Visual Control Using Projective Kinematics
Andreas Ruf, Frederick Martin, Bart Lamirov, Radu Horaud

To cite this version:

HAL Id: inria-00590124
https://inria.hal.science/inria-00590124
Submitted on 3 May 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Visual Servoing Using Projective Kinematics

Andreas Ruf\(^1\), Frédéric Martin\(^2\), Bart Lamiro\(y\)\(^3\) and Radu Horaud\(^4\)

INRIA Rhône-Alpes and GRAVIR-CNRS, 655, avenue de l’Europe, 38330 Montbonnot St. Martin, FRANCE,
\(^1\)Andreas.Ruf@inrialpes.fr, \(^2\)Frederick.Martin@inrialpes.fr, \(^3\)Bart.Lamiroy@inrialpes.fr, \(^4\)Radu.Horaud@inrialpes.fr

Abstract

This paper presents a new approach to controlling a robot from video-feedback. In contrast to classical approaches, the entire system is modeled by means of projective geometry. For the robot’s geometry in particular, we introduce a new formalism “projective kinematics”. As a result, motions in joint- and image-space can be related without metric calibration, and a corresponding visual control law can be derived. We present experiments for projective robot calibration and visual servoing, both using uncalibrated cameras.

1. Introduction

Visual servoing of robot manipulators is a key technique where the appearance of an object in the image plane is used to control the velocity of the end-effector, aiming to guide it to a desired position. The majority of visual servoing methods proposed so far use calibrated robots and calibrated cameras. Although it has been shown that the behavior of visual control loops does not degrade too much in the presence of coarse calibration, the latter remains a complex and time-consuming procedure, requiring special-purpose devices such as theodolites and calibration jigs.

In this paper we introduce a new representation of robot manipulators and of articulated mechanisms in general. Unlike previous approaches, a non-metric description of robot motion based on projective transformations is investigated. To control a robot using projective representations rather than Euclidean ones, both the robot’s direct kinematic map and the Jacobian matrix must be defined non-metrically.

The elementary joint motions that can be performed by a robot manipulator are pure rotations and translations. They give rise to corresponding projective mappings which can be parameterized as special kinds of homographies, called projective rotations and projective translations. The Lie-group properties of these projective representations of elementary motions allow us to characterize the direct kinematic map and the Jacobian matrix of a manipulator, leading to projective equivalents to the classical metric ones. They replace in a single expression both the classical robot Jacobian (joint to end-effector velocities), and the classical Jacobian of image-based visual servoing (3D to 2D point-velocities).

Finally, we provide a practical method in order to estimate the projective kinematic model, to calculate the projective Jacobian, and we report on an experimental system for visual control using projective kinematics and a pair of uncalibrated cameras.

2. Metric and Projective Camera Spaces

In this section, we recapitulate the properties of a stereo camera. We will view it as a triangulation device which can be operated either in metric or in projective mode, depending on the degree of calibration, and perform a respective reconstruction of the three-dimensional workspace. A stereo rig being calibrated allows a metric camera space to be defined. It being uncalibrated allows only a projective camera space to be defined, in which Euclidean notions such as orthogonality, angle, and length, as well as affine notions such as parallelism, and length-ratio lack definition. But in both sorts of ambient spaces stereo projection can be expressed by a pair of projection matrices, mapping 3D world-points \(M\) or \(N = (x, y, z, 1)^T\) onto 2D image-points \(m = (u, v, 1)^T, m' = (u', v', 1)^T\). In homogeneous\(^1\) coordinates they are:

\[
\text{metric: } m \simeq [K|0] N, \quad m' \simeq [K'R'|K'0] N,  \\
\text{projective: } m \simeq PM, \quad m' \simeq P'M,  
\]

(1)

where \(K, K'\) and \(R', t'\) hold an intrinsic and an extrinsic calibration, respectively, and \(P, P'\) hold an epipolar calibration, only.

