Vision-based Guidance and Control of Robots in Projective Space

Andreas Ruf, et al

GRAVIR-IMAG, INRIA Rhône-Alpes
655, avenue de l'Europe
38330 Montbonnot St.Martin, France
Andreas.Ruf@inrialpes.fr,
http://www.inrialpes.fr/movi/people/Ruf

Abstract. In this paper, we address the problem of visually guiding and controlling a robot in projective three-space using stereo vision. As the proposed method is entirely formulated in projective space, metric models of the system have become obsolete and "easy-to-find" projective models of both, the stereo geometry and the robot's "projective kinematics", may be used.

More precisely, a given task is decomposed into its elementary parts, a translation and two rotations, based on projective constraints on its mobility- and visibility. These primitives allow trajectories to be defined which are visually and globally feasible, i.e. problems like self-occlusions, local minima, or divergent control vanish. Although robot guidance through tracking the trajectories by image-based visual servoing is now feasible, we investigate a directly computed control which combines feed-forward control with a feed-back error in each component of the task.

The method is validated and evaluted on a classical benchmark test in visual robot control – a 180^o turn – for which most image-based and stereo-based control laws fail.

1 Introduction

Research in computer vision was very much driven by the robot vision problem, but although many approaches have been proposed, visual servoing has not made the step yet from "the labs to the fabs", and scientific progress is still being made. It was actually changes in the modeling of the system that stimulated such progress.

Former position-based approaches are based on CAD-models and precise calibration of cameras and robots, so open-loop control allows to operate globally on tasks in workspace. Recent image-based approaches [6] are based on an approximate local linear model of the robot-image interaction, so closed-loop control allows to operate locally on tasks in image space. These classical approaches are essentially based on, once a geometric, once a differential, but in both cases on a metric model of a robot vision system.

Research in computer vision has made significant progress lately in modeling multi-camera systems by using algebraic projective geometry. One of the most interesting results is three-dimensional projective reconstruction [9] using stereo vision [7]: The stereo camera provides instantaneously a representation of depth and 3D structure that is independent of its metric geometry. Consequently, strong metric calibration and CAD-models are obsolete.

While vision research has focussed on additionally recovering the metric geometry of a system [4], [16], only a few researcher proposed robot vision systems that no longer rely on metric models [15], even though such models a very costly and "difficult-to-find". Curiously, there has been very little work trying to utilize the above mentioned projective stereo rig, despite its appeal as a dynamic 3D sensor [8].

In this paper, we investigate how such a projectively modeled robot vision system, which has already shown to allow for non-metric control in image-based servoing [13], can exploit the 3D capabilities of its sensor in order to overcome the most import problems of the image-based approach [3], which are: self-occlusion, local minima and divergence in presence of large rotations in the task.

Overview of the Paper and on the State-of-the-Art

Section 2 gives a minimun of preliminaries useful for reading the paper, where we refer for greater technical detail to [12]. In section 3, we devise mobility contraints on projective space in order to define several 1-dof motions – a virtual "visual mechanism" – which are used later to formulate parameterized trajectory functions. Previous work, such as [2], considers only a single camera and thus has to use Rouleaux-surfaces as constraints, i.e. a cylinder for a revolution or a prism for a translation. In contrast, we can define visual mechanisms in three-space from a minimal set of primitives, even in the projective case.

In section 4, we show for a given 6-dof reaching task how corresponding mobility and visibility constraints can be derived, and how the respective motions decompose the task into three independent visual mechanisms, i.e. three 1-dof motions. The construction relies on the robot's interaction matrix but not on position-based information. A parametric trajectory function is detailed which will later guide the reaching motion. In previous work on visual servoing, explicit trajectory generation [1] is often related to motions in camera-space but not in task-space, as desirable. Furthermore, they all rely on metric knowledge. In subsection 4.2, we devise visibility constraints for an object's face, leading to constraints among two components of the trajectory but independent of the third one. Most work on visibility uses reactive local methods to avoid image borders or obstacles. In contrast, we consider the often neglected but important problem of object self-occlusion and propose a global analytic solution leading to occlusion-free trajectories.

In section 5, we describe a directly computed control, consisting of a feed-forward part which guides the motion along a globally valid and visually feasible trajectory and a feed-back part which drives a Cartesian configuration-error to zero. Recent research shows a tendency towards integrating 3D- or pose information into the original image-based approach [6], aiming for the control-error to reflect no longer a linear image motion, but a 3D rigid motion [11]. In these approaches, the calculation of the pose-error is independent of the interaction matrix, and it relies on a (coarse) metric calibration. Although stability is rarely affected, calibration errors can degrade the system's performance, especially trajectories. In our approach, the 3D control-error and the direct control are calculated at the same time, depending only on an accurate and analytically sound interaction matrix. Moreover, the proposed non-metric solution is independent of metric parameters and works in the most general case, i.e. with an interaction matrix from robot joint-space to projective image-space.

In section 6, we present experiments based on simulation with real data. We validate the efficiency of our method in a well-known benchmark test, and evaluate its performance. The results are further discussed in section 7 to finally conclude with perspectives for future work.

Notation. Bold type $\mathbf{H}, \mathbf{T}$ is used for matrices, bold italic $\boldsymbol{A}, \boldsymbol{a}$ for vectors, and Roman a, b, θ for scalars, angles, coefficients, etc. Column vectors are written as $\boldsymbol{A}$, and row vectors as the transpose $\boldsymbol{h}^T$, where uppercase stand for spatial points, and lowercase for planes or image points.

