightarrow{\mathcal{O}\mathcal{O}^{*}}$ in Cartesian space since $\dot{\mathbf{t}} = {}^{*}\mathbf{R}_{c}\, \boldsymbol{\upsilon} = -\gamma\, {}^{*}\mathbf{R}_{c}\, {}^{c}\mathbf{R}^{*}\, \mathbf{t} = -\gamma\, \mathbf{t}$.

5. Results

The results obtained by the proposed efficient E-3D visual servoing technique are shown and discussed in this section. Concerning the subtask of planar region detection (see Subsection III-B), its input is composed of: corresponding interest points; the user-requested accuracy for considering that two triplets of points have the same normal vector; and of the camera calibration parameters. In order to illustrate the robustness characteristics of this detector, besides the unavoidable mismatched features, we used erroneous camera parameters for all tested pairs of images. Despite all these sources of noise, actual planes (according to user-requested accuracy) are detected. Representative examples are shown in Figure 2. The reader may also refer to (Silveira et al. 2006a) for other demonstrations. Obviously, the used pairs of images verify the geometric conditions (see Proposition 3.1) for segmenting real planes. In order to satisfy real-time requirements, only a part of each plane is clustered. Nevertheless, a region growing process could be used to partition a larger extent of them, e.g. by iteratively verifying if other input features (not shown in the figure for the sake of clarity) projectively fit a given plane model. After the Euclidean characterization, all detected planes (formed by the convex hull of the clustered points) can be directly exploited by the pose recovery technique, which also simultaneously tracks them during navigation.

In order to have a ground truth for the proposed vision-based control technique, a textured scene was constructed: its base is composed of four planes disposed in pyramidal form and cut by another plane on its top. Then, real images were mapped onto each one of the five plans so as to simulate realistic situations as closely as possible. With respect to the navigation task, a desired Cartesian trajectory with loop closing is specified and afterward subdivided into 10 elementary positioning tasks. The trajectory has a total displacement of approximately 3.3 m. An elementary task is said to be completed here when the translational error drops below a certain precision (it was set when $\|\mathbf{e}_\upsilon\| < 1$ mm). It is evident that the total amount of time (and hence the total number of images) needed to perform the task also depends on the chosen control gain, which is set here to $\gamma = 0.5$. The images obtained at the convergence for some of these tasks are shown in Figure 3, where the detected and exploited planes are superposed as well. Note that even though a known plane (shown e.g. in the third image of Figure 3) leaves the field-of-view, the entire navigation task is successfully performed since new planes are identified. In addition, when such a known plane reenters the image it is automatically re-detected. The true errors obtained by the pose recovery process along the entire task are shown in Figure 4, since the real ground truth is available. One can also observe that when the image loses resolution (e.g. the camera moves away from the object), the precision of the reconstruction also decreases. Nevertheless, one important result comes from performing the specified closed-loop trajectory: errors smaller than 0.1 mm and than 0.01° are obtained after the camera comes back to the same pose at the beginning (compare first and last images of Figure 3). This demonstrates the level of accuracy achieved by the framework.

Another important result from the approach concerns the reconstruction of the scene in the 3D space (up to a scale factor), which is shown in Figure 5 for different views of the scene. This demonstrates that the proposed efficient E-3D visual servoing approach can be also used as a Plane-based Structure from Controlled Motion technique, improving the stability, the accuracy and the rate of convergence of Structure From Motion methods.

6. Conclusions

This work proposes a new visual servoing approach where the desired image (corresponding to the given desired pose) is not available beforehand. In addition, we consider the case where the metric model of the scene is not known a priori. By modeling the scene as a collection of planar regions, a real-time pose reconstruction is used. As the robot moves, since the known planes may eventually get out of the field-of-view, new planes are identified and then exploited by the pose recovery algorithm. Hence, distant goals may be specified. Navigation tasks were performed and only negligible Cartesian errors were obtained. In addition, it is shown that the proposed vision-based control scheme can be used as a Plane-based Structure from Controlled Motion technique as well. Future works will be devoted to improving the accuracy of the reconstructed scene by also refining their parameters within the optimization process.