Both cases, (1) and (2), describe a pair of projection rays, and solving for \(M\) or \(N\) amounts to triangulating

\(^1\)\(\simeq\) denotes the homogeneous equality “up-to-scale”.
in the respective ambient space:

\[ \lambda M = o \]  

(3)

The so reconstructed points are related by a constant “upgrade” mapping from projective to Euclidean camera space, which encapsulates calibration of a rigid stereo rig [1]: a pair of rigidly linked cameras with fix intrinsic parameters, defining the special plane \( a^T \) described later on,

\[ N \simeq H_{PE}M, \quad H_{PE} = \begin{bmatrix} K^{-1} & 0 & 1 \\ & & \\ 0 & & \\ & & & 1 \end{bmatrix}. \]  

(4)

Switching between ambient spaces also means switching between transformation groups, the consequences of which will be illustrated now. For robotics, coordinates and transformations being homogeneous allows displacements to be linearly represented by matrices \( T \in SE(3) \) at a price no higher than adding a fourth coefficient \( 1 \) and a fourth row \((0, 0, 0, 1)\). For computer vision, coordinates being homogeneous and projective allows perspective projection – being actually non-linear –, projective reconstruction (3), or a general projective transformation (“homographies”) to be linearly represented by matrix equations. The price is no higher than allowing a free scalar in all vectors and matrices, e.g. all \( \lambda M \) solve the reconstruction (3).

Consequently, a \( 4 \times 4 \) homography in projective camera space has 15 degrees-of-freedom (\( dof \)), thus requires five points to be completely defined, in contrast to three points required to define a displacement. Here, we only sketch a linear method to determine a homography given five 3D-points \( A_p \) and their coordinates \( B_p \) after a transformation, e.g. a motion or a change of perspective. For greater detail consult [2].

\[
\begin{align*}
\lambda_pA_p = HB_p, & \quad p = 1, \ldots, 5 \\
B = [B_1 B_2 B_3 B_4], & \quad A = [A_1 A_2 A_3 A_4], \\
H' = A^{-1}B, & \quad D = \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4) \\
H = \gamma DH', & \quad DA_5 \equiv HB_5.
\end{align*}
\]

In particular, the fix homographic relation \( H_{PE} \) between projective and metric camera space determines a projective camera frame, that can be imagined as five points in rigid linkage with the cameras. Therefore, rigid displacements \( T_{RT} \) in metric camera space act on projective camera space as a projective displacements \( H_{RT} \), a group of homographies conjugate to \( SE(3) \). They inherit its Lie group structure, and corresponding Lie-algebra representations \( H_{RT} \) [3]

\[ H_{RT} \simeq H_{PE}^{-1}T_{RT}H_{PE}, \quad H_{RT} = \exp(H_{RT}). \]  

(5)

Naturally, points \((x, y, z)T \) in metric camera space lie on a plane \((0, 0, 0, 1)\) called “at infinity”. Finite displacements leave this topological property invariant, expressed by a fourth coefficient either \( 0 \) or \( 1 \). Switching to projective means that the infinity plane has the fix but unknown coordinates \( a^T \) in \( H_{PE} \), but still a finite projective displacement always maps the plane at infinity onto itself.

### 3. Projective Kinematics

In this section, closed form expressions for the motion and kinematics of a robot manipulator are recapitulated, where the chosen formalism based on the exponential of \( 4 \times 4 \) matrices is chosen in order to introduce first metric analogs to the notions of projective kinematics introduced subsequently. In contrast to the popular Denavit-Hartenberg method, the joint-transforms are expressed relative to a single reference frame: here (metric) camera space. Therefore, switching the ambient space from metric to projective allows to elegantly and directly generalize to a projective formulation of robot kinematics. Its building-blocks are “projective rotations” which represent elementary revolutions of the joints.

