Robot Hand-Eye Calibration Using Structure-from-Motion

Nicolas Andreff, Radu Horaud and Bernard Espiau

The International Journal of Robotics Research 2001 20: 228
DOI: 10.1177/02783640122067372

Robot Hand-Eye Calibration Using Structure-from-Motion

Nicolas Andreff

Institut Français de Mécanique Avancée
BP 265, 63175 Aubière Cedex, France
nicolas.andreff@ifma.fr

Radu Horaud
Bernard Espiau

INRIA Rhône-Alpes and GRAVIR-IMAG
655, av. de l'Europe, 38330 Montbonnot Saint Martin, France
radu.horaud@inrialpes.fr
bernard.espiau@inrialpes.fr

Abstract

In this paper, we propose a new flexible method for hand-eye calibration. The vast majority of existing hand-eye calibration techniques require a calibration rig that is used in conjunction with camera pose estimation methods. Instead, we combine structure-from-motion with known robot motions, and we show that the solution can be obtained in linear form. The latter solves for both the hand-eye parameters and the unknown scale factor inherent with structure-from-motion methods. The algebraic analysis that is made possible with such a linear formulation allows investigation of not only the well-known case of general screw motions but also of such singular motions as pure translations, pure rotations, and planar motions. In essence, the robot-mounted camera looks to an unknown rigid layout, tracks points over an image sequence, and estimates the camera-to-robot relationship. Such a self-calibration process is relevant for unmanned vehicles, robots working in remote places, and so forth. We conduct a large number of experiments that validate the quality of the method by comparing it with existing ones.

KEY WORDS—self-calibration, structure-from-motion, Sylvester equation

1. Introduction

The background of this work is the guidance of a robot by visual servoing (Espiau, Chaumette, and Rives 1992; Andreff 1999). In this framework, a basic issue is to determine the spatial relationship between a camera mounted onto a robot end-effector (Fig. 1) and the end-effector itself. This spatial relationship is a rigid transformation, that is, a rotation and a translation, known as the hand-eye transformation. The determination of this transformation is called hand-eye calibration.

The International Journal of Robotics Research
Vol. 20, No. 3, March 2001, pp. 228-248,

The goal of this paper is to describe a technique allowing the hand-eye calibration to be performed in the working site. In practice, this requires that

- No calibration rig will be allowed.

  A calibration rig is a very accurately manufactured 3D object holding targets as visual features. Mobile robots and space applications of robotics are typical examples where a calibration rig cannot be used. During their mission, such robots may nevertheless need to be calibrated again. However, as affordable on-board weight is limited, they cannot carry a calibration object and should use their surrounding environment instead. Thus, the availability of a hand-eye self-calibration method is mandatory.

- Special and/or large motions are difficult to achieve and hence should be avoided.

  Indeed, since the hand-eye system must be calibrated on-site, the amount of free robot workspace is limited and the motions must therefore be of small amplitude. Therefore, the self-calibration method must be able to handle a large variety of motions, including small ones.

Hand-eye calibration was first studied a decade ago (Tsai and Lenz 1989; Shiu and Ahmad 1989). It was shown that any solution to the problem requires consideration of both Euclidean end-effector motions and camera motions.[1] While the end-effector motions can be obtained from the encoders, the camera motions are to be computed from the images. It was also shown, both algebraically (Tsai and Lenz 1989) and geometrically (Chen 1991), that a sufficient condition to the uniqueness of the solution is the existence of two calibration motions with nonparallel rotation axes.

1. Notice that this requirement may be implicit as in Rémy et al. (1997).

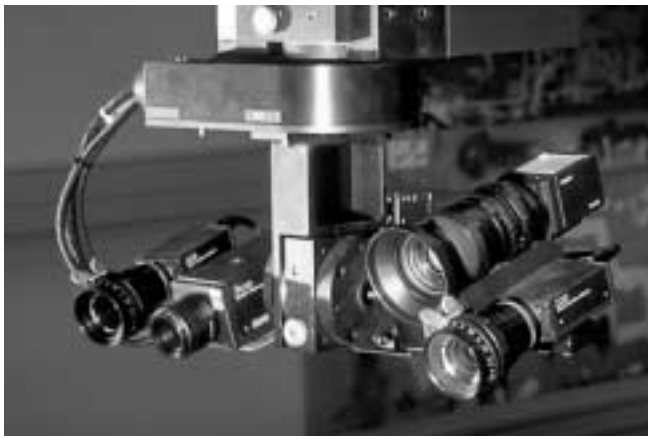

Fig. 1. Some cameras mounted on our 5-DoF robot.

Several methods were proposed (Tsai and Lenz 1989; Daniilidis and Bayro-Corrochano 1996; Horaud and Dornaika 1995; Shiu and Ahmad 1989; Chou and Kamel 1991; Wang 1992) to solve for hand-eye calibration under the assumption that both end-effector and camera motions were known. All these methods represent rotations by a minimal parameterization and use pose algorithms to estimate the camera position relative to the fixed calibration rig. Recall that pose algorithms require the 3D Euclidean coordinates of the rig targets to be known together with their associated 2D projections onto each image.

Another approach is proposed by Wei, Arbter, and Hirzinger (1998), who simultaneously perform hand-eye calibration and camera calibration without any calibration rig. However, this requires a complex nonlinear minimization and the use of a restrictive class of calibration motions. Moreover, no algebraic analysis of the problem is given.

With regard to the existing approaches, we propose a new hand-eye self-calibration method that exploits two main ideas. The first idea is that a specific algebraic treatment is necessary to handle small rotations. Indeed, minimal representations of rotation may be either singular (Euler angles, axis/angle) or nonunique (quaternions) for specific configurations. Moreover, when the rotation angle is small, it is hard to accurately estimate the rotation axis. Yet, existing approaches need to estimate the latter. Hence, one should use, as far as hand-eye calibration is concerned, as large calibration motions as possible: a rule for such a choice of large calibration motions is even given in Tsai and Lenz (1989). Instead, we prefer to use orthogonal matrices, which are always defined, even though we need both to estimate more parameters (nine instead of three to four) and to take into account the orthogonality constraints.

The second idea is that camera motion can be computed from structure-from-motion algorithms rather than from pose algorithms. Structure-from-motion estimation methods give very accurate results in terms of camera motion parameters, and a variety of techniques exist from factorization methods to nonlinear bundle-adjustment. From a practical point of view, an iterative factorization method such as the one described in Christy and Horaud (1996) achieves a good compromise between purely linear methods and highly nonlinear ones.

Our contributions can be summarized in the following way. First, hand-eye calibration is reformulated to take into account the estimation of camera motions from structure-from-motion algorithms. Indeed, camera motions are thus obtained up to an unknown scale factor, which is introduced in the formulation. Second, a linear formulation, based on the representation of rotations by orthogonal matrices, is proposed that enables small calibration motions. It allows us to give a common framework to the (already solved) general motion case as well as to singular motion cases (where the general solution fails) such as pure translations, pure rotations, and planar motions. Third, an algebraic study of this linear formulation is performed that shows that partial calibration can nevertheless be performed when the sufficient condition for the uniqueness of the solution is not fulfilled. Fourth, in-depth experiments are conducted with comparison to other methods. They show that the use of structure-from-motion does not affect the numerical accuracy of our method with respect to existing ones.

The remainder of this paper is organized as follows. Section 2 recalls the classical formulation of hand-eye calibration and the structure-from-motion paradigm. Section 3 contains the formulation of the linear hand-eye self-calibration method. Section 4 contains its algebraic analysis. Finally, Section 5 gives some experimental results and Section 6 concludes this work.

2. Background

In this section, after defining the notation used in this article, we briefly present the classical formulation of hand-eye calibration with a short description of three methods that will be used as references in the experimental section (Section 5). We then describe the estimation of camera motions, concluding in favor of Euclidean reconstruction rather than pose computation.

2.1. Notation

Matrices are represented by uppercase boldface letters (e.g., $\mathbf{R}$) and vectors by lowercase boldface letters (e.g., $\mathbf{t}$).

Rigid transformations (or, equivalently, Euclidean motions) are represented with homogeneous matrices of the form

$$\begin{pmatrix} \mathbf{R} & \mathbf{t} \\ 0\,0\,0 & 1 \end{pmatrix},$$

where $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t}$ is a 3×1 translation vector. This rigid transformation will be often referred to as the couple $(\mathbf{R}, \mathbf{t})$.

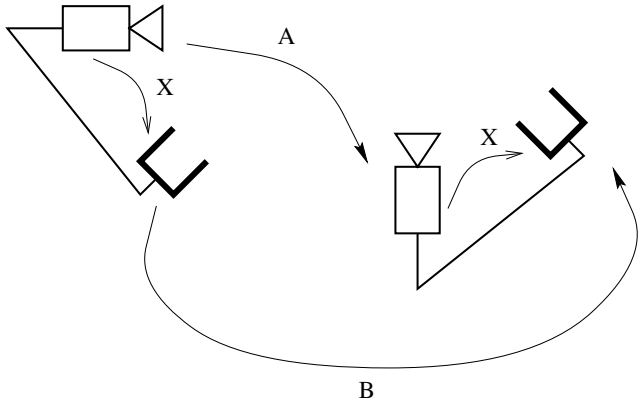

Fig. 2. End-effector (represented here by a gripper) and camera motions are conjugated by the hand-eye transformation $\mathbf{X}$.