Acknowledgments

This work is also partially supported by the CAPES Foundation under grant no. 1886/03-7, and by the international agreement FAPESP-INRIA under grant no. 04/13467-5.

References

Baillard, C. and A. Zisserman. 1999. Automatic reconstruction of piecewise planar models from multiple views. *Proceedings of IEEE Computer Vision and Pattern Recognition*, 559–565.

Basri, R., E. Rivlin, and I. Shimshoni. 1999. Visual homing: surfing on the epipoles. *Int. J. of Comp. Vision 33*(2): 22–39.

Benhimane, S. and E. Malis. 2006. Integration of Euclidean constraints in template based visual tracking of piecewise-planar scenes. *Proceedings of IEEE/RSJ International Conf. on Intelligent Robots and Systems*, 1218–1223.

Chaumette, F. and S. Hutchinson. 2006. Visual servo control part I: Basic approaches. *IEEE Robotics & Automation Magazine 13*(4): 82–90.

Comport, A., E. Malis, and P. Rives. 2007. Accurate quadrifocal tracking for robust 3D visual odometry. *Proceedings of IEEE International Conf. on Robotics and Automation*, 40–45.

Dick, A., P. Torr, and R. Cipolla. 2000. Automatic 3D modelling of architecture. *Proceedings of British Machine Vision Conference*, 372–381.

Faugeras, O. and F. Lustman. 1988. Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence 2*(3): 485–508.

Irani, M. and P. Anandan. 1999. About direct methods. *Proceedings of Workshop on Vision Algorithms*, 267–277.

Mezouar, Y. and F. Chaumette. 2002. Path planning for robust image-based control. *IEEE Trans. on Rob. and Autom. 18*(4): 534–549.

Molton, N. D., A. J. Davison, and I. D. Reid. 2004. Locally planar patch features for real-time structure from motion. *Proceedings of British Machine Vision Conference*, 1–10.

Okada, K., S. Kagami, M. Inaba, and H. Inoue. 2001. Plane segment finder: Algorithm, implementation and applications. *Proceedings of IEEE International Conf. on Robotics and Automation*, 2120–2125.

Rives, P. 2000. Visual servoing based on epipolar geometry. *Proceedings of IEEE/RSJ International Conf. on Intelligent Robots and Systems*, 602–607.

Saeedi, P., P. D. Lawrence, and D. G. Lowe. 2006. Vision-based 3-D trajectory tracking for unknown environments. *IEEE Trans. on Robotics 22*(1): 119–136.

Silveira, G. and E. Malis. 2007a. Direct visual servoing with respect to rigid objects. *Proceedings of IEEE/RSJ International Conf. on Intelligent Robots and Systems*, 1963–1968.

Silveira, G. and E. Malis. 2007b. Real-time visual tracking under arbitrary illumination changes. *Proceedings of IEEE Computer Vision and Pattern Recognition*, 1–6.

Silveira, G., E. Malis, and P. Rives. 2006a. Real-time robust detection of planar regions in a pair of images. *Proceedings of IEEE/RSJ International Conf. on Intelligent Robots and Systems*, 49–54.

Silveira, G., E. Malis, and P. Rives. 2006b. Visual servoing over unknown, unstructured, large-scale scenes. *Proceedings of IEEE International Conf. on Robotics and Automation*, 4142–4147.

Stewart, C. V. 1999. Robust parameter estimation in computer vision. *SIAM Rev. 41*(3): 513–537.

Szeliski, R. and P. H. S. Torr. 1998. Geometrically constrained structure from motion: points on planes. *Proceedings of European Workshop on 3D Struct. from Mult. Images of Large-Scale Environments*, 171–186.