For concreteness, consider a robot at the origin zero \( q = o \) of its joint-angle space \( q \in R^6 \). Its joint axes, being located at points \( e_i \) and having unit directions \( \omega_i \), determine twists \( \hat{T}_R \) [4] which act on camera space as a pure rotation \( T_R(\theta_i) \) ([\( \omega_i \] is \( 3 \times 3 \) anti-symmetric form).

\[
\hat{T}_R = \begin{bmatrix} [\omega_i]_\times \omega_i \times c_i \\ 0 \end{bmatrix}, \quad T_Ri = \exp(\theta_i\hat{T}_Ri). \]  

(6)

Therefore, given a general joint-space motion \( q \in R^6 \), the revolutions of all joints can be concatenated to a product-of-exponentials (\( poe \)) [5], describing the end-effector’s displacement

\[
T(q) = \prod_{i=1}^{6} \exp(\theta_i\hat{T}_Ri). \]  

(7)

Obviously, for the robot having moved to a configuration \( Q \) other than zero, the twists change say to \( \hat{T}_{Rj}(Q) \), reflecting the axis’ new locations. Thus the formulation of the robot Jacobian in (8) describing the end-effector’s velocity for the robot at \( Q \) and driving joint-velocity \( \dot{q} \):

\[
\hat{T}(q) = \hat{\theta}_1\hat{T}_{R1}(Q) + \cdots + \hat{\theta}_6\hat{T}_{R6}(Q). \]  

(8)
Let’s now consider what happens to the above formulæ if we switch to projective camera space, where the interested reader is referred to [3] or [6] for greater detail. Formally, the projective robot motion writes simply as $H(q) = H^{-1}_{PE}T(q)H_{PE}$ (5), (7). Moreover, the fact that all revolutions are represented in camera space permits also to conjugate the inner factors of the projective law in projective camera space that allows for directly servoing robot joint velocities using video-feedback from a rigid but uncalibrated stereo rig.

Having a closer look at the projective rotations $H_R(\theta)$ in (10), (see [3]) for greater detail, reveals that each of them gives rise to a one-parameter group of homographies, which is algebraically characterized by its Jordan matrix $J_R$ and respective Jordan decomposition

$$H_R(\theta) = H^{-1}_{PE}T^{-1}_J \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} T_J H_{PE};$$

(11)

$$H^{-1}_J J_R(\theta) H_J,$$

(12)

where $\exp(\theta T_R) = T^{-1}_J J_R T_J$ (10). We can interpret this geometrically like: first $H_{PE}$ changes from projective to metric camera space, $T_J$ changes rigidly to a frame whose $z$-axis is on the joint axis, and finally $J_R$ is the joint’s action in this frame. However, an entirely projective representation (12) cannot separate the calibration $H_{PE}$, since camera and kinematic parameters are “scrambled” in $H_J$. All the same, the groups $H_R(\theta)$ being conjugate forms of $J_R(\theta)$, a matrix representation of $SO(2)$, form a Lie group, having a Lie algebra (14) and an exponential form (13) with closed-form solution in “Rodrigues-form”$^3$

$$H_R(\theta) = e^{\theta H_R} = I + \sin \theta \dot{H}_R + (1 - \cos \theta) \dot{H}^2_R$$

(13)

$$\dot{H}_R = H^{-1}_J J_R \dot{H}_J,$$

(14)

Therefore, we propose these Lie algebras as a faithful and purely projective representations of revolute joints. Fortunately, their matrix representation can be recovered from a group element other than by an unstable Jordan decompositions:

$$\dot{H}_R = \frac{1}{2 \sin \theta} (H_R - H^{-1}_R), \cos \theta = \frac{1}{2}(\text{tr } H_R - 2).$$

(15)

$^3$C.f. $I + \sin \theta [w]_\times + (1 - \cos \theta) [w]_\times^2$ to go from $so(3)$ to $SO(3)$.

$^4$C.f. $\frac{1}{2 \sin \theta} (R - RT)$ to go from $SO(3)$ to $so(3)$.

In practice (13) is used to define and calculate the forward kinematic map and (15) is used to calculate the projective model of a joint from a single trial motion.