In the linear formulation of the problem, we will use the linear operator *vec* and the tensor product, also known as Kronecker product. The *vec* operator was introduced in Neudecker (1969) and reorders (one line after the other) the coefficients of a $(m \times n)$ matrix $\mathbf{M}$ into the mn vector

$$vec(\mathbf{M}) = (M_{11}, \ldots, M_{1n}, M_{21}, \ldots, M_{mn})^T.$$

The Kronecker product (Bellman 1960; Brewer 1978) is noted $\otimes$. From two matrices $\mathbf{M}$ and $\mathbf{N}$ with respective dimensions $(m \times n)$ and $(o \times p)$, it defines the resulting $(mo \times np)$ matrix:

$$\mathbf{M} \otimes \mathbf{N} = \begin{pmatrix} M_{11}\mathbf{N} & \ldots & M_{1n}\mathbf{N} \\ \vdots & \ddots & \vdots \\ M_{m1}\mathbf{N} & \ldots & M_{mn}\mathbf{N} \end{pmatrix}. \quad (1)$$

2.2. Hand-Eye Problem Formulation

We present here the classical approach (Tsai and Lenz 1989; Chen 1991; Daniilidis and Bayro-Corrochano 1996; Horaud and Dornaika 1995; Shiu and Ahmad 1989; Chou and Kamel 1991; Wang 1992), which states that when the camera undergoes a motion $\mathbf{A} = (\mathbf{R}_a, \mathbf{t}_a)$ and the corresponding end-effector motion is $\mathbf{B} = (\mathbf{R}_b, \mathbf{t}_b)$, then they are conjugated by the hand-eye transformation $\mathbf{X} = (\mathbf{R}_x, \mathbf{t}_x)$ (Fig. 2). This yields the following homogeneous matrix equation:

$$\mathbf{AX} = \mathbf{XB}, \quad (2)$$

where $\mathbf{A}$ is estimated, $\mathbf{B}$ is assumed to be known, and $\mathbf{X}$ is the unknown.

Equation (2), applied to each motion i, splits into

$$\mathbf{R}_{ai}\mathbf{R}_x = \mathbf{R}_x\mathbf{R}_{bi} \quad (3)$$

$$\mathbf{R}_{ai}\mathbf{t}_x + \mathbf{t}_{ai} = \mathbf{R}_x\mathbf{t}_{bi} + \mathbf{t}_x. \quad (4)$$

In the method proposed in Tsai and Lenz (1989), the first equation is solved by least-square minimization of a linear system obtained by using the axis/angle representation of the rotations. Once $\mathbf{R}_x$ is known, the second equation is also solved with linear least-squares techniques.

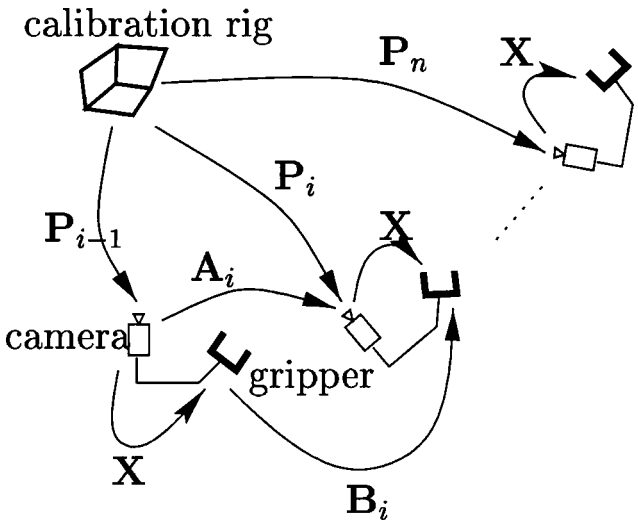

Fig. 3. Hand-eye calibration from pose estimation.

To avoid this two-stage solution that propagates the error on the rotation estimation onto the translation, a nonlinear minimization method based on the representation of the rotations with unit quaternions was proposed in Horaud and Dornaika (1995). Similarly, a method based on the unit dual quaternion representation of Euclidean motions was developed in Daniilidis and Bayro-Corrochano (1996) to solve simultaneously for hand-eye rotation and hand-eye translation.

2.3. Computing the Camera Motions

In the prior work (Tsai and Lenz 1989; Chen 1991; Daniilidis and Bayro-Corrochano 1996; Horaud and Dornaika 1995; Shiu and Ahmad 1989; Chou and Kamel 1991; Wang 1992), camera motions were computed considering images one at a time, as follows. First, 2D-to-3D correspondences were established between the 3D targets on the calibration rig and their 2D projections onto each image i. Then, from the 3D coordinates of the targets, their 2D projections, and the intrinsic camera parameters, the pose (i.e., position and orientation) of the camera with respect to the calibration rig is estimated $\mathbf{P}_i = (\mathbf{R}_i, \mathbf{t}_i)$. Finally, the camera motion between image $i-1$ and image i $\mathbf{A}_i = (\mathbf{R}_{ai}, \mathbf{t}_{ai})$ is hence obtained by simple composition (Fig. 3):

$$\mathbf{A}_i = (\mathbf{R}_i\mathbf{R}_{i-1}^T, \mathbf{t}_i - \mathbf{R}_i\mathbf{R}_{i-1}^T\mathbf{t}_{i-1}).$$

Alternatively, one may simultaneously consider all the images that were collected during camera motion. Thus, one may use the multiframe structure-from-motion paradigm (see Oliensis 1997 for a review). The advantage of structure-from-motion over pose algorithms is that the former does not require any knowledge about the observed 3D object. Indeed, structure-from-motion only relies on 2D-to-2D

correspondences. The latter are easier to obtain since they depend on image information only. There are two classes of (semi-)automatic methods to find them: a discrete approach, known as *matching* (Gruen 1985), and a continuous approach, known as *tracking* (Hager and Toyama 1998).

A relevant class of structure-from-motion methods is known as Euclidean reconstruction (Cui, Weng, and Cohen 1994; Taylor, Kriegman, and Anandan 1991; Poelman and Kanade 1994; Tomasi and Kanade 1991; Koenderink and van Doorn 1991; Christy and Horaud 1996). It assumes that the camera is calibrated (i.e., the camera intrinsic parameters are known). From this knowledge, one can reconstruct the structure of the scene and the motion of the camera up to an unknown scale factor (Fig. 4) using various methods (see below). This unknown scale factor is a global scale factor associated with the fact that rigidity is defined up to a similitude and is the same for all the camera motions in the sequence (Fig. 5). Therefore, the estimated camera motions are of the form

$$\mathbf{A}_i(\lambda) = \begin{pmatrix} \mathbf{R}_{ai} & \lambda \mathbf{u}_{ai} \\ 0\,0\,0 & 1 \end{pmatrix}, \tag{5}$$

where $\mathbf{R}_{ai}$ is the rotation of the camera between image $i-1$ and image i, λ is the unknown scale factor, and $\mathbf{u}_{ai}$ is a vector, parallel to the camera translation $\mathbf{t}_{ai}$ and such that

$$\mathbf{t}_{ai} = \lambda \mathbf{u}_{ai}. \tag{6}$$

Taking, without loss of generality, the first motion as a motion with nonzero translation allows to arbitrarily choose $\mathbf{u}_{a1}$ as a unit vector. Hence, $\lambda = \|\mathbf{t}_{a1}\|$. Consequently, the $\mathbf{u}_{ai}$s are related by $\mathbf{u}_{ai} = \mathbf{t}_{ai}/\|\mathbf{t}_{a1}\|$, and λ can be interpreted as the unknown norm of the first translation.

In summary, camera rotations are completely recovered while camera translations are recovered up to a single unknown scale factor.

In practice, which structure-from-motion algorithm should we choose? Affine camera models (Poelman and Kanade 1994; Tomasi and Kanade 1991; Koenderink and van Doorn 1991) yield simple linear solutions to the Euclidean reconstruction problem, based on matrix factorization. However, affine models are first-order approximations of the perspective model. Hence, only approximations of the Euclidean camera motions can be obtained. Besides, solutions exist (Cui, Weng, and Cohen 1994; Taylor, Kriegman, and Anandan 1991), based on the perspective model, that offer some Euclidean information on the camera motions but are nonlinear. Our choice lies, in fact, between these two classes of methods: we propose a method for Euclidean reconstruction by successive affine approximations of the perspective model (Christy and Horaud 1996), which combines the simplicity of affine methods and the accuracy of nonlinear methods.

In summary, to estimate camera motions, structure-from-motion methods are more flexible than pose computation methods, since no 3D model is needed. The drawback of lowering this constraint is that camera motions are estimated up to an unknown scale factor, which we must take into account in the hand-eye self-calibration method.

3. A New Linear Formulation

In this section, we first modify the formulation of hand-eye calibration to take into account the use of Euclidean reconstruction to compute camera motions. Then, we give a solution to this problem that handles small rotations.

3.1. Using Structure-from-Motion

For using structure-from-motion to estimate camera motions, we have to take into account the unknown scale factor λ. Indeed, the homogeneous eq. (2) becomes (compare Fig. 3 and Fig. 6)

$$\mathbf{A}_i(\lambda)\mathbf{X} = \mathbf{X}\mathbf{B}_i, \tag{7}$$

where $\mathbf{A}_i(\lambda)$ is the ith estimated camera motion. From (5) and (6), we thus obtain a set of two equations, similar to (3)–(4):

$$\mathbf{R}_{ai}\mathbf{R}_x = \mathbf{R}_x\mathbf{R}_{bi} \tag{8}$$

$$\mathbf{R}_{ai}\mathbf{t}_x + \lambda\mathbf{u}_{ai} = \mathbf{R}_x\mathbf{t}_{bi} + \mathbf{t}_x, \tag{9}$$

where the unknowns are now $\mathbf{R}_x$, $\mathbf{t}_x$, and λ.

3.2. Linear Formulation

We propose a new formulation that handles rotations of any kind. Its underlying idea is to embed the rotation part of the problem, intrinsically lying in $SO(3)$, in a larger space to deliberately free ourselves from the nonlinear orthogonality constraint. This allows us to easily find a subspace of matrices verifying (2). Then, the application of the orthogonality constraint selects, in this subspace, the unique rotation, which is the solution to the problem. This general idea is very powerful here since, as we will see, the nonlinear orthogonality constraint reduces to a linear norm constraint.