Thuilot, B., P. Martinet, L. Cordesses, and J. Gallice. 2002. Position based visual servoing: keeping the object in the field of vision. *Proceedings of IEEE International Conf. on Robotics and Automation*, 1624–1629.

Weiss, L. E. and A. C. Anderson. 1987. Dynamic sensor-based control of robots with visual feedback. *IEEE Journal of Robotics and Automation 3*(5): 404–417.

Wilson, W. J., C. C. W. Hulls, and G. S. Bell. 1996. Relative end-effector control using Cartesian position based visual servoing. *IEEE Trans. on Rob. and Automation 12*(5): 684–696.

List of Figures

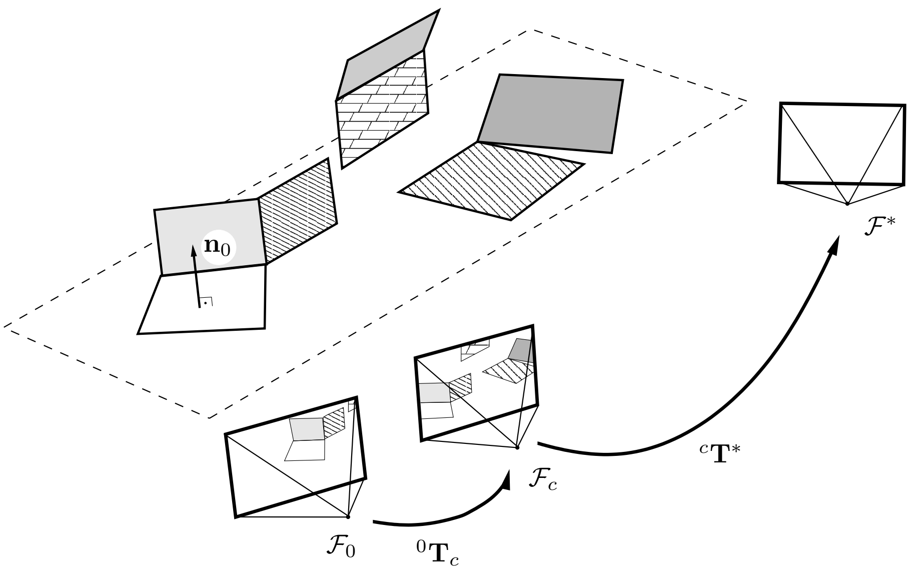
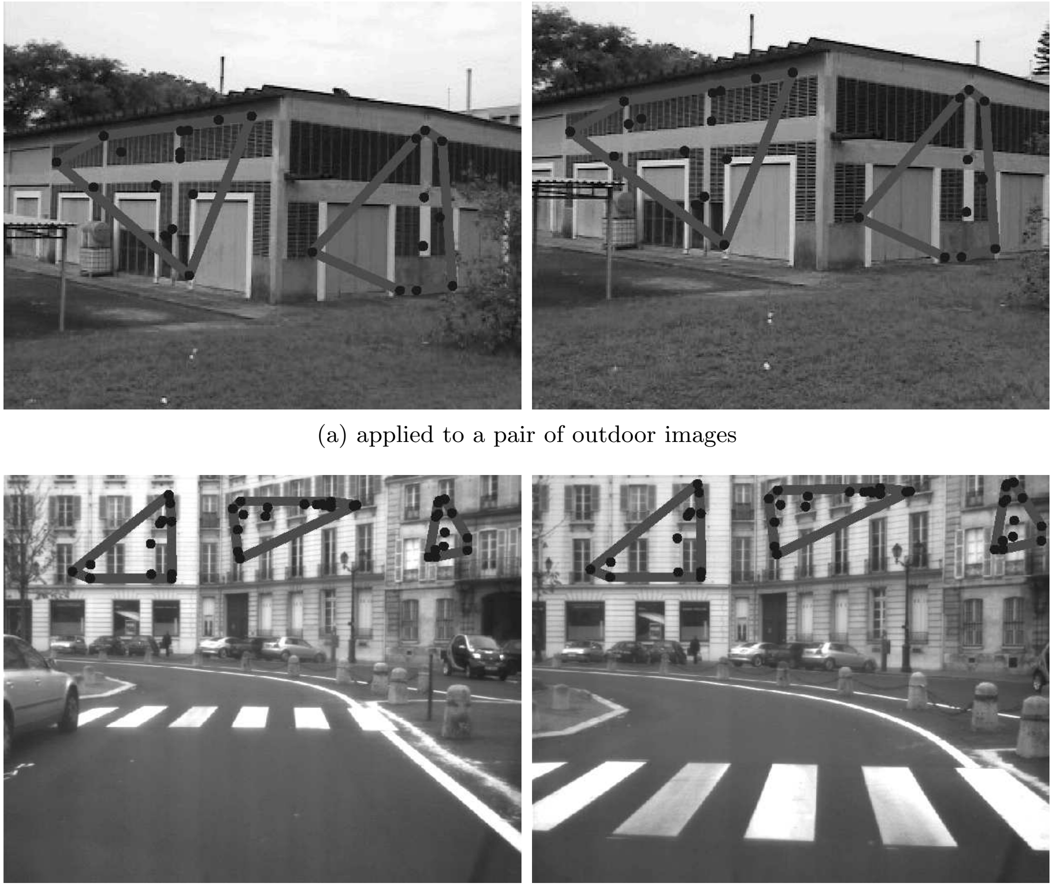