As in the metric case (8), an operator $\dot{H}_R$ represents a joint in the spatial position it has at zero, but also encodes how this rotation acts on projective camera space. For the robot having moved to configuration $Q$, the projective model gives also the necessary update to the axis’ new locations (C.f. $T_R(Q)$):

$$\dot{H}_R(Q) = H_i(Q) \dot{H}_R H^{-1}_i(Q),$$

where

$$H_i(Q) = \prod_{j=1}^{i} e^{Q_j H_{R_j}}, \text{ and } \dot{H}_R(Q) = \dot{H}_R.$$  

(17)

In practice, this formalism is used in projective robot-calibration (section 5.1.) and in the computation of the Jacobian matrix (section 4.1.).

4. Visual Control

In this section, we derive and explain the relationship between velocities of joint-space motion, projective motion, and image motion. We will formulate a control law in projective camera space that allows for directly servoing robot joint velocities using video-feedback from a rigid but uncalibrated stereo rig.

4.1. Projective motor-image Jacobian

The Jacobian matrix will be derived as a closed form expression, depending on the robot’s current configuration $q(t)$ and the current coordinates $M(t)$ of the considered 3D point. Its model parameters are the projective operators $H_R$ representing the joints. The projective kinematic model is update in (18) for the robot now at a general configuration $Q$ using updated operators $H_R(Q)$ described in (16). Since currently $Q = q(t)$, all exponential obviously evaluate to identity, such that the partial derivatives of $H(q - Q)$ (18) with respect to $\theta$ actually equal the updated operators:

$$H(q - Q) = \prod_{i=1}^{6} \exp \left( (q_i - Q_i) \dot{H}_R(Q) \right),$$

(18)

$$\frac{dH(q(t) - Q)}{dt} = \sum_{i=1}^{6} \frac{\partial H(q(t) - Q)}{\partial \theta_i} \frac{\partial \theta_i}{dt},$$

(19)

Equation (19) is to be read: for a general articulated motion at an instant $t$, a “velocity” of projective end-
effect motion is obtained as instantaneous linear superposition of velocity components caused by each of the joints.

Consider now a 3D-point \( M \) undergoing the trajectory \( H(q(t)) M \). A matrix form of \( M \), its “velocity” in projective camera space, follows immediately, again an instantaneous linear superposition (21)

\[
M = \frac{H(q(t)) - Q M}{dt} = \frac{\partial H(q - Q M)}{\partial q} \cdot \dot{q}, \quad (20)
\]

\[
= \begin{bmatrix} H_{R1}(Q) M, \ldots, H_{R6}(Q) M \end{bmatrix}^{4 \times 6} \cdot \dot{q}. \quad (21)
\]

The free scale \( \lambda \) in the projective reconstruction of each point (3) allows the velocity of a point to be defined likewise up to this scalar, only [3]. This unpleasant ambiguity however vanishes as soon as its image-projections \( s = G(P M(t)) \) (22) and the respective image velocity \( \dot{s} = dG(P M(t))/dt \) are considered:

\[
\dot{s} = \frac{\partial G(P M)}{\partial m} P \dot{M}, \quad \dot{s}' = \frac{\partial G(P' M)}{\partial m'} P' \dot{M} \quad (22)
\]

Globally, the Jacobians of perspective projection \( J_P \), \( J_{P'} \), and that of articulated motion \( J_H \) combine to the motor-image Jacobians, \( J(M, Q) \) and \( J'(M, Q) \), which once again show an instantaneous linear superposition of the image-velocities caused by each of the joints:

\[
\dot{s} = J_G(M) J_H(Q, M) \cdot \dot{q} = J(M, Q) \cdot \dot{q}, \quad (23)
\]

\[
= \sum_{i=1}^{6} \theta_i \cdot \begin{bmatrix} J_G(M) \hat{H}_{R_i}(Q) M \end{bmatrix}^{2 \times 6}. \quad (24)
\]