2 Preliminaries

Stereo Vision in Projective Space. Consider a rigid pair of pinhole cameras with known epipolar geometry. A corresponding pair of projections $\mathbf{P}, \mathbf{P}'$ can be found [9], so that a pair of image points $\boldsymbol{m}, \boldsymbol{m}' \in \mathbb{P}^2$ can be reconstructed as $\boldsymbol{M} \in \mathbb{P}^3$ in a projective space spanned by 5 points, which could be imagined "rigidly linked" with the stereo camera (1). Consider further the intrinsic parameters $\mathbf{K}$ to be fixed. There exists a constant matrix $\mathbf{H}_{PE}$ relating the projective and Euclidean camera frame, and so respective points $\boldsymbol{M}$ and $\boldsymbol{N}$, up to scale ρ.

$$\begin{bmatrix} \zeta\, \boldsymbol{m} \\ \zeta' \boldsymbol{m}' \end{bmatrix} = \begin{bmatrix} \mathbf{P} \\ \mathbf{P}' \end{bmatrix}^{6\times 4} \boldsymbol{M}, \qquad \rho \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}_{\boldsymbol{N}} = \begin{bmatrix} \mathbf{K}^{-1} & \mathbf{o} \\ \boldsymbol{a}^T & 1 \end{bmatrix}^{4\times 4}_{\mathbf{H}_{PE}} \boldsymbol{M}. \tag{1}$$

The projective motion of an object consisting of 5 rigid points is defined through the 4×4 homography between them and can generally be written as a conjugate form to a rigid motion $\mathbf{T}_{RT} \in SE(3)$, i.e. a "projective displacement":

$$\mathbf{H}_{RT} = \gamma\ \mathbf{H}_{PE}^{-1}\ \mathbf{T}_{RT}\ \mathbf{H}_{PE}, \qquad \text{with } \gamma \text{ such that } \det \mathbf{H}_{RT} = 1. \tag{2}$$

This normalization preserves the hidden scale ρ in $\boldsymbol{M}$. The latter thus has a hyperplane orbit in $\mathbb{R}^4$ under action of $\mathbf{H}_{RT}$, characterized by $\rho = (\boldsymbol{a}^T 1)\boldsymbol{M}$. Thanks to this orbit, applying a projective motion to a point $\mathbf{H}_{RT}(t)\boldsymbol{M}$ results in its projective point velocity $\dot{\boldsymbol{M}}$ being well-defined up to its scale ρ. More generally, for a Euclidean plane $\boldsymbol{p}^\top = (\boldsymbol{n}^\top d)$ and point $\boldsymbol{N}$, their distance is "descaled" in the projective frame to $\phi\ \boldsymbol{n}^\top \mathbf{H}_{PE}^{-1} \mathbf{H}_{PE} \boldsymbol{N}\ \rho$. However, if $\boldsymbol{r}^\top = \boldsymbol{p}^\top \mathbf{H}_{PE}^{-1}$ and $\boldsymbol{M}$ have fix scale, $\boldsymbol{r}^T \boldsymbol{M}$ is their projective distance being well-defined up to the fix scale $\varphi\rho$.

Finally, let's briefly state the dual forms ($\perp$) of some projective representations, as they are used lateron. For a point or a line through two points, their duals are determined by their nullspace or kernel ker, being geometrically three or two planes, respectively, which intersects in the point or the line (3). For a homography $\mathbf{H}$, its dual is $\mathbf{H}^{-\top}$ acting on planes written as columns $\boldsymbol{a}$. For a projective displacement $\mathbf{H}_{RT}$ — its conjugacy to the Lie group $SE(3)$ defines by an isomorphism argument a corresponding Lie algebra, the elements of which we denote $\hat{\mathbf{H}}_{RT}$ — its Lie algebra is dualized like $-\hat{\mathbf{H}}_{RT}^\top$, a tangent operator also acting on plane vectors $\boldsymbol{a}$.

$$\begin{bmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \\ \boldsymbol{a}_3^T \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1 \end{bmatrix} = \mathbf{o}, \quad \begin{bmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 \end{bmatrix} = \mathbf{o}, \quad \mathbf{H} \perp \mathbf{H}^{-\top}, \quad \hat{\mathbf{H}}_{RT} \perp -\hat{\mathbf{H}}_{RT}^\top. \tag{3}$$

Robot Kinematics in Projective Space [12]. Consider an uncalibrated stereo rig observing a robot manipulator and continuously reconstructing some marked points on the end-effector, say five of them (Fig. 3). The effective projective motion of the end-effector is a product of the joints' projective motions: a projective rotation $\mathbf{H}_R(\theta)$ of a revolute joint (4), or a projective translation $\mathbf{H}_T(\tau)$ of a prismatic joint (5). Mathematically, these are conjugate representations of the classical one-parameter Lie groups $SO(2)$ and $\mathbb{R}^1$, where the elements of the corresponding Lie algebras will be denoted as $\hat{\mathbf{H}}_R$, or $\hat{\mathbf{H}}_T$. Conjugacy preserves this underlying algebraic structure and thus allows to manipulate the projective representations without resolving the conjugate form, which is actually by $\mathbf{H}_{PE}$ (2), containing all the metric parameters. Therefore, respective formulae for going from " algebra-to-group" (left), and from "group-to-algebra" (right) can be shown to have the closed forms:

$$\mathbf{H}_R(\theta) = \mathbf{I} + sin\theta\hat{\mathbf{H}}_R + (1 - cos\theta)\hat{\mathbf{H}}_R^2, \qquad \hat{\mathbf{H}}_R = \tfrac{1}{2\sin\theta}\,(\mathbf{H}_R - \mathbf{H}_R^{-1}), \tag{4}$$

$$\mathbf{H}_T(\tau) = \mathbf{I} + \tau\hat{\mathbf{H}}_T, \qquad \hat{\mathbf{H}}_T = \tfrac{1}{2\tau}\,(\mathbf{H}_T - \mathbf{H}_T^{-1}). \tag{5}$$

Practically speacking, this means on the one hand that an observed trial motion ($\mathbf{H}_i(q_i) \in$ group) for each single joint i defines the corresponding tangent operator ($\hat{\mathbf{H}}_i \in$ algebra), which geometrically is a twist. On the other hand, joint values readings $q_i = \theta_i$ or τ_i, in association with a projective kinematic model, which actually consists of these (six) joint operators $\hat{\mathbf{H}}_i$, allow the forward kinematics $\mathbf{H}_q(\boldsymbol{q})$ to be written as the product-of-expontial (6), where each exponential is of one of the above analytic forms: $\mathbf{H}_{Ri}(\theta_i)$ or $\mathbf{H}_{Ti}(\tau_i)$.

$$\mathbf{H}_q(\boldsymbol{q}) = \bar{\mathbf{H}}_6(\boldsymbol{q}), \quad \bar{\mathbf{H}}_i(\boldsymbol{q}) = \exp(q_1\hat{\mathbf{H}}_1)\cdots\exp(q_i\hat{\mathbf{H}}_i), \quad \partial\mathbf{H}_q/\partial q_i\big|_{\boldsymbol{q}=0} = \hat{\mathbf{H}}_i. \tag{6}$$