The new formulation is inspired by the similarity of (8) with the Sylvester equation: $\mathbf{UV} + \mathbf{VW} = \mathbf{T}$. This matrix equation, which often occurs in system theory (Brewer 1978), is usually formulated as a linear system (Rotella and Borne 1989; Hu and Reichel 1992; Deif, Seif, and Hussein 1995):

$$(\mathbf{U} \otimes \mathbf{I} + \mathbf{I} \otimes \mathbf{W})vec(\mathbf{V}) = vec(\mathbf{T}). \tag{10}$$

One fundamental property of the Kronecker product is (Brewer 1978)

$$vec(\mathbf{CDE}) = (\mathbf{C} \otimes \mathbf{E}^T)vec(\mathbf{D}), \tag{11}$$

where $\mathbf{C}$,$\mathbf{D}$,$\mathbf{E}$ are any matrices with adequate dimensions. Applying this relation to eq. (8) yields

$$(\mathbf{R}_{ai} \otimes \mathbf{R}_{bi})vec(\mathbf{R}_x) = vec(\mathbf{R}_x). \tag{12}$$

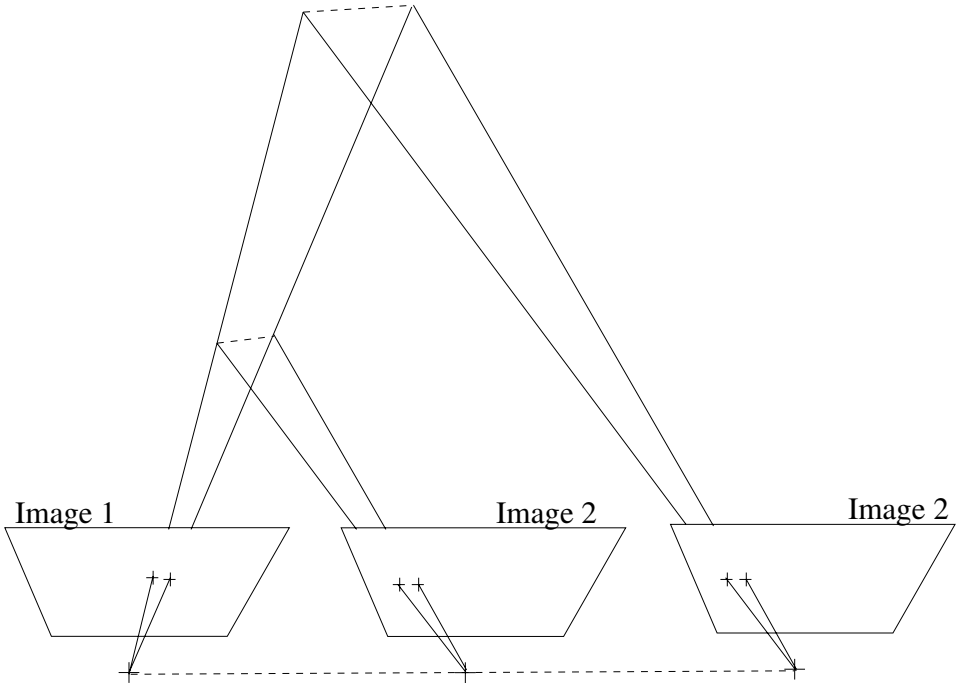

Fig. 4. Given two images, structure-from-motion methods determine the direction of the baseline between the two camera positions, but not the baseline length. Thus, the size of the reconstructed object is not determined since it is proportional to the unknown baseline length.

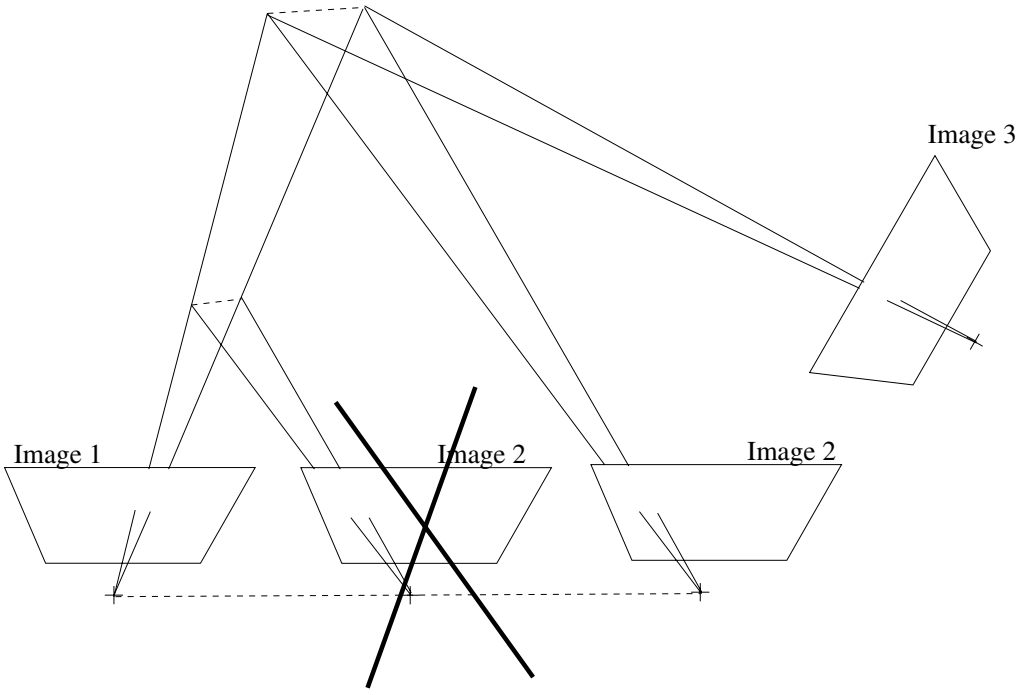

Fig. 5. Once the scale factor is resolved between the first two images, the second camera position is uniquely defined with respect to the first one and, consequently, the following camera positions are also uniquely defined.

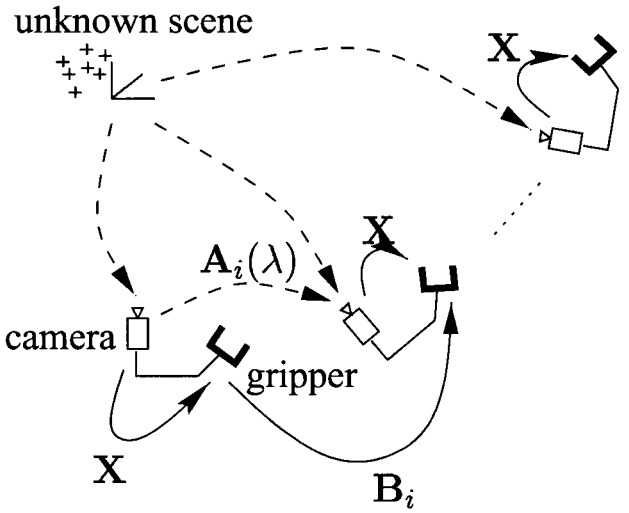

Fig. 6. From images of an unknown scene and the knowledge of the intrinsic parameters of the camera, structure-from-motion algorithms estimate, up to an unknown scale factor λ, the camera motions $\mathbf{A}_i(\lambda)$.

Introducing the notation $vec(\mathbf{R}_x)$ in eq. (9), we obtain

$$(\mathbf{I}_3 \otimes (\mathbf{t}_{bi}^T))vec(\mathbf{R}_x) + (\mathbf{I}_3 - \mathbf{R}_{ai})\mathbf{t}_x - \lambda \mathbf{u}_{ai} = 0. \quad (13)$$

We can then state the whole problem as a single homogeneous linear system:

$$\begin{pmatrix} \mathbf{I}_9 - \mathbf{R}_{ai} \otimes \mathbf{R}_{bi} & \mathbf{0}_{9\times 3} & \mathbf{0}_{9\times 1} \\ \mathbf{I}_3 \otimes (\mathbf{t}_{bi}^T) & \mathbf{I}_3 - \mathbf{R}_{ai} & -\mathbf{u}_{ai} \end{pmatrix} \begin{pmatrix} vec(\mathbf{R}_x) \\ \mathbf{t}_x \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{0}_{9\times 1} \\ \mathbf{0}_{3\times 1} \end{pmatrix}. \quad (14)$$

The question is now, What is the condition for this system to have a unique solution? and a subsequent one is, What occurs when this condition is not fulfilled?

4. Algebraic Analysis

From earlier work on hand-eye calibration (Tsai and Lenz 1989; Chen 1991), we know that two motions with nonparallel rotation axes are sufficient to determine the hand-eye transformation. We will show in this section that our new linear solution owns the same sufficient condition but also allows us to identify what can be obtained when such a sufficient condition is not fulfilled (the so-called partial calibration).

Hence, let us determine what can be obtained using various combinations of end-effector motions by successively considering pure translations, pure rotations, planar motions (i.e., containing the same rotational axis and independent translations), and finally general motions. The results of this study are gathered up in Table 1. Notice that by inverting the roles of the end-effector and the camera, we obtain the same results for the recovery of the *eye-hand* transformation (i.e., the inverse of the hand-eye transformation).

4.1. Pure Translations

Recall from eq. (3) that when end-effector motions are pure translations (i.e., $\mathbf{R}_{bi} = \mathbf{I}_3$), then camera motions are pure translations too (i.e., $\mathbf{R}_{ai} = \mathbf{I}_3$). Hence, eq. (4) becomes

$$\mathbf{t}_{ai} = \mathbf{R}_x \mathbf{t}_{bi}. \quad (15)$$

Consequently, the amplitude of camera motion is the same as the amplitude of end-effector motion, which is not the case when rotations are involved. One can therefore keep control of the camera displacements and guarantee that a small end-effector motion will not generate an unexpected large camera motion. Concerning calibration, we have the following result:

PROPOSITION 1. Three independent pure translations yield a linear estimation of hand-eye rotation $\mathbf{R}_x$ and of the unknown scale factor λ. Hand-eye translation cannot be observed.