4.2. Control law

Finally, we explicit a control law based on the projective kinematic formalism. It controls the joint-velocities of the robot arm in order to move the tool to a desired workspace position. The latter is implicitly specified by the image targets \( s^*, s''^* \), i.e. the image projections of tool markers pre-recorded at the goal. The feedback consists of the current left- and right images \( s, s' \) of the markers, as they are extracted from the video stream. Along the lines of [8], a law with exponential convergence rate (Fig. 6) is obtained by applying joint velocities \( \dot{q} \) that impose the image-velocities \( \dot{s}, \dot{s}' \) to be proportional to the current error vectors \( s^* - s, s''^* - s' \), and so leveling it down to zero

\[
\dot{q} = - \begin{pmatrix} s_1 - s_1^* \\ s_1' - s_1'^* \\ s_2 - s_2^* \\ s_2' - s_2'^* \\ s_3 - s_3^* \\ s_3' - s_3'^* \end{pmatrix} + \begin{pmatrix} J(M_1, Q) \\ J'(M_1, Q) \\ J(M_2, Q) \\ J'(M_2, Q) \\ J(M_3, Q) \\ J'(M_3, Q) \end{pmatrix}. \quad (25)
\]

The Jacobian matrix is of size \( 4n \times 6 \), where \( n \) is the number of control points, but for the Jacobian matrix to be non-singular generally at least three points are required. Intuitively, a pair of image-velocities for a single point results through \( J_P, J_{P'} \) in a 3D point-velocity, thus further constrains the solution through \( J_H \) to a 3-dof linear form of joint-velocities. A second pair however, due to an additional rigid constraint, results just in a 2-dof pencil of 3D point-velocities, thus constrains the solution to a 1-dof linear form. The third pair, taking into account further two rigidity constraints, fixes the last dof of the solution. The used pseudo-inverse assures a best-fit to the desired “linear” image-velocities in the least-squares sense. A constant gain yields exponential convergence.

5. Experiments

A number of experiments including self-calibration of a robot (the robot moves its articulations, one by one, in front of an uncalibrated stereo rig) and a visual servoing experiment (the target position and robot trajectories are specified in the image planes) will illustrate the interest of the new paradigm outlined in this paper.

Throughout all the experiments, the input data consists of the image-points of four white markers on a black plate which is rigidly attached onto the end-effector. They are extracted from the raw images by first thresholding the intensity image and then localizing the centers of gravity with sub-pixel accuracy. Additionally, the robot’s joint-configuration \( q(t) \) is read
for all \( j \) \( \min_{H_{Rj}} \sum_k d\left( s^{jk}, P \exp((\theta^{jk} - \theta^{1j})H_{Rj})M^{1j}\right) + d\left( s'^{jk}, P' \exp((\theta^{jk} - \theta^{1j})H_{Rj})M^{1j}\right) \) \quad (26)

\[
\begin{align*}
\min_{\text{all } H_{Rj}} & \sum_j \sum_k d\left( s^{jk}, P H_{Rj}(q^{jk})M_Q \right) + d\left( s'^{jk}, P' H_{Rj}(q^{jk})M_Q \right), \\
\min_{\text{all } H_{Rj}} & \sum_j \sum_k d\left( s^{jk}, P H_{Rj}(q^{jk})M_Q \right) + d\left( s'^{jk}, P' H_{Rj}(q^{jk})M_Q \right) \quad (27)
\end{align*}
\]

5.1. Recovering Projective Kinematics

The basic principle is to define an objective function which takes model parameters and physical measurements, while at the same time giving a physically meaningful error measure. Minimizing this objective then determines a best fit of model parameters. Projective quantities, such as reconstructions \( M \), allow a-priori only an “algebraic” error to be defined, which lacks a physical or statistical meaning. In our case, the measurements are the robot joint angles, and the measured pixel-coordinates of the markers’ images. The angles are considered accurate, whereas the image measurements are assumed to be disturbed by random Gaussian noise. Therefore, the main objective to be minimized is the Euclidean pixel distance \( d(s, m) \) between the actually measured image points \( s = (s_u, s_v) \) and the projective image points \( m \), which are a function of the model parameters. For the sake of simplicity, we use \( d \) to denote the total error over the four image measurements.