This kinematic model is relative to the robots "zero" $\boldsymbol{q} = 0$, for which the robot Jacobian $\mathbf{J}_H$, defined in terms is of its partial derivatives, evaluates trivially to the matrices $\hat{\mathbf{H}}_i$ (6). Further below in section 5, the general form for $\boldsymbol{q} \neq \mathbf{o}$ can be found, based on the "truncated" products $\bar{\mathbf{H}}_i$ given above(6). For the moment, let's consider an end-effector point $\boldsymbol{M}$ and derive its projective velocity $\dot{\boldsymbol{M}} = \frac{d}{dt}\mathbf{H}_q\boldsymbol{M}$, and its image velocity $\dot{\boldsymbol{s}}$, where $G(\boldsymbol{m}) = (\frac{m_1}{m_3}, \frac{m_2}{m_3})^\top$ is the perspective projection map with Jacobian $\mathbf{J}_G(\boldsymbol{m} = \mathbf{P}\boldsymbol{M})$:

$$\dot{\boldsymbol{s}} = \left[\begin{matrix} \frac{1}{m_3} & 0 & -\frac{m_1}{m_3^2} \\ \frac{1}{m_3} & 0 & -\frac{m_2}{m_3^2} \end{matrix}\right]^{2\times 3}_{\mathbf{J}_G} \mathbf{P}\dot{\boldsymbol{M}}, \qquad \dot{\boldsymbol{M}} = \left[\hat{\mathbf{H}}_1\boldsymbol{M}, \cdots, \hat{\mathbf{H}}_6\boldsymbol{M}\right]^{4\times 6}_{\mathbf{J}_H} \dot{\boldsymbol{q}}. \tag{7}$$

3 Projective Mechanisms

In this section we show how the interaction matrix $\mathbf{J}_H$ of a projective kinematic model can be "decoupled" by means of very basic constraints on the visual domain, i.e. on projective space in the most general case. The visual formulation of these constraints will allow the decoupling to reflect the geometry underlying the current task (section 4.1), to represent visibility conditions (section 4.2) and feasible trajectories (section 4.3), and to directly control joint-velocities which drive the visual mechanisms (section 5.2).

Translation along an Axis. Given two points[1] $\boldsymbol{A}_1, \boldsymbol{A}_2$ defining the translation's direction. Their dual is a pencil of planes, spanned by two planes $\boldsymbol{a}_1^\top, \boldsymbol{a}_2^\top$, with both points on its axis. The only rigid motion for which both planes have zero velocity is a translational motion along this axis. For $-\hat{\mathbf{H}}_i^\top$ being the plane-operators corresponding to the i^{th} joint, the joint-space motion $\hat{\boldsymbol{q}}_T$ generating this translation thus equals the kernel in (8), resulting in the corresponding one-dimensional group of projective translations with operator $\hat{\mathbf{H}}_T$:

$$\hat{\boldsymbol{q}}_T\ \nu = \ker\begin{bmatrix} -\hat{\mathbf{H}}_1^\top\boldsymbol{a}_1 & \ldots & -\hat{\mathbf{H}}_6^\top\boldsymbol{a}_1 \\ -\hat{\mathbf{H}}_1^\top\boldsymbol{a}_2 & \ldots & -\hat{\mathbf{H}}_6^\top\boldsymbol{a}_2 \end{bmatrix}, \qquad \hat{\mathbf{H}}_T = \hat{\boldsymbol{q}}_{T1}\hat{\mathbf{H}}_1 + \cdots + \hat{\boldsymbol{q}}_{T6}\hat{\mathbf{H}}_6. \tag{8}$$

[1] We assume for the sake of simplicity only finite points and finite planes.

Revolutions around an Axis. Given two points $\boldsymbol{A}_1, \boldsymbol{A}_2$, the only rigid motion for which both points have zero velocity is a revolution around the axis between the points. For $\hat{\mathbf{H}}_i$ being the plane-operators corresponding to the i^{th} joint, the joint-space motion $\hat{\boldsymbol{q}}_R$ generating this pure rotation writes as (9), yet not scaled to radians. The operator $\hat{\mathbf{H}}_R$ (9) of the corresponding one-dimensional projective rotation group, i.e. its eigenvalues allow to do so.

$$\hat{\boldsymbol{q}}_R \ \nu = \begin{bmatrix} \hat{\mathbf{H}}_1\boldsymbol{A}_1 & \dots & \hat{\mathbf{H}}_6\boldsymbol{A}_1 \\ \hat{\mathbf{H}}_1\boldsymbol{A}_2 & \dots & \hat{\mathbf{H}}_6\boldsymbol{A}_2 \end{bmatrix}, \qquad \hat{\mathbf{H}}_R = (\hat{\boldsymbol{q}}_{R1}\hat{\mathbf{H}}_1 + \cdots + \hat{\boldsymbol{q}}_{R6}\hat{\mathbf{H}}_6), \tag{9}$$

where ν is chosen such that $\hat{\mathbf{H}}_R$ has eigenvalues $i, -i$.

Revolution around a Point in a Plane. Another visual constraint for a revolution is as follows: Given a plane $\boldsymbol{a}_1^\top$ and a point $\boldsymbol{A}_1$, not necessarily on this plane. The only rigid motion for which both, the plane and the point, have zero velocity is a revolution around an axis perpendicular to $\boldsymbol{a}_1^\top$ and going through the point $\boldsymbol{A}_1$. For $\hat{\mathbf{H}}_i$ and $-\hat{\mathbf{H}}_i^\top$ being the point- and plane-operators, the corresponding joint-space motion $\hat{\boldsymbol{q}}_P$ and its group-operator $\hat{\mathbf{H}}_P$ are calculated and normalized as follows:

$$\hat{\boldsymbol{q}}_P \ \nu = \ker \begin{bmatrix} -\hat{\mathbf{H}}_1^\top\boldsymbol{a}_1 & \dots & -\hat{\mathbf{H}}_6^\top\boldsymbol{a}_1 \\ \hat{\mathbf{H}}_1\boldsymbol{A}_1 & \dots & \hat{\mathbf{H}}_6\boldsymbol{A}_1 \end{bmatrix}, \qquad \hat{\mathbf{H}}_P = (\hat{\boldsymbol{q}}_{P1}\hat{\mathbf{H}}_1 + \cdots + \hat{\boldsymbol{q}}_{P6}\hat{\mathbf{H}}_6). \tag{10}$$

where ν is chosen such that $\hat{\mathbf{H}}_P$ has eigenvalues $i, -i$.