Proof. In the case of pure translations, the upper part of the system in (14) vanishes and its lower part simplifies into

$$\left(\mathbf{I}_3 \otimes (\mathbf{t}_{bi}^T)\right) vec(\mathbf{R}_x) = \lambda \mathbf{u}_{ai}. \quad (16)$$

This implies that hand-eye translation $\mathbf{t}_x$ cannot be estimated. However, the nine coefficients of the hand-eye rotation $\mathbf{R}_x$ can be obtained as we show below. This was also demonstrated in Zhuang (1998) in the particular case where λ is known.

Let us assume temporarily that λ is known. If $\mathbf{t}_{bi} \neq 0$, then $\mathbf{I}_3 \otimes (\mathbf{t}_{bi}^T)$ has rank 3 since

$$\mathbf{I}_3 \otimes (\mathbf{t}_{bi}^T) = \begin{pmatrix} \mathbf{t}_{bi}^T & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{1\times 3} & \mathbf{t}_{bi}^T & \mathbf{0}_{1\times 3} \\ \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times 3} & \mathbf{t}_{bi}^T \end{pmatrix}. \quad (17)$$

Consequently, three linearly independent pure translations yield a full rank (9×9) system:

$$\underbrace{\begin{pmatrix} \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T) \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b3}^T) \end{pmatrix}}_{\mathbf{M}} vec(\mathbf{R}_x) = \lambda \begin{pmatrix} \mathbf{u}_{a1} \\ \mathbf{u}_{a2} \\ \mathbf{u}_{a3} \end{pmatrix}, \quad (18)$$

of which the solution $\tilde{\mathbf{R}}_x$ is such that

$$vec(\tilde{\mathbf{R}}_x) = \lambda \mathbf{M}^{-1} \begin{pmatrix} \mathbf{u}_{a1} \\ \mathbf{u}_{a2} \\ \mathbf{u}_{a3} \end{pmatrix}. \quad (19)$$

Table 1. Summary of the Results for Two Independent Motions

Motion 1	Translation $\mathbf{R}_B = \mathbf{I}$	Rotation $\mathbf{R}_B \neq \mathbf{I}$	General Motion $\mathbf{R}_B \neq \mathbf{I}$
Motion 2	$\mathbf{t}_B \neq 0$	$\mathbf{t}_B = 0$	$\mathbf{t}_B \neq 0$
Translation $\mathbf{R}_B = \mathbf{I}$ $\mathbf{t}_B \neq 0$	$\mathbf{R}_x, \lambda$	$\mathbf{R}_x, \lambda$ $\mathbf{t}_x(\alpha)$	$\mathbf{R}_x, \lambda$ $\mathbf{t}_x(\alpha)$
Rotation $\mathbf{R}_B \neq \mathbf{I}$ $\mathbf{t}_B = 0$	$\mathbf{R}_x, \lambda$ $\mathbf{t}_x(\alpha)$	$\mathbf{R}_x, \mathbf{t}_x(\lambda)$ Decoupled solution	$\mathbf{R}_x, \mathbf{t}_x(\lambda)$ General solution
General Motion $\mathbf{R}_B \neq \mathbf{I}$ $\mathbf{t}_B \neq 0$	$\mathbf{R}_x, \lambda$ $\mathbf{t}_x(\alpha)$	$\mathbf{R}_x, \mathbf{t}_x(\lambda)$ General solution	$\mathbf{R}_x, \mathbf{t}_x(\lambda)$ General solution

Since $(\mathbf{A}\otimes\mathbf{B})(\mathbf{C}\otimes\mathbf{D}) = \mathbf{AC}\otimes\mathbf{BD}$ (Bellman 1960), it is easy to verify that the analytic form of the inverse of $\mathbf{M}$ is

$$\mathbf{M}^{-1} = \frac{1}{\Delta} \begin{pmatrix} \mathbf{I}_3 \otimes (\mathbf{t}_{b2} \times \mathbf{t}_{b3}) & \mathbf{I}_3 \otimes (\mathbf{t}_{b3} \times \mathbf{t}_{b1}) & \mathbf{I}_3 \otimes (\mathbf{t}_{b1} \times \mathbf{t}_{b2}) \end{pmatrix}, \tag{20}$$

where $\times$ denotes the cross-product and $\Delta = \det(\mathbf{t}_{b1}, \mathbf{t}_{b2}, \mathbf{t}_{b3})$. This allows the rewriting of (19) in closed form:

$$vec(\tilde{\mathbf{R}}_x) = \frac{\lambda}{\Delta} (\mathbf{I}_3 \otimes (\mathbf{t}_{b2} \times \mathbf{t}_{b3})\mathbf{u}_{a1} + \mathbf{I}_3 \otimes (\mathbf{t}_{b3} \times \mathbf{t}_{b1})\mathbf{u}_{a2} + \mathbf{I}_3 \otimes (\mathbf{t}_{b1} \times \mathbf{t}_{b2})\mathbf{u}_{a3}). \tag{21}$$

Applying (11) yields

$$vec(\tilde{\mathbf{R}}_x) = \frac{\lambda}{\Delta} vec\left(\mathbf{u}_{a1}(\mathbf{t}_{b2} \times \mathbf{t}_{b3})^T + \mathbf{u}_{a2}(\mathbf{t}_{b3} \times \mathbf{t}_{b1})^T + \mathbf{u}_{a3}(\mathbf{t}_{b1} \times \mathbf{t}_{b2})^T\right), \tag{22}$$

and from the linearity of the *vec* operator, we finally obtain

$$\tilde{\mathbf{R}}_x = \frac{\lambda}{\Delta}\left(\mathbf{u}_{a1}(\mathbf{t}_{b2} \times \mathbf{t}_{b3})^T + \mathbf{u}_{a2}(\mathbf{t}_{b3} \times \mathbf{t}_{b1})^T + \mathbf{u}_{a3}(\mathbf{t}_{b1} \times \mathbf{t}_{b2})^T\right). \tag{23}$$

Let us analyze this result and prove now that $\tilde{\mathbf{R}}_x$ is equal to $\mathbf{R}_x$, when measurements are exact. To do that, first recall that $\mathbf{R}_x\mathbf{t}_{bi} = \lambda\mathbf{u}_{ai}$. Hence,

$$\tilde{\mathbf{R}}_x = \frac{1}{\Delta}\mathbf{R}_x \underbrace{\left(\mathbf{t}_{b1}(\mathbf{t}_{b2} \times \mathbf{t}_{b3})^T + \mathbf{t}_{b2}(\mathbf{t}_{b3} \times \mathbf{t}_{b1})^T + \mathbf{t}_{b3}(\mathbf{t}_{b1} \times \mathbf{t}_{b2})^T\right)}_{\mathbf{N}}. \tag{24}$$

Recalling that $\mathbf{R}_x$ is orthogonal and verifying that $\mathbf{N} = \Delta\mathbf{I}_3$, we obtain that $\tilde{\mathbf{R}}_x = \mathbf{R}_x$.

This analysis proves that even if λ is unknown the columns of $\tilde{\mathbf{R}}_x$, estimated from (23), are orthogonal to each other. Thus, only the unity constraint (i.e., $\det(\tilde{\mathbf{R}}_x) = 1$) remains to be verified by $\tilde{\mathbf{R}}_x$. From (23) again, the unity constraint immediately gives λ. Consequently, the hand-eye rotation can be recovered from three linearly independent translations. □

PROPOSITION 2. A minimum of two linearly independent pure translations are intrinsically enough to estimate the hand-eye rotation $\mathbf{R}_x$ and the unknown scale factor λ.

Proof. The solution is not linear any more and comes in two steps.

1. Scale factor estimation

   As $\mathbf{R}_x$ is orthogonal, it preserves the norm. Hence, for each pure translation i, we have

   $$\|\mathbf{R}_x\mathbf{t}_{bi}\| = \|\mathbf{t}_{bi}\|.$$

   Applying (15) and (6) on the left-hand side of this expression gives for all i

   $$\lambda\|\mathbf{u}_{ai}\| = \|\mathbf{t}_{bi}\|,$$

   where $\mathbf{u}_{ai}$ and $\mathbf{t}_{bi}$ are known.

2. Hand-eye rotation estimation

Remark that if $\mathbf{t}_{b1}$ and $\mathbf{t}_{b2}$ are two linearly independent vectors, then $\mathbf{t}_{b1} \times \mathbf{t}_{b2}$ is linearly independent from them. Moreover, one can prove that

$$\mathbf{R}_x(\mathbf{t}_{b1} \times \mathbf{t}_{b2}) = (\mathbf{R}_x\mathbf{t}_{b1}) \times (\mathbf{R}_x\mathbf{t}_{b2}).$$

Therefore, we can form the following full-rank (9×9) system:

$$\begin{pmatrix} \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T) \\ \mathbf{I}_3 \otimes ((\mathbf{t}_{b1} \times \mathbf{t}_{b2})^T) \end{pmatrix} vec(\mathbf{R}_x) = \lambda \begin{pmatrix} \mathbf{u}_{a1} \\ \mathbf{u}_{a2} \\ \lambda(\mathbf{u}_{a1} \times \mathbf{u}_{a2}) \end{pmatrix}. \tag{25}$$

Since λ is now known, $\mathbf{R}_x$ can be obtained by inverting this system and the orthogonality of the solution is guaranteed by the proof of Proposition 1.

□

4.2. Pure Rotations

By "pure rotations," we mean motions of the end-effector such that $\mathbf{t}_{bi} = 0$. In practice, these motions can be realized by most of the robotic arms, since the latter are usually designed in such a manner that their end-effector reference frame is centered on a wrist (i.e., the intersection of the last three revolute joint axes). For similar reasons, pan-tilt systems may also benefit from the subsequent analysis.