\[
d(s, m) = \left( \frac{m_1}{m_3} s_u - s_v \right)^2 + \left( s_u - \frac{m_2}{m_3} s_v \right)^2 \right)^{1/2}. \quad (29)
\]

For concreteness, consider the \( j = 1 \ldots 6 \) joints and a trial motion (Fig. 2) of each joint \( j \) from configuration \( q^{j1} \) to configurations \( q^{jk} \), where all but the \( j^{th} \) joint remain locked, which itself moves from angle \( \theta^{j1} \) to angles \( \theta^{jk} \). The positions \( q^{j1} \) are chosen in order to obtain nice circular trajectories in the image. Analogously, the reconstructed markers in the
Table 1: Results of projective robot calibration. Note, that angles and link length result from upgrading to a metric model, subject to stability matters.

<table>
<thead>
<tr>
<th>residual image-error</th>
<th>fix 3D-points image-err [px]</th>
<th>free 3D-points image-err [px]</th>
</tr>
</thead>
<tbody>
<tr>
<td>axis 1</td>
<td>0.55</td>
<td>0.13</td>
</tr>
<tr>
<td>axis 2</td>
<td>0.14</td>
<td>0.10</td>
</tr>
<tr>
<td>axis 3</td>
<td>0.15</td>
<td>0.09</td>
</tr>
<tr>
<td>axis 4</td>
<td>0.16</td>
<td>0.12</td>
</tr>
<tr>
<td>axis 5</td>
<td>0.16</td>
<td>0.10</td>
</tr>
<tr>
<td>axis 6</td>
<td>0.24</td>
<td>0.16</td>
</tr>
<tr>
<td>axis angles</td>
<td>Factory [°]</td>
<td>Lab [°]</td>
</tr>
<tr>
<td>axis 1, 2</td>
<td>90</td>
<td>89.72</td>
</tr>
<tr>
<td>axis 2, 3</td>
<td>0</td>
<td>0.28</td>
</tr>
<tr>
<td>axis 3, 4</td>
<td>90</td>
<td>89.71</td>
</tr>
<tr>
<td>axis 4, 5</td>
<td>90</td>
<td>90.002</td>
</tr>
<tr>
<td>axis 5, 6</td>
<td>90</td>
<td>90.10</td>
</tr>
<tr>
<td>link lengths</td>
<td>Factory [mm]</td>
<td>Lab [mm]</td>
</tr>
<tr>
<td>link 1</td>
<td>0</td>
<td>0.5522</td>
</tr>
<tr>
<td>link 2 (scale)</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>link 3</td>
<td>800</td>
<td>800.5457</td>
</tr>
<tr>
<td>link 4</td>
<td>0</td>
<td>0.0633</td>
</tr>
<tr>
<td>link 5</td>
<td>0</td>
<td>0.0347</td>
</tr>
<tr>
<td>link axis 4, 6</td>
<td>0.3</td>
<td>0.1895</td>
</tr>
</tbody>
</table>

The first position are indexed as $M_j^1$, their subsequent 3D-positions as $M_j^k$, and their projective images as $m_j^k$ and $m_0^k$ (2). Hence, given $M_j^1$, all the subsequent positions are fully determined by equations (18) and (2), while the operators $H_{Rj} = H_{Ri}(q^j)$ are the unknown model parameters. Having also indexed the image measurements as $s_j^k$ and $s_j^k$, the above characterized objective function is obtained in (26). An initial guess is found by applying (15) to a single rotation. Still, the implementation of this minimization is non-trivial, since the 16 coefficients of $H_{Rj}$ have only 10 degrees of freedom, due to algebraic constrains and free “gauge parameters”. In [10] we devise a minimal parameterization, together with explicit solutions of the arising constraints. Note that in the experiments, the numerical implementations of equations (26)-(28) are over sets of 10 parameters which expand to a $4 \times 4$ operator.