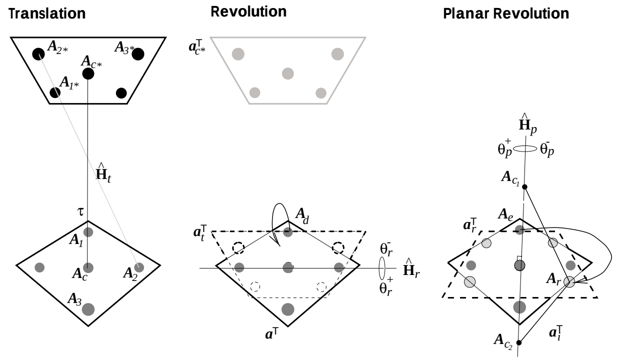

Fig. 1. Three partitions of task.

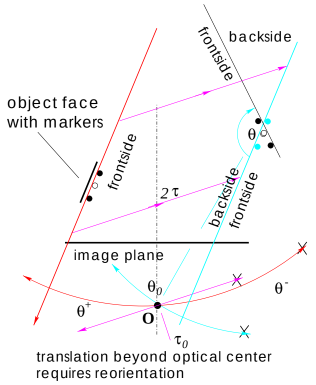

Fig. 2. Visibility.

4 Trajectories

In this section we show how a task defined for three marked points on an object-face can be partioned into well-defined components: a translation and two rotations, and how respective well-defined trajectories can be generated such that they realize the overall task while at the same time assuring the visibility of the face. Note that in the final trajectory functions (4.3), the three components of the task will be driven simultaneously. Above that, the aspect of partitioned and controlled trajectories assures global "validity" of the visual control, which will be extended in section 5 to have a feed-forward alike behaviour, as known from classical workspace control, while at the same time keeping the advantages of a feed-back control.

4.1 Partitioning

The **first component (translation of center)** is to choose one marker or the marker's midpoint as a center point $\boldsymbol{A}_c$ in order to partition the task into a pure translational motion of the center and its face, modulo a residiual rotation $\mathbf{H}_s$ of the face around the center. The respective operator $\hat{\mathbf{H}}_t$ is obtained by applying (8) to the center's current and target position, $\boldsymbol{A}_c$ and $\boldsymbol{A}_{c*}$, whereas the amplitude τ of translation is obtained by intersecting the straigh-line path of $\boldsymbol{A}_c$ with a transversally oriented plane $\boldsymbol{a}_{c*}^\top$ through $\boldsymbol{A}_{c*}$ (Fig. 1, left)

$$\boldsymbol{a}_{c*}^\top \left(\mathbf{I} + \tau\, \hat{\mathbf{H}}_t\right) \boldsymbol{A}_c = \mathbf{o}, \qquad \tau = -\frac{\boldsymbol{a}_{c*}^\top \boldsymbol{A}_c}{\boldsymbol{a}_{c*}^\top \hat{\mathbf{H}}_t \boldsymbol{A}_c}. \tag{11}$$

In order to further reason about the residual rotation, the translation of the points has to be eliminated "as if it were already done". In this sense, the effect of the first step is anticipated by applying the projective translation $\mathbf{H}_t(\tau) = \mathbf{I} + \tau\hat{\mathbf{H}}_t$ "backwards" onto the target primitives, label then with a subscripted $_t$.

$$\boldsymbol{A}_{it} = \mathbf{H}_t(-\tau)\boldsymbol{A}_{i*}, \qquad \boldsymbol{a}_t = \mathbf{H}_t^{-\top}(-\tau)\boldsymbol{a}_* \tag{12}$$

The **second component (rotation onto plane)** is to choose the line of intersection between the initial face and the translated target face as an axis of revolution in order to partition the remaining task $\mathbf{H}_s$ into a rotation $\mathbf{H}_r(\theta_r)$ of the face around this axis modulo a residual rotation $\mathbf{H}_p$ within the face plane (Fig. 1 center). The respective operator $\hat{\mathbf{H}}_r$ is obtained by applying (9) to two points dual to the planes $\boldsymbol{a}^\top$ and $\boldsymbol{a}_t^\top$, whereas the angle θ_r requires to intersect the circular path of a properly chosen point $\boldsymbol{A}_d$ with the anticipated target plane $\boldsymbol{a}_t^\top$ (13). The resulting first-order trigonometric equation in θ_r and coefficients p_0, p_s, p_c (14) has an analytic solution which is found by half-angle substitution and solving a quadratic equation (15).[2]

$$\boldsymbol{a}_t^\top \left(\mathbf{I} + \sin\theta_r \hat{\mathbf{H}}_r + (1 - cos\theta_r)\hat{\mathbf{H}}_r^2\right) \boldsymbol{A}_d = 0, \tag{13}$$

$$p_0 + p_s \sin\theta_r + p_c \cos\theta_r = 0, \tag{14}$$

$$\theta_r^+ = arctan2(\alpha(p_0, p_s, p_c), \beta(p_0, p_s, p_c)), \qquad \theta_r^- = \pi - \theta_r^+ \tag{15}$$

Again, the effects of the second step can be anticipated by applying the projective rotation $\mathbf{H}_r(-\theta_r)$ now to the beforehand translated target primitives $_t$, which are then labeled with a subscripted $_r$.

$$\boldsymbol{A}_{ir} = \mathbf{H}_r(-\theta_r)\boldsymbol{A}_{it}, \qquad \boldsymbol{a}_r = \mathbf{H}_r^{-\top}(-\theta_r)\boldsymbol{a}_t \tag{16}$$

The **third component (rotation within plane)** is to choose a planar revolution of the face in order to finally move the markers onto the anticipated target positions $\boldsymbol{A}_{ir}$ by means of a rotation

[2] If demanded, full technical detail can be provided as an appendix.

$\mathbf{H}_p(\theta_p)$ in $\boldsymbol{a}^\top$ around $\boldsymbol{A}_c$. The respective operator $\hat{\mathbf{H}}_p$ is obtained applying (10) to $\boldsymbol{a}^\top$ and $\boldsymbol{A}_c$, whereas the angle θ_P again requires intersecting the circular path of a point $\boldsymbol{A}_e$ with a transversally oriented plane $\boldsymbol{a}_i^\top$ through the corresponding target point $\boldsymbol{A}_{ir}$ (Fig. 1, right). Once this plane determined, e.g. as the span of $\boldsymbol{A}_{ir}$, and two points $\boldsymbol{A}_{c1}$, $\boldsymbol{A}_{c2}$ on $\hat{\mathbf{H}}_p$'s axis, the calculations are analogous to (13)ff.