In such a case, we can state the following proposition:

PROPOSITION 3. If the robot end-effector undergoes $n \geq 2$ pure rotations, two of which having nonparallel axes, then one can linearly estimate the hand-eye rotation $\mathbf{R}_x$ and the hand-eye translation up to the unknown scale factor $\mathbf{t}_x/\lambda$. These two estimations are decoupled.

Notice that in the case where camera motion is obtained through pose computation, λ is known and the hand-eye translation can thus be fully recovered, as did Li (1998).

Proof. With pure rotations, the system in (14) is block-diagonal and decouples into

$$(\mathbf{I}_9 - \mathbf{R}_{ai} \otimes \mathbf{R}_{bi})\, vec(\mathbf{R}_x) = \mathbf{0}_{9\times 1}, i = 1..n \tag{26}$$

$$(\mathbf{I}_3 - \mathbf{R}_{ai})\, \mathbf{t}_x = \lambda \mathbf{u}_{ai}, i = 1..n. \tag{27}$$

Let us first study eq. (27). Consider $n \geq 2$ rotations, such that, without loss of generality, $\mathbf{R}_{a1}$ and $\mathbf{R}_{a2}$ have nonparallel axes, and note $\mathbf{t}_x = \lambda \mathbf{t}_{x0}$. Then, from (27), we form the following system:

$$\begin{pmatrix} \mathbf{I}_3 - \mathbf{R}_{a1} \\ \vdots \\ \mathbf{I}_3 - \mathbf{R}_{an} \end{pmatrix} \lambda \mathbf{t}_{x0} = \lambda \begin{pmatrix} \mathbf{u}_{a1} \\ \vdots \\ \mathbf{u}_{an} \end{pmatrix}. \tag{28}$$

This system has a one-dimensional solution subspace. Indeed, $\mathbf{t}_{x0}$ is solution to the following system:

$$\begin{pmatrix} \mathbf{I}_3 - \mathbf{R}_{a1} \\ \vdots \\ \mathbf{I}_3 - \mathbf{R}_{an} \end{pmatrix} \mathbf{t}_{x0} = \begin{pmatrix} \mathbf{u}_{a1} \\ \vdots \\ \mathbf{u}_{an} \end{pmatrix}, \tag{29}$$

which has full rank because $\mathbf{R}_{a1}$ and $\mathbf{R}_{a2}$ have nonparallel axes.

Notice that if $n > 2$, $\mathbf{t}_{x0}$ is the least squares solution of this system. Notice also that the parameter of the solution subspace is the unknown scale factor. This is not surprising since pure rotations of the robot do not contain metric information.

Let us now study the first subsystem (26). One of the properties of the Kronecker product is that the eigenvalues of $\mathbf{M} \otimes \mathbf{N}$ are the product of the eigenvalues of $\mathbf{M}$ by those of $\mathbf{N}$. In our case, $\mathbf{R}_{ai}$ and $\mathbf{R}_{bi}$ have the same eigenvalues: $\{1, e^{i\theta_i}, e^{-i\theta_i}\}$ and thus the eigenvalues of $\mathbf{R}_{ai} \otimes \mathbf{R}_{bi}$ are $\{1, 1, 1, e^{i\theta_i}, e^{i\theta_i}, e^{-i\theta_i}, e^{-i\theta_i}, e^{2i\theta_i}, e^{-2i\theta_i}\}$.

Consequently, when the angle of rotation θ_i is not a multiple of π, then the (9×9) matrix of (26) $\mathbf{I}_9 - \mathbf{R}_{ai} \otimes \mathbf{R}_{bi}$ has rank 6. Hence, the solution $\mathbf{R}_x$ lies in a three-dimensional manifold. Using the orthogonality constraints of $\mathbf{R}_x$, the solution manifold dimension can only be reduced to 1. This dimension corresponds to the common eigenvalue 1 of $\mathbf{R}_{ai}$ and $\mathbf{R}_{bi}$. This confirms the need for two rotations.

In the case of two or more independent rotations, we can state the following lemma:

LEMMA 1. If the robot end-effector undergoes $n \geq 2$ pure rotations, two of which having nonparallel axes, then system (26) has rank 8, its null space $\mathcal{K}$ is one-dimensional, and the hand-eye rotation $\mathbf{R}_x$ is equal to

$$\mathbf{R}_x = \frac{sign(\det(\mathbf{V}))}{|\det(\mathbf{V})|^{\frac{1}{3}}}\mathbf{V}, \tag{30}$$

where $sign()$ returns the sign of its argument, $\mathbf{V} = vec^{-1}(\mathbf{v})$ and $\mathbf{v}$ is any vector of the null space $\mathcal{K}$.

The proof of this Lemma is technical and, hence, given in Appendix A. In this proof, we determine the null-space of the matrix

$$\begin{pmatrix} \mathbf{I}_9 - \mathbf{R}_{a1} \otimes \mathbf{R}_{b1} \\ \mathbf{I}_9 - \mathbf{R}_{a2} \otimes \mathbf{R}_{b2} \\ \vdots \\ \mathbf{I}_9 - \mathbf{R}_{an} \otimes \mathbf{R}_{bn} \end{pmatrix},$$

where, without loss of generality, $\mathbf{R}_{a1}$ and $\mathbf{R}_{a2}$ are the two rotations with nonparallel axes. We show that in the noise-free case, any matrix $\mathbf{V}$ extracted for the null space $\mathcal{K}$ is guaranteed to have perpendicular rows, even if more than the minimal two rotations with nonparallel axes are used. □

In practice, $\mathbf{v}$ can be determined using a singular value decomposition (SVD), which is known to accurately estimate the null space of a linear mapping. Nevertheless, noise

yields a nonorthogonal matrix, which requires correction by an orthogonalization step (Horn, Hilden, and Negahdaripour 1988). However, even though it has not been quantified, the results we obtained before orthogonalization are close to being orthogonal.

4.3. Planar Motions

Some robots are restricted to move on a plane, such as car-like robots. In this case, all the robot and camera rotations have the same axis $\mathbf{n}_b$ (resp. $\mathbf{n}_a = \mathbf{R}_x\mathbf{n}_b$), which is orthogonal to the plane of motion. Then, we can demonstrate that

LEMMA 2. One planar motion with nonidentity rotation and one nonzero pure translation (which is not parallel to the rotation axis of the planar motion, see Fig. 7) are intrinsically enough to recover the hand-eye rotation $\mathbf{R}_x$ and the unknown scale factor λ. The hand-eye translation can only be estimated up to an unknown height α along the normal to the camera plane of motion (Fig. 8).

Notice that this lemma is not limited to the planar motion case, since the pure translation is not restricted to lie in the plane of motion.

Proof. Assume without loss of generality that the first motion is a pure translation ($\mathbf{R}_{a1} = \mathbf{R}_{b1} = \mathbf{I}_3$, $\mathbf{t}_{b1} \in \Re^3$) and the second is a planar motion with nonidentity rotation such that its rotation axis $\mathbf{n}_b$ is not parallel to $\mathbf{t}_{b1}$ (Fig. 7). Then, the general system (14) rewrites as

$$\begin{pmatrix} \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) & \mathbf{0}_{3\times 3} & -\mathbf{u}_{a1} \\ \mathbf{I}_9 - \mathbf{R}_{a2} \otimes \mathbf{R}_{b2} & \mathbf{0}_{9\times 3} & \mathbf{0}_{9\times 1} \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T) & \mathbf{I}_3 - \mathbf{R}_{a2} & -\mathbf{u}_{a2} \end{pmatrix} \begin{pmatrix} vec(\mathbf{R}_x) \\ \mathbf{t}_x \\ \lambda \end{pmatrix} = \mathbf{0}_{15\times 1}, \tag{31}$$

which is equivalent to the following two equations:

$$\begin{pmatrix} \mathbf{I}_9 - \mathbf{R}_{a2} \otimes \mathbf{R}_{b2} & \mathbf{0}_{9\times 1} \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) & -\mathbf{u}_{a1} \end{pmatrix} \begin{pmatrix} vec(\mathbf{R}_x) \\ \lambda \end{pmatrix} = \mathbf{0}_{12\times 1} \tag{32}$$

$$(\mathbf{I}_3 - \mathbf{R}_{a2})\mathbf{t}_x = -\lambda\mathbf{u}_{a2} - \left(\mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T)\right) vec(\mathbf{R}_x). \tag{33}$$

The solution comes in three steps:

1. Scale factor estimation

   As in the proof of Proposition 2.

2. Hand-eye rotation estimation

   Recall that the camera axis of rotation $\mathbf{n}_a$ and the robot axis of rotation $\mathbf{n}_b$ are related by

   $$\mathbf{R}_x\mathbf{n}_b = \mathbf{n}_a,$$

   which is similar to (15). Since $\mathbf{t}_{b1}$ and $\mathbf{n}_b$ are assumed to be nonparallel, they are linearly independent. Therefore, we obtain, as in the proof of Proposition 2, a full-rank (9×9) system where $\mathbf{R}_x$ is the only unknown:

   $$\begin{pmatrix} \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) \\ \mathbf{I}_3 \otimes (\mathbf{n}_b^T) \\ \mathbf{I}_3 \otimes ((\mathbf{t}_{b1} \times \mathbf{n}_b)^T) \end{pmatrix} vec(\mathbf{R}_x) = \begin{pmatrix} \lambda\mathbf{u}_{a1} \\ \mathbf{n}_a \\ \lambda(\mathbf{u}_{a1} \times \mathbf{n}_a) \end{pmatrix}. \tag{34}$$

3. Hand-eye translation estimation

   We can insert the estimated $\mathbf{R}_x$ and λ into (33) and obtain a system, where only $\mathbf{t}_x$ is unknown. This system is always underconstrained. Hence, it admits as solution any vector of the form

   $$\mathbf{t}_x(\alpha) = \mathbf{t}_\perp + \alpha\mathbf{n}_a, \tag{35}$$

   where α is any scalar value and $\mathbf{t}_\perp$ is a solution in the plane of the camera motion. The latter vector is unique since $\mathbf{I}_3 - \mathbf{R}_{a1}$ has rank 2 and the plane of motion is two-dimensional. In practice, $\mathbf{t}_\perp$ can be obtained by an SVD of $\mathbf{I}_3 - \mathbf{R}_{a1}$ (Press et al. 1992, Section 2.6).