Fig. 1. Main objective of the approach: to perform a vision-based navigation task where neither the desired image (corresponding to the given desired pose) nor the metric model of the scene are available.



(a) applied to a pair of outdoor images

(b) applied to a pair of urban images

Fig. 2. Results obtained by the planar region detector. The other input features are not shown for the sake of clarity. Due to real-time requirements, only a part of each plane is segmented. A larger extent can be obtained by region growing.

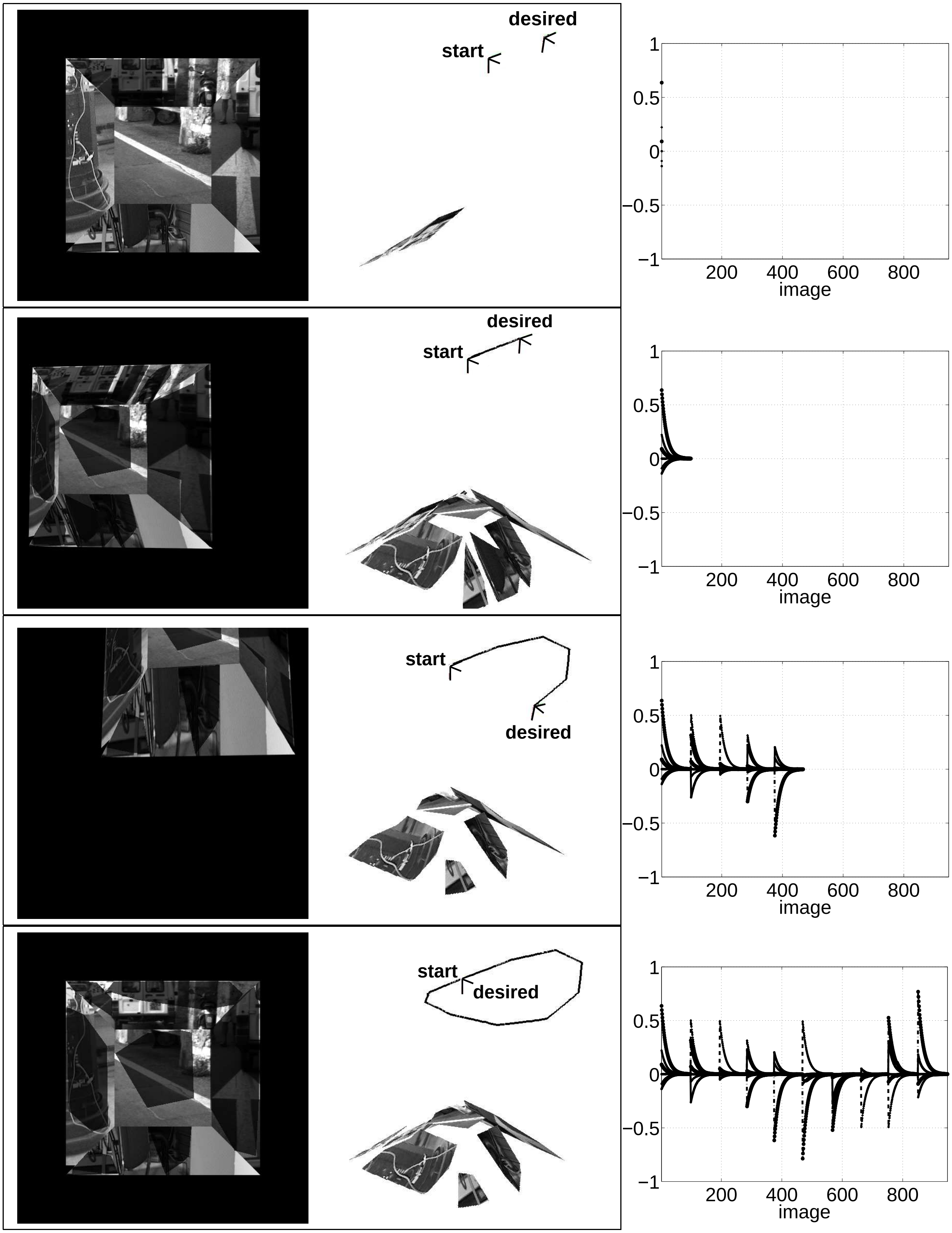
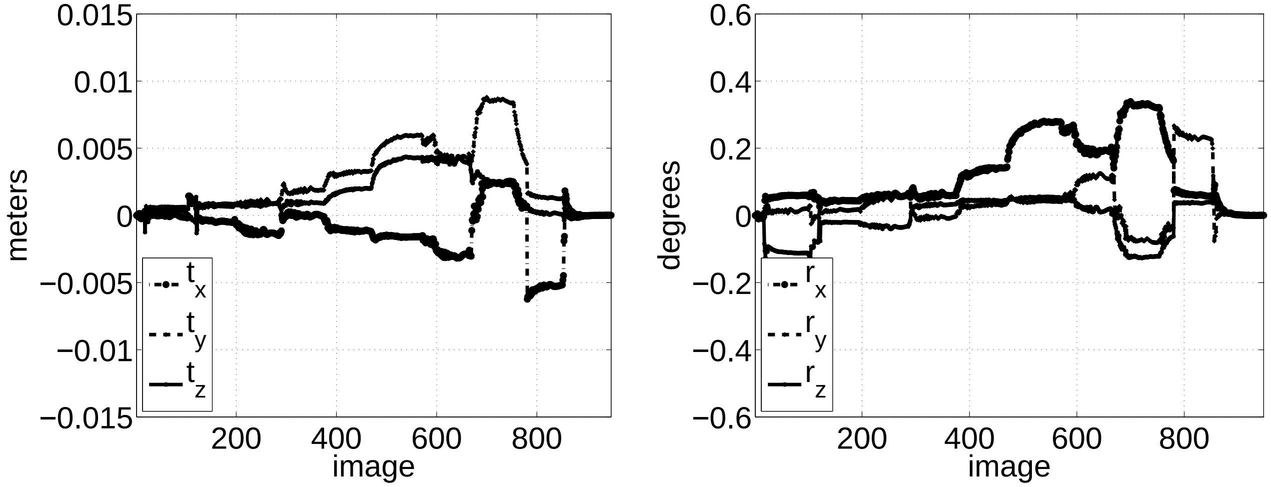


Fig. 3. A navigation task comprised of 10 elementary positioning ones with loop closing. A plane is initialized in the first image. For each elementary task shown, it is drawn respectively: the obtained image at the convergence superposed by the exploited planes, the corresponding reconstructed pose and scene, and the control input (in m/s and radians/s). Observe that a plane leaves the field-of-view (third image) but when it reenters it is again identified (fourth image).