4.2 Visibility

General projective coordinates are "unoriented", so the distinction between the front- and backside of the face can only be defined incrementally by way of relating the current to an initially visible or invisible position of the face. For a displacement of the face, it changes visibility if-and-only-if for the respective "dual" motion of the optical center $\boldsymbol{O}$, the latter changes sides with respect to the face's plane, i.e. if $\boldsymbol{a}^\top \cdot \boldsymbol{O}$ changes sign.[3] For the above introduced one-parameter motions such an event is precisely characterized by an amplitude τ_0, or a pair of angles θ_0^+, θ_0^-, resulting from (11), or respectively (13)ff, applied to the optical center (Fig. 2, right).

Thus, if the task requires to translate towards and beyond τ_0, pure translational feed-forward is limited to $\delta\tau_0$, $0 < \delta < 1$ properly chosen, and a respective reorientation of the face is required before τ_0. However, if the task requires to rotate one plane onto another, there exists always two trivially ambiguous solutions (15), where the one outside $\theta_0 \notin \left[\theta_0^-, \theta_0^+\right]$ changes visibilty and the other one does not. This argument is used for section 4.3 particularly to determine the "right" sense of θ_r (15), such that the frontside but not the backface is turned towards to camera (Fig. 2). Additionally, heavy use is made of this argument in the implementation of section 4.1 in order to avoid turning the face backside onto its fronside, or vice versa (Fig. 1). Besides that, there are only three simple classes of motions which don't affect visibility: a translation along the optical axis, revolution around the optical axis, and thirs a planar motion of the feature points within their plane, which is the component $\hat{\mathbf{H}}_p$ of our trajectories that doesn't affect visibility. Please that that in presence of the second stereo camera, the above arguments apply independently to both of them, such that the most conservative solution has to be taken.

4.3 Generation

We now show how to create continuous trajectories that simultaneously drive the three independent parts of the task. Basically, what we will obtain is a "Cartesian motion" of the face with respect to the chosen center (Fig. 5). As for the product-of-exponentials (6), the product has to be in reverse order for the trajectory function to have the desired characteristics:

$$\mathbf{H}_d\left(\tau(\sigma), \theta_r(\sigma), \theta_p(\sigma)\right) \quad = \quad \exp(\tau(\sigma)\hat{\mathbf{H}}_t)\ \exp(\theta_r(\sigma)\hat{\mathbf{H}}_r)\ \exp(\theta_p(\sigma)\hat{\mathbf{H}}_p), \tag{17}$$

$$\tau(\sigma) = \mu_d(\sigma)\tau, \qquad \theta_r(\sigma) = \mu_r(\sigma)\theta_r, \qquad \theta_p(\sigma) = \mu_p(\sigma)\theta_p, \tag{18}$$

where the μ are monotonically growing functions $[0, t_*] \to [0, 1]$ subject to visibility constraints. More precisely, $\mu_p(\sigma)$ is always unconstrained, whereas the visibility constraints on μ_r and μ_t are coupled: if $\mathbf{H}_r^{-\top}(\theta_P(\sigma))\boldsymbol{a}$ is visible then $\mu_t(\sigma) < \tau_0$, and $\mu_t(\sigma) > \tau_0$ otherwise. Vice versa, if $\mathbf{H}_t^{-\top}(\tau(\sigma - 1))\boldsymbol{a}_*$ is visible then $\mu_r(\sigma) < \theta_0$, and $\mu_r(\sigma) > \theta_0$ otherwise. Remember that θ_0,τ_0 are the values where visibility changes.

[3] Remember that projective displacments do preserve the scale of orbit ρ and especially their sign.

Either of these cases can be used to drive a feed-forward in either τ or θ_r while constraining the other correspondingly. Clearly, a linear decay of time-to-goal arises for $\mu(\sigma) = \sigma$, whereas exponential decay arises for $\mu(\sigma) = 1 - \exp(-\frac{\sigma}{t_*})$ like in a classical feed-back loop. Additionally, an initial very flat plateaux in the respective μ allows to implement trajectories like rotation-first, transation-first, planar-first, etc. Although robot guidance through tracking the trajectories by image-based visual servoing is now feasible, we investigate a directly computed control which combines feed-forward control with a feed-back error in each component of the task.

5 Control

In this section we devise a feed-forward strategy to steer a control loop towards trajectories defined under the restrictions imposed by visibility or by additional trajectory constraints $\mu(\sigma)$, where image feed-back implicitely results in a 3-dof or 2-dof Cartesian control-error. So, the control-error will reflect a 3D rigid motion and no longer a linear image motion. For directly applying the results of the previous section, however, we have to manipulate the projective kinematic model to come up for varying joint configurationa as arising during robot motion.

5.1 Projective Kinematics of a Moving Robot

The kinematic model as it has been presented so far is valid only around the zero of the robot. For the robot in a general configuration $\boldsymbol{q}_g$, it has to be expanded around $\boldsymbol{q}_g$, which actually means to shift the joint-space like $\boldsymbol{q}(t) = \boldsymbol{q}^0(t) - \boldsymbol{q}_g$, where values coming from the former zero $\boldsymbol{q}^0 = \mathbf{o}$ will be label with a superscripted ^{0}. So, all equations remain valid as long as the model can always be shifted such that $\boldsymbol{q}(t) = 0$. This is done in (19), which is intuitively explained as follows: for each joint i, first its own displacement expressed by the truncated forward kinematics $\bar{\mathbf{H}}_i = \bar{\mathbf{H}}_i(\boldsymbol{q}_g)$ (6) must be undone, whereupon the velocity $\dot{q}_i$ can be applied to operator $\hat{\mathbf{H}}_i^0$ as beforehand, after that, it must return to $\boldsymbol{q}_g$, again.

$$\hat{\mathbf{H}}_i\Big|_{\boldsymbol{q}_g} = \mathbf{H}_i^0 \cdot \hat{\mathbf{H}}_i^0 \cdot \hat{\mathbf{H}}_i^{0^{-1}}, \qquad \mathbf{J}_H\Big|_{\boldsymbol{q}_g}(\boldsymbol{M}(t)) = \Big[\hat{\mathbf{H}}_1\boldsymbol{M}(t)\cdots\hat{\mathbf{H}}_6\boldsymbol{M}(t)\Big]^{4\times6} \tag{19}$$

The current value of the Jacobian and of $\boldsymbol{H}_q$ can now be calculated as in (6), however using the current values of the operators and of the projective points $\boldsymbol{M}(t)$. Summarizingly, we have an analytical expression for the instantaneous model of the robot's kinematics and Jacobian, both projective, the latter being a local linear model of the interaction between joint-velocities and projective- or image-velocities. So, the joint-space motions $\hat{\boldsymbol{q}}$ arising in sections 3 and 4 can be used to directly control the robot!