□

The previous lemma serves as a basis to the case of planar motions as

PROPOSITION 4. Two planar motions allow the estimation of the hand-eye rotation $\mathbf{R}_x$ and the unknown scale factor λ if one of the following three sets of conditions is fulfilled:

- the two motions are linearly independent pure translations
- one of the two motions is a nonzero pure translation
- the two motions contain a nonidentity rotation and

$$(\mathbf{I}_3 - \mathbf{R}_{b2})\mathbf{t}_{b1} - (\mathbf{I}_3 - \mathbf{R}_{b1})\mathbf{t}_{b2} \neq 0.$$

In the last two cases, the hand-eye translation can only be estimated up to an unknown height α along the normal to the camera plane of motion (Fig. 8).

Proof. The first set of conditions falls back into the pure translation case, and Proposition 2 applies. The second set of conditions is contained in Lemma 2.

Let us now show that the last set of conditions can be brought back to the second one. To do that, consider the system that is built upon the two planar motions:

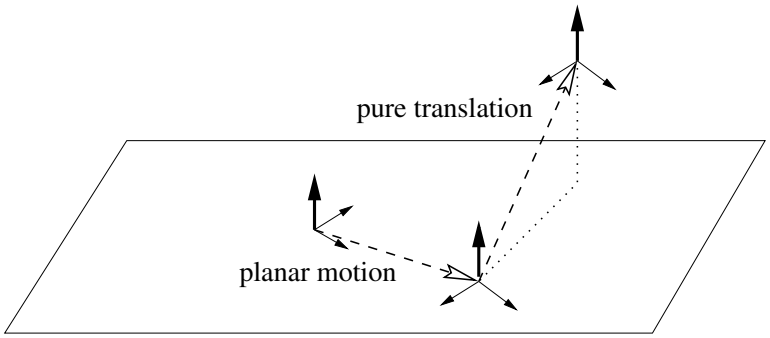

Fig. 7. One planar motion with nonidentity rotation and one nonzero pure translation that is not parallel to the rotation axis of the planar motion.

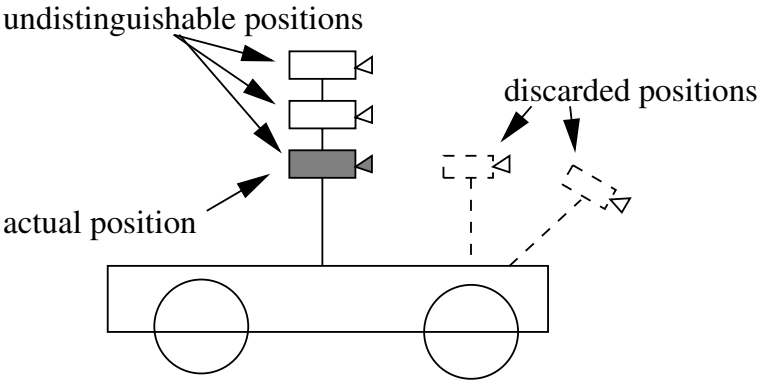

Fig. 8. In the case of planar motions, one cannot determine the altitude of a camera that is rigidly mounted onto the base.

$$\begin{matrix} L_1 \to \\ L_2 \to \\ L_3 \to \\ L_4 \to \end{matrix} \begin{pmatrix} \mathbf{I}_9 - \mathbf{R}_{a1} \otimes \mathbf{R}_{b1} & \mathbf{0}_{9\times 3} & \mathbf{0}_{9\times 1} \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) & \mathbf{I}_3 - \mathbf{R}_{a1} & -\mathbf{u}_{a1} \\ \mathbf{I}_9 - \mathbf{R}_{a2} \otimes \mathbf{R}_{b2} & \mathbf{0}_{9\times 3} & \mathbf{0}_{9\times 1} \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T) & \mathbf{I}_3 - \mathbf{R}_{a2} & -\mathbf{u}_{a2} \end{pmatrix} \begin{pmatrix} vec(\mathbf{R}_x) \\ \mathbf{t}_x \\ \lambda \end{pmatrix} = \mathbf{0}_{15\times 1}. \tag{36}$$

The block line L_1 and the third one L_3 of this system are equivalent since both motions have the same rotation axis. Hence, we can discard the first one and obtain

$$\begin{matrix} L'_1 \to \\ L'_2 \to \\ L'_3 \to \end{matrix} \begin{pmatrix} \mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T) & \mathbf{I}_3 - \mathbf{R}_{a1} & -\mathbf{u}_{a1} \\ \mathbf{I}_9 - \mathbf{R}_{a2} \otimes \mathbf{R}_{b2} & \mathbf{0}_{9\times 3} & \mathbf{0}_{9\times 1} \\ \mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T) & \mathbf{I}_3 - \mathbf{R}_{a2} & -\mathbf{u}_{a2} \end{pmatrix} \begin{pmatrix} vec(\mathbf{R}_x) \\ \mathbf{t}_x \\ \lambda \end{pmatrix} = \mathbf{0}_{15\times 1}. \tag{37}$$

Consider now the linear combination $(\mathbf{I}_3 - \mathbf{R}_{a2})L'_1 - (\mathbf{I}_3 - \mathbf{R}_{a1})L'_3$, which gives

$$\big[(\mathbf{I}_3 - \mathbf{R}_{a2})\big(\mathbf{I}_3 \otimes (\mathbf{t}_{b1}^T)\big) - (\mathbf{I}_3 - \mathbf{R}_{a1})\big(\mathbf{I}_3 \otimes (\mathbf{t}_{b2}^T)\big)\big] vec(\mathbf{R}_x) \tag{38}$$

$$+\big[(\mathbf{I}_3 - \mathbf{R}_{a2})(\mathbf{I}_3 - \mathbf{R}_{a1}) - (\mathbf{I}_3 - \mathbf{R}_{a1})(\mathbf{I}_3 - \mathbf{R}_{a2})\big]\mathbf{t}_x \tag{39}$$

$$-\lambda\big[(\mathbf{I}_3 - \mathbf{R}_{a2})\mathbf{u}_{a1} - (\mathbf{I}_3 - \mathbf{R}_{a1})\mathbf{u}_{a2}\big] = 0. \tag{40}$$

As $\mathbf{R}_{a1}$ and $\mathbf{R}_{a2}$ have the same rotation axis, they commute and, hence, $(\mathbf{I}_3 - \mathbf{R}_{a2})(\mathbf{I}_3 - \mathbf{R}_{a1}) - (\mathbf{I}_3 - \mathbf{R}_{a1})(\mathbf{I}_3 - \mathbf{R}_{a2}) = 0$. Therefore, the term on line (39) is null. As for the term on line (40), let us denote it as $\mathbf{u}'_{a1}$.

Let us now consider the first term (38) and show that it can be rewritten under the form $\mathbf{R}_x \mathbf{t}'^T_{b1}$. To do that, recall that $(\mathbf{I}_3 \otimes \mathbf{t}_{bi}^T) vec(\mathbf{R}_x) = \mathbf{R}_x \mathbf{t}_{bi}$. Hence, the first term equals

$$(\mathbf{I}_3 - \mathbf{R}_{a2})\mathbf{R}_x \mathbf{t}_{b1} - (\mathbf{I}_3 - \mathbf{R}_{a1})\mathbf{R}_x \mathbf{t}_{b2}.$$

Using $\mathbf{R}_{ai}\mathbf{R}_x = \mathbf{R}_x\mathbf{R}_{bi}$, we then obtain

$$\mathbf{R}_x\big(\underbrace{(\mathbf{I}_3 - \mathbf{R}_{b2})\mathbf{t}_{b1} - (\mathbf{I}_3 - \mathbf{R}_{b1})\mathbf{t}_{b2}}_{\mathbf{t}'_{b1}}\big).$$

Consequently, L'_1 is equivalent to

$$\mathbf{R}_x\mathbf{t}'_{b1} = \lambda\mathbf{u}'_{a1},$$

where we recognize the pure translation case. Hence, system (36) rewrites under the same form as in (31) of Lemma 2. Therefore, a solution exists if the virtual robot pure translation $\mathbf{t}'_{b1}$ is not parallel to $\mathbf{n}_b$. As both $\mathbf{t}_{b1}$ and $\mathbf{t}_{b2}$ are orthogonal to $\mathbf{n}_b$, this condition reduces to a nonzero condition on $\mathbf{t}'_{b1}$, which is expressed as

$$(\mathbf{I}_3 - \mathbf{R}_{b2})\mathbf{t}_{b1} - (\mathbf{I}_3 - \mathbf{R}_{b1})\mathbf{t}_{b2} \neq 0.$$

□

In conclusion, we exhibited sufficient conditions to obtain, from two planar motions, the hand-eye rotation and the hand-eye translation, up to a component perpendicular to the camera plane of motion. In the case of a car, this unknown component can be interpreted as a height with respect to the base of the car (Fig. 8).

4.4. The General Case

In the case of two independent general motions with nonparallel axes, there exists a unique solution to the hand-eye calibration problem. We obtain the same result for our hand-eye self-calibration problem:

PROPOSITION 5. If the robot end-effector undergoes two independent general motions with nonparallel axes, then the hand-eye transformation $(\mathbf{R}_x, \mathbf{t}_x)$ can be fully recovered, as well as the Euclidean reconstruction unknown scale factor λ.

Using our formulation, one possibility to solve the whole system in (14) is to find its null space, which is a subspace of $\Re^{13}$. The latter subspace must be one-dimensional and only depend on λ, according to the sufficient condition for hand-eye calibration. Hence, the solution to hand-eye self-calibration is a 13×1 vector to be found in a one-dimensional subspace. It can therefore be extracted from this null space by applying the unity constraint to the first nine coefficients representing the hand-eye rotation, as seen in the pure translation case.