(a) Errors in the position recovery.

(b) Errors in the attitude recovery.

Fig. 4. Errors in the pose recovery with respect to ground truth along the entire navigation task (≈3.3 m). The Euclidean norm of these errors at the end of this closed-loop trajectory (camera comes back to the same pose at the beginning) is smaller than 0.1 mm and than 0.01°, respectively for the position and orientation.

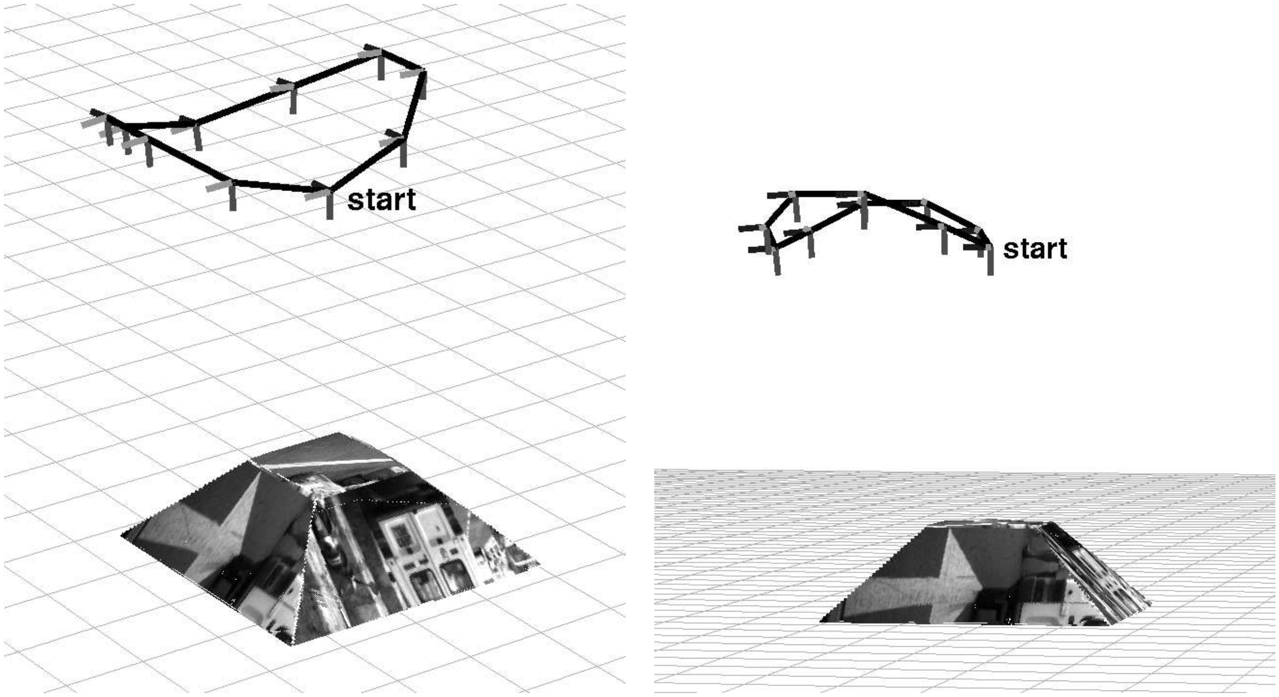


Fig. 5. The desired poses to be reached (represented by frames), the trajectory performed by the camera (line linking the frames), and the reconstructed 3D scene (after performing a region growing of the exploited planes) seen from different viewpoints.