5.2 Direct Control

At each iteration, consider the current operators $\hat{\mathbf{H}}_i$ and the current 3D-reconstruction[4] of the markers as input $\boldsymbol{A}_i = \boldsymbol{M}_i(t)$ to the task partitioning in section 4.1, and determine for the 3D-reconstruction of the targets $\boldsymbol{A}_{i*}$ the three joint-space motions $\hat{\boldsymbol{q}}_t, \hat{\boldsymbol{q}}_r, \hat{\boldsymbol{q}}_p$ which correspond locally around $\boldsymbol{q}_g$ to the desired robot motions. The distance-to-target along the trajectory is thus taken

[4] Again, we allow for the general case of a projective reconstruction.

as 3-dof feed-back error $(\tau, \theta_R, \theta_P)$ which allows to compute directly the joint-command as a gain-weighted sum (20). However, a systematic integration-error is arising in this control law (21), since the rotational motions $\hat{\mathbf{H}}_{r,p}$ do not commutate, and thus fail to integrate to the desired trajectory $\mathbf{H}_d$ (17). For the gains being small, however, or for the control being recalculated at high frequencies, this version **directTHREE** of the direct control should allow executing a complicated reaching task without the deviations from the desired behaviour becoming too strong.

$$\boldsymbol{e} = (\tau, \theta_r, \theta_p)^\top, \quad -\dot{\boldsymbol{e}} = (\lambda_t, \lambda_r, \lambda_p)\ \boldsymbol{e}, \quad \dot{\boldsymbol{q}} = \left[\hat{\boldsymbol{q}}_t\ \hat{\boldsymbol{q}}_r\ \hat{\boldsymbol{q}}_p\right] \dot{\boldsymbol{e}}, \tag{20}$$

$$\mathbf{H}_d(\dot{\boldsymbol{e}}) \approx \exp(\lambda_t \tau \hat{\mathbf{H}}_t + \lambda_r \theta_r \hat{\mathbf{H}}_r + \lambda_r \theta_r \hat{\mathbf{H}}_p), \quad \text{for } \theta_r, \theta_p \text{ small.} \tag{21}$$

By construction, the feed-forward is such that the center is undergoing a pure translation, and the face is undergoing the pure rotation $\mathbf{H}_s = \exp(\theta_s \hat{\mathbf{H}}_s)$ around the center. These two motions are in fact decoupled, as the summed operators can be shown to integrate as expected (22). Hence, this part of directTHREE is valid and so λ_t can be set as high as visibility permits.

In consequence, the second version **directTWO** is based on a single effective rotation (e.g. for constant $\frac{\lambda_r}{\lambda_p}$), controlling the 2-dof feed-back error $(\tau, \theta_s)^\top$ now however by a sound direct sum $\hat{\boldsymbol{q}}_t + \hat{\boldsymbol{q}}_s$, where $\hat{\boldsymbol{q}}_s$ is the rotation's joint-velocity (23).

$$\exp^{(\lambda_t \tau \hat{\mathbf{H}}_t)} \exp^{(\lambda_s \theta_s \hat{\mathbf{H}}_s)} = \exp^{(\lambda_t \tau \hat{\mathbf{H}}_t + \lambda_s \theta_s \hat{\mathbf{H}}_s)} \tag{22}$$

$$\boldsymbol{e} = (\tau, \theta_s)^\top, \quad -\dot{\boldsymbol{e}} = (\lambda_t, \lambda_s)\ \boldsymbol{e}, \quad \dot{\boldsymbol{q}} = \left[\hat{\boldsymbol{q}}_t\ \hat{\boldsymbol{q}}_s\right] \dot{\boldsymbol{e}}, \quad \hat{\mathbf{H}}_s = \Sigma_{i=1}^{k} \hat{\boldsymbol{q}}_{si} \hat{\mathbf{H}}_i, \tag{23}$$

In order to find this rotation, we make us of the Campbell-Baker-Hausdorff formula (CHB) known in Lie-group theory. For two given operators, it relates their product-of-exponentials to the exponential of an (infinite) sum of higher order Lie brackets $[\hat{\mathbf{H}}_r, \hat{\mathbf{H}}_p]$[5] of the operators. In contrast to truncating this sum as done in [5], we have for our case a direct solution (24), [14] since $\hat{\mathbf{H}}_r$ and $\hat{\mathbf{H}}_p$ are both conjugate forms of $so(3)$, the Lie algebra of 3D rotations, here those around the center point. Note additionally that only a and b in (25) have a cos term, so the first-order approximation in (21) is valid.

$$\mathrm{e}^{(\theta_r \hat{\mathbf{H}}_r)} \mathrm{e}^{(\theta_p \hat{\mathbf{H}}_p)} = \mathrm{e}^{(\theta_s \hat{\mathbf{H}}_s)}, \quad \hat{\mathbf{H}}_s = (\sin \tfrac{\theta_s}{2})^{-1} \left(a \hat{\mathbf{H}}_r + b \hat{\mathbf{H}}_p + c \left[\hat{\mathbf{H}}_r, \hat{\mathbf{H}}_p\right] \right), \tag{24}$$

$$a = \sin \tfrac{\theta_r}{2} \cos \tfrac{\theta_p}{2}, \quad b = \cos \tfrac{\theta_r}{2} \sin \tfrac{\theta_p}{2}, \quad c = 2 \sin \tfrac{\theta_p}{2} \sin \tfrac{\theta_r}{2}. \tag{25}$$

6 Experiments

In this section, we validate and evaluate the above theoretical results on benchmar test configuration which is known to be a degenerate one in the monocular case[3]: a rotation of 180^o around the optical axis, i.e. the stereo rig's roll axis in our case (Fig. 3). Additionally, a potential self-occlusion is enforced by the face having a transversal orientation relative to the image planes. Besides that, the chosen dimensions correspond to them of our experimental system and the projective kinematic data was taken from a former self-calibration experiment [13]. First, three classical stereo servoing laws are tried (Fig. 4): pseudo-inverse of the stacked Jacobians [8], their block-wise pseudo-inverse

[5] $[\hat{\mathbf{H}}_r, \hat{\mathbf{H}}_p] = \hat{\mathbf{H}}_r \hat{\mathbf{H}}_p - \hat{\mathbf{H}}_p \hat{\mathbf{H}}_r = \mathbf{H}_{PE}^{-1} \hat{\mathbf{T}}_r \hat{\mathbf{T}}_p - \hat{\mathbf{T}}_p \hat{\mathbf{T}}_r \mathbf{H}_{PE} = \mathbf{H}_{PE}^{-1} [\hat{\mathbf{T}}_r, \hat{\mathbf{T}}_p] \mathbf{H}_{PE}$, with $\hat{\mathbf{T}}_{r,p}$ having the classical anti-symmetric form of $so(3)$ as upper 3×3 block.