We devise a three-step estimation process, in the first of which the joints $H_{Rj}$ were treated separately, and the markers $M_j^1$ were a-priori reconstructed, separately for each joint. In the second step, the so far independent operators are gathered to an explicit and complete projective kinematic model, expressed in terms of operators at zero $H_{Ri} = H_{Ri}(0)$. An initial guess can be obtained from the results of the first step using the formalism in (16). In addition, the multiply reconstructed markers are replaced by a single a-priori reconstruction, e.g. $M_Q = M_{j_1}$. Thus, step two reflects that the joints are part of one and the same articulated chain and that the four markers are rigidly linked with its end. As a results, we get a first estimate for the total of 60 parameters in the projective kinematic model after minimizing equation (27).

In an additional third step, also the 3D-coordinates of points $M_Q$ are allowed to vary and are refined a-posteriori together with the kinematic parameters (28). This removes the bias resulting from a-priori privileging points $M_Q$ as done in (27) and (26), and effectively improved precision (Table 1). In order to evaluate the metric dimensions “hidden” by the projective model, we have also quoted in Table 1 the results of a recent experiment on self-calibration of a Stäubli RX90 robot having a PUMA-alike geometry [11]. There, the camera parameters and kinematic parameters are separated such that the robot’s Denavit-Hartenberg parameters can be recovered up-to-scale. Conclusively, a projective calibration of the robot has been fit that agrees with the measurements down to their absolute precision, i.e. subpixel-accuracy of the markers. A metric self-calibration, known for its inherent unstability, shows good agreement with factory-given Denavit-Hartenberg parameters.
5.2. Visual Control using Projective Kinematics

We present results obtained with the visual servoing system under development in the VIGOR-project (ESPRIT LTR 26247). The general setup consists of an independent stereo camera controlling a six-axes Stäubli robot RX90. The task under consideration is a 6-dof alignment of tool and workpiece, in our case of a welding torch or a gripper and a metal ship-part (Fig. 3). The alignment is once recorded or just calculated from CAD-data, and reprojected onto the images of the work-piece taken at task-time [12]. Also at task-time, the projective kinematics relating the robot to the stereo system is established (section 5.1.). Then running the visual servoing loop of section 4.2. performs the alignment. Thanks to the precise model estimate, equations (2) and (18) allow to predict the marker positions from joint-angle measurements so that the burden of feature extraction and tracking is greatly reduced, resulting in increased robustness and tracking range. Figure 3 gives an idea of the complexity of displacements that can be reached. Nevertheless, the resulting joint-space trajectories are still very smooth (Fig. 5), and image-trajectories are close to the expected linear behaviour, as far as rigidity permits (Fig. 4). Additionally, the gain can be largely varied (Fig. 6) without stability of control or robustness of tracking being really affected, allowing even for an adaptive gain with steep close-to-linear convergence.

6. Discussion

We have shown how to projectively represent a kinematic model and how to recover it from image measurements only. Neither a-priori knowledge, nor metric information is required. This representation of robot kinematics is predestinated for visual control of a robot in order to perform basic reaching, grasping, or manipulation tasks. In comparison to previous work, our new approach has a number of advantages. Visual servoing can be along trajectories covering the whole visible configuration space. The sound relationship between joint and image-space prepares the ground for more intelligent control laws that incorporate joint-space reasoning. So, avoidance of joint-limits, singularities, or “workspace limits” (collisions, occlusions projected onto joint-space) become feasible, in addition to optimal control techniques, such as shortest-path, shortest-time, etc. Last but not least, this formalism opens perspectives for the design of “visual mechanisms”, meaning that a camera is assuring by means of a visual control loop that a visual kinematic constraint is respected. For instance the control of a robot arm reaching through a hole, manipulating without occluding the tele-operators’ view, or co-operating with a mobile observer robot.

Acknowledgements

The authors are very grateful towards the European Commission for financial support through the Marie-Curie fellowship FMBICT972281, and the Esprit LTR project VIGOR 26247.
References