However, Wei, Arbter, and Hirzinger (1989) remarked, in the case where camera motions are obtained through pose computation, that the accuracy of the simultaneous estimation of hand-eye rotation and translation is not independent of the physical unit used for the translation. By analogy with this remark, solving directly for the whole system may yield the same dependence. In addition, such a solution does not guarantee that the estimated $\mathbf{R}_x$ is an orthogonal matrix. Then, one has to perform a correction of the result by applying the orthogonality constraint. However, this correction is nonlinear in essence and it is hence improbable to find the corresponding correction on the hand-eye translation estimation.

On the opposite, a two-step solution, as in Tsai and Lenz (1989), guarantees an orthogonal estimate of the hand-eye rotation. Indeed, the first step consists in the linear estimation of the hand-eye rotation as in the case of pure rotations (26), which had this property. Recall that the solution is given by estimating the nullspace of

$$\begin{pmatrix} \mathbf{I}_3 - \mathbf{R}_{a1} \otimes \mathbf{R}_{b1} \\ \vdots \\ \mathbf{I}_3 - \mathbf{R}_{an} \otimes \mathbf{R}_{bn} \end{pmatrix} vec(\mathbf{R}_x) = 0, \tag{41}$$

and, then, by proceeding to an orthogonalization step to counterbalance the effect of noise.

This part of the solution is, in essence, not very different from the existing approaches, but, here again, we want to point out that our goal is to give a generic formulation in which the particular cases and the general case can be unified.

As for the second step, it exploits the remaining lines in (14):

$$\begin{pmatrix} \mathbf{I}_3 - \mathbf{R}_{a1} & -\mathbf{u}_{a1} \\ \vdots & \\ \mathbf{I}_3 - \mathbf{R}_{an} & -\mathbf{u}_{an} \end{pmatrix} \begin{pmatrix} \mathbf{t}_x \\ \lambda \end{pmatrix} = \begin{pmatrix} -\mathbf{R}_x\mathbf{t}_{b1} \\ \vdots \\ -\mathbf{R}_x\mathbf{t}_{bn} \end{pmatrix}. \tag{42}$$

As soon as two general motions with nonparallel axes are used, this system has full rank. It thus yields a linear least squares solution to the hand-eye translation and the scale factor.

5. Experiments

In this section, we will first choose a distance to measure the errors between rigid transformations since their group $SE(3)$ does not hold an intrinsic metric (Murray, Li, and Sastry 1994). Second, we will show some simulation results to test the robustness to noise of our method, compared to the reference methods. Finally, we will give experimental results in real conditions. Notice that more experimental results can be found in Andreff (1999).

In this section, we numbered the methods we compared as follows: axis/angle method (Tsai and Lenz 1989) (**M1**), dual quaternion method (Daniilidis and Bayro-Corrochano 1996) (**M2**), nonlinear minimization (Horaud and Dornaika 1995) (**M3**), our linear formulation adapted to the case where camera motions are obtained through pose computation (**M4**), and self-calibration (**M5**).

5.1. Error Measurement

To measure the errors in translation, we chose the usual relative error in $\Re^3$: $\|\hat{\mathbf{t}} - \mathbf{t}\|/\|\mathbf{t}\|$, where the $\hat{}$ notation represents the estimated value.

For the errors in orientation, no canonical measure is defined. We chose the quaternion norm used in Daniilidis and Bayro-Corrochano (1996): $\|\hat{\mathsf{q}} - \mathsf{q}\|$ for its simplicity and its direct relation to α, the angle of the residual rotation between these two orientations. Indeed, if $\hat{\mathsf{q}}$ and q are unitary, then $\|\hat{\mathsf{q}} - \mathsf{q}\| = 2 - 2\cos\frac{\alpha}{2}$. It is thus strictly increasing from 0 to 4 as α goes from 0 to 2π. Moreover, this metric avoids the singularity in $\alpha = \pi$ appearing when using geodesics (Samson, Le Borgne, and Espiau 1991, p. 35).

5.2. Simulations

We first performed simulations to gain some insight of the numerical behavior of our linear method (**M4**) with comparison to the reference methods (**M1–M3**). We thus tested the robustness of the methods to noise and their accuracy with respect to the number of calibration motions in use.

5.2.1. Simulation Procedure

For each simulation series and for each value of the parameter of interest (noise, number of motions), we followed the same methodology. First, we defined a hand-eye transformation by random choice of the roll-pitch-yaw angles of its rotation matrix as well as of the coefficients of its translation vector, according to Gaussian laws. Second, we similarly chose a sequence of robot motions and defined, from it and the hand-eye transformation, the corresponding camera motion sequence. Third, we added noise to the camera motions (see below). Finally, we performed hand-eye calibration with the various methods and compared their results to the initial hand-eye transformation.

5.2.2. Inserting Noise

We added noise to the camera translations $\mathbf{t}_{A_i}$ by defining $\tilde{\mathbf{t}}_{A_i} = \mathbf{t}_{A_i} + \nu\|\mathbf{t}_{A_i}\|\mathbf{n}$, where ν is a scalar and $\mathbf{n}$ is a Gaussian 3-vector with zero mean and unit variance (white noise). As for the camera rotations, we added noise to their roll-pitch-yaw angles as $\tilde{\alpha} = (1 + \nu r)\alpha$, where α is any of these angles, ν is the same as for the translation, and r is a one-dimensional white noise. Hence, ν defines a signal-to-noise ratio.

5.2.3. Robustness to Noise

We tested for the value of ν, making it vary from 0% to 20% in two simulation series. In the first one, we made 100 different choices of hand-eye transformations and motion sequences for each noise level. These sequences contained only two motions, with maximal amplitude of 1 m in translation and 180 deg in rotation. Figure 9 gathers the calibration errors. It shows that Tsai and Lenz's (1989) method (**M1**) and ours (**M4**) obtain the highest accuracy in rotation. For translations, they are very powerful as long as the noise level is low but are less accurate than the dual quaternion method (**M2**) or the nonlinear minimization method (**M3**) when the noise level increases.

In a second simulation series, we almost repeated the first one, just reducing the amplitude of the calibration motions to 2 cm in translation and 10 deg in rotation. The results (Fig. 10) show that our linear formulation is less sensitive to this reduction than the other methods.

5.2.4. Influence of Motion Number

In this experiment, we kept the noise level constant ($\nu = 0.01$) and generated sequences of varying length, i.e., from 2 to 15 calibration motions. Their amplitude was chosen to be small (1 cm in translation and 10 deg in rotation). For each sequence length, we proceeded to 100 random choices of hand-eye transformations and calibration motions. The results (Fig. 11) show here again a higher accuracy for our linear formulation.

5.3. Experiments on Real Data

When dealing with real data, no ground-truth value is available for comparison. Therefore, we defined two measures of accuracy. The first one compares, for each motion i, $\mathbf{A}_i\mathbf{X}$ and $\mathbf{X}\mathbf{B}_i$. We then gathered all these errors into RMS errors. The second one is a relative error with respect to a reference estimation of the hand-eye transformation, computed by a RANSAC method as follows. From a set of recorded 33 reference positions, we chose 100 sets of 11 positions (10 motions) among the 33 available; then we computed from them 100 estimations of the hand-eye transformation; and finally we computed the "mean" rigid transformation of these estimations (see Andreff 1999 for details).

The following two experiments follow the same procedure. First, a single trajectory is recorded. Second, camera and robot motions are estimated. Third, the hand-eye transformation is estimated using several methods. Our linear methods are used in their original form, without any RANSAC method. Finally, each estimation is measured as described above.

5.3.1. Experiment 1

To evaluate the correctness of the solution obtained by hand-eye self-calibration, we had to compare it with those obtained by classical calibration methods with the same data.

Hence, we took images of our calibration grid (Fig. 12) and performed hand-eye calibration with the axis/angle method (Tsai and Lenz 1989) (**M1**), the dual quaternion method (Daniilidis and Bayro-Corrochano 1996) (**M2**), the nonlinear minimization (Horaud and Donaika 1995) (**M3**), and the linear formulation (**M4**). Finally, using the same points, extracted from the images of the calibration grid, but not their 3D model, we applied the hand-eye self-calibration method (**M5**). The Euclidean 3D reconstruction method we used is the one proposed in Christy and Horaud (1996).