[10], and a Jacobian for 3D Euclidean(!) points (like (6) but in Euclidean space). The second one, which basically sums two independent monocular controls, diverges while moving towards infinity. The other two run into the self-occlusion while moreless translating towards the target straight into a local minimum. Both slowly manage to escape due to some perturbations, and turn the face almost in-place, again through a self-occlusion, before finally converging.

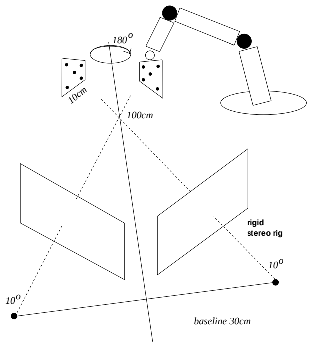

Fig. 3. Experimental setup.

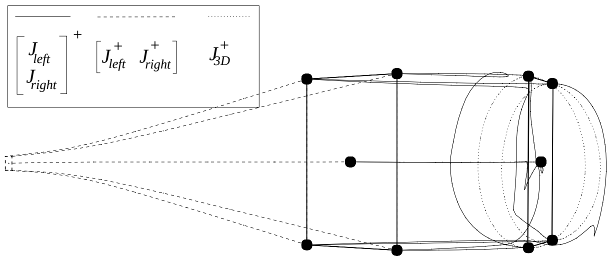

Fig. 4. Failure of classical stereo visual servoing

Second, trajectory generation from section 4 is tested. Figure 5 illustrates the solutions found using (17), where all the $\mu(\sigma)$ are linear and μ_t is rather steep in Fig. 6. The figures are rendered from a central view close to the stereo center, thus evidently the self-occlusion has been successfully. Besides this illustrative example, each iteration of the direct control consists of a newly generated feed-forward trajectory. So, the control experiments also validate the reliability and precision of the trajectories (Figs. (10), (9)).

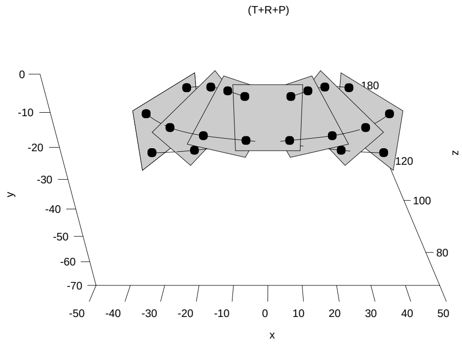

Fig. 5. Trajectories: visibility is preserved.

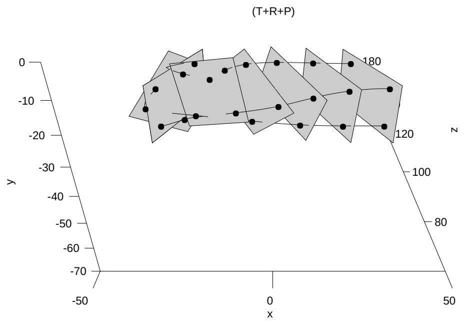

Fig. 6. Trajectories: early translation.

Third, the two control law are compared with respect to the intrinsic bias of directTHREE (21). Figure 7 shows this as a very small deviation of their image trajectories. More remarkably is the center point's deviation from the desired straight-line trajectory. Also figure 8 shows this deviation, but decreasing with smaller gains. A first conjecture is that this behaviour reflects the integration-error between the desired "Carthesian"-velocities $\hat{\mathbf{H}}$ and the actually driven joint-velocities $\hat{\mathbf{q}}$. This is confirmed by the innermost trajectory for which the joint-velocities were limited to 5^o.

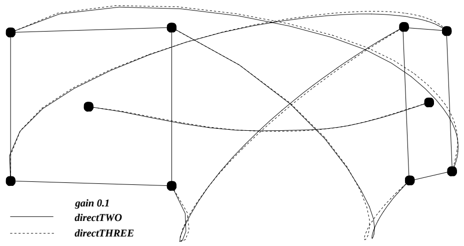

Fig. 7. directTHREE versus directTWO.

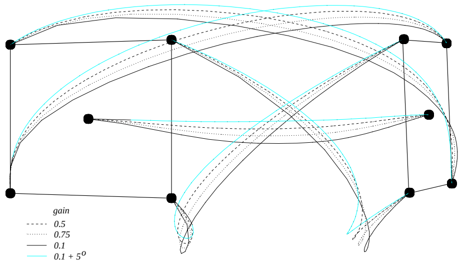

Fig. 8. Various gains + joint-speed limit.

Forth, both control-errors (20), (23) are confirmed to have exponential convergence rate (Figs. 9, 10). In the case of directTHREE, the translation error τ, which is ambiguous by the center's scale ρ, is normalized to 100, and its decay is compared to Euclidean ground-truth. The evident difference is only an apparent one, since it is actually absorbed by a reciprocal scaling of $\hat{\boldsymbol{q}}_t$. In addition to that, the influence of using a projective model is compared to using a respective Euclidean one, where for both the direct control was run, resulting in an almost perfect overlap of the curves for θ_r and θ_p (Fig. 10). The results of the directTWO law, once with and once without limited joint-speed, is depicted in figure 9. The curve of θ_s clearly reflects the task's overall rotation of 180^o, which was beforehand split between two rotational motions.

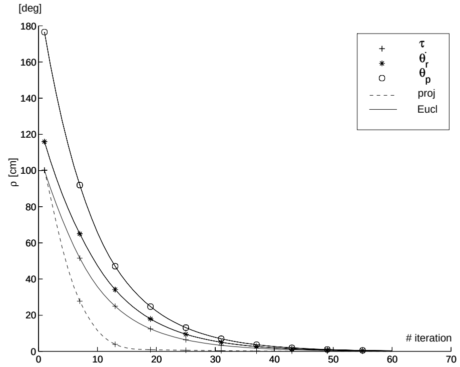

Fig. 9. directTWO: control-error.

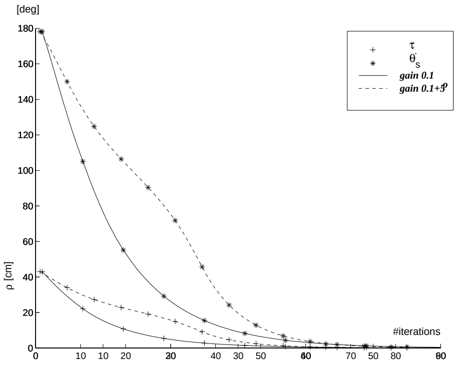

Fig. 10. directTHREE: control-error.

Fifth, the error of the marker's image projections as plotted in figure 11 clearly shows that it no longer has an exponential decay, not even a monotonic one. The zero-line actually reflects the center's straight horizontal trajectory. Finally, the corresponding trajectories in joint-space are given in figures 12, once without and once with the 5^o limit. Apparently it is the initially high velocities of q_1 and q_3 which are the cause for the above mentioned integration-errors resulting in a drift away from the straight line.

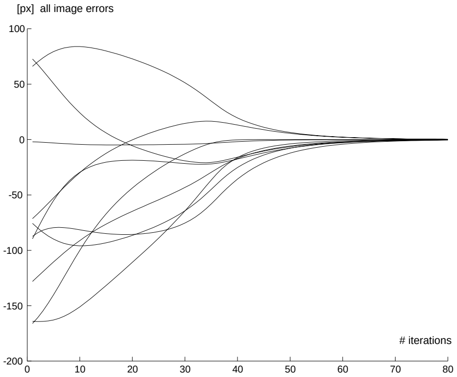

Fig. 11. Image errors.

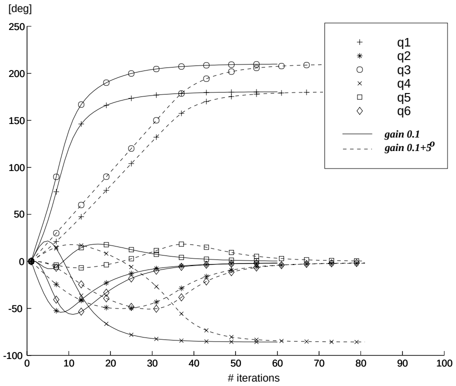

Fig. 12. Joint-space motion.

7 Discussion and Perspectives

In summary, we have presented an efficient method for visual guidance and control of a robot in projective space. It is based on stereo vision and a projective model of the robot vision system, thus independent of its metric geometry. The described mobility- and visibility constraints exploit spatial information from the stereo, aiming to guide the robot along visually feasible trajectories and so assure global validity of the visual servoing. The proposed control combines the advantages of t3D configuration is obtained. The experiments show the efficiency of the visual control scheme, where the identified "drift" away from the nominal behaviour is due to the systematic difference between integrating joint-velocities and Cartesian velicities and is typical for low control rates. The compensation of this drift in the feed-back part is straight-forward, but an important trade-off has to be made: tracking a once calculated trajectory or recalculation on-line new trajectories.

In conclusion, the presented formulation of trajectory functions from visually defined motion primitives significantly improves the performance of visual servoing systems and extends its range of applicability together with the high flexibility of now available non-metric systems. Future work should further exploit the underlying geometry of the problem in order to come up with task description beyond simple reaching and control laws allowing to execute such complex task, especially their dynamics.

References

1. E. Berry, P. Martinet, and J. Gallice. Trajectory generation by visual servoing. In *Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS'97)*, volume 2, pages 1066–1072, Grenoble, France, September 1997.

2. F. Chaumette. *Visual Servoing*, volume 7 of *World Scientific Series in Robotics and Automated Systems*, chapter Classification and Realization of the Different Vision-based Tasks, pages 199–228. World Scientific, Singapour, 1993.
3. F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. In *The Confluence of Vision and Control*, number 237 in LNCIS. Springer-Verlag, 1996.
4. F. Devernay and O. Faugeras. From projective to Euclidean reconstruction. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'96)*, pages 264–269, San Francisco, CA., June 1996.
5. T. Drummond and R. Cipolla. Visual tracking and control using Lie algebras. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'1999)*, volume 2, pages 652–657, Ft. Collins, CO, June 1999.
6. B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation*, 8(3):313–326, June 1992.
7. O. D. Faugeras. What can be seen in three dimensions with an uncalibrated stereo rig. In *Proc. 2nd European Conference on Computer Vision (ECCV'92)*, pages 563–578. Springer Verlag, Santa Margherita Ligure, Italy, May 1992.
8. G. Hager. A modular system for robust positioning using feedback from stereo vision. *IEEE Trans. on Robotics and Automation*, 13(4):582 – 595, August 1997.
9. R. I. Hartley. Projective reconstruction and invariants from multiple images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(10):1036–1041, October 1994.
10. R. Horaud, F. Dornaika, and B. Espiau. Visually guided ooject grasping. *IEEE Transactions on Robotics and Automation*, 14(4):525–532, August 1998.
11. E. Malis, F. Chaumette, and S. Boudet. 2 1/2 d visual servoing. *IEEE Trans. on Robotics and Automation*, 15(2):238–250, April 1999.
12. A. Ruf and R. Horaud. Rigid and articulated motion seen with an uncalibrated stereo rig. In *Proc. of the International Conference on Computer Vision (ICCV'99)*, pages 789 – 796, Korfu, Greece, September 1999.
13. A. Ruf, F. Martin, B. Lamiroy, and R. Horaud. Visual control using projective kinematics. In *Proc. of 9th International Symposium on Robotics Research (ISRR'99)*, page to appear, Snowbird, UT, October 1999. Springer-Verlag.
14. J. Selig. *Geometrical Methods in Robotics*. Springer, 1996.
15. M. Spratling and R. Cipolla. Uncalibrated visual servoing. In E. T. R. B. Fisher, editor, *British Machine Vision Conference (BMVC'96)*, pages 545–554, 1996.
16. A. Zisserman, P. A. Beardsley, and I. D. Reid. Metric calibration of a stereo rig. In *Proc. IEEE Workshop on Representation of Visual Scenes*, pages 93–100, Cambridge, Mass., June 1995.