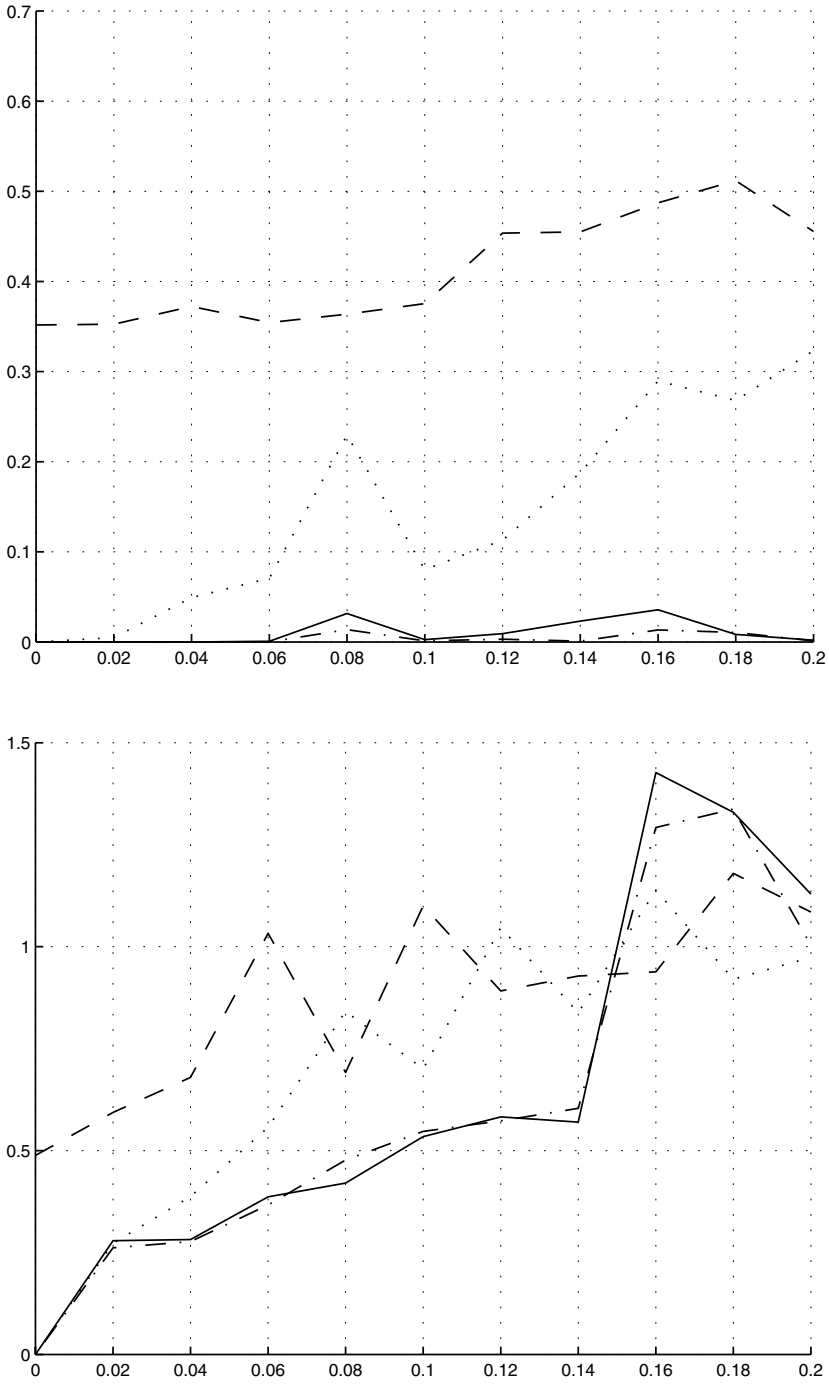

Fig. 9. Rotation (top) and translation (bottom) relative calibration errors with respect to noise level: **M1** (—), **M2** (· · ·), **M3** (- -), **M4** (— ·).

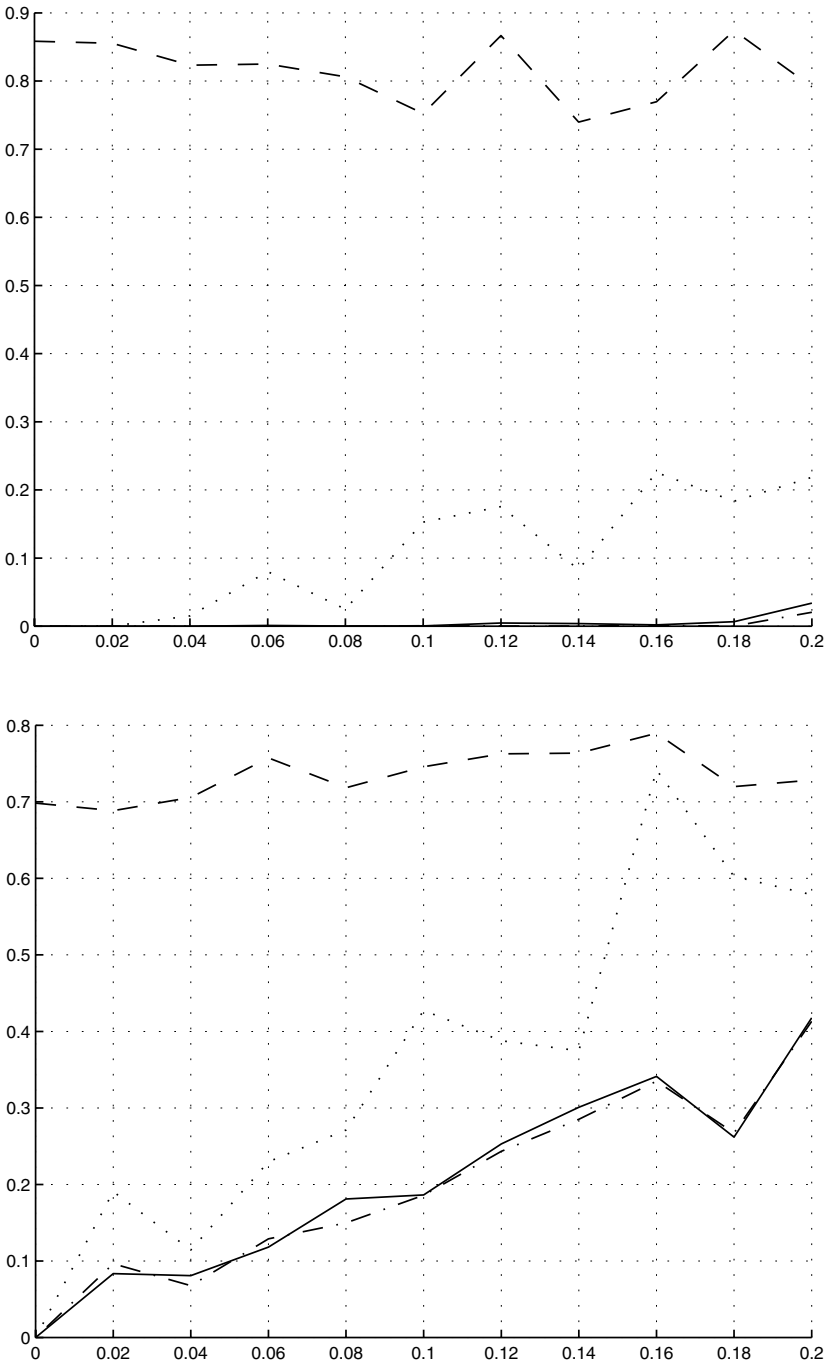

Fig. 10. Calibration errors with respect to noise level using small motions (same conventions as in Fig. 9).

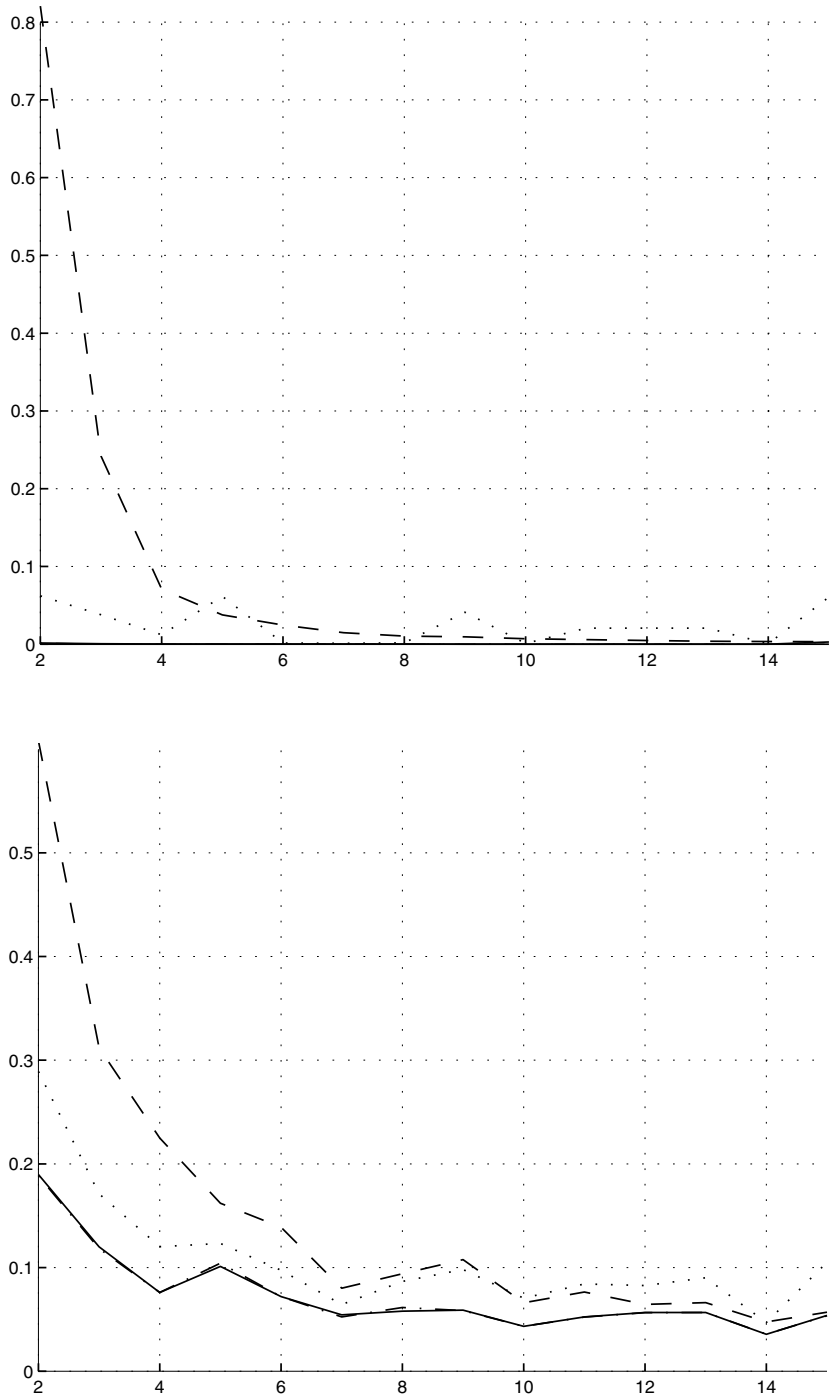

Fig. 11. Calibration errors with respect to the number of calibration motions using small motions (same conventions as in Fig. 9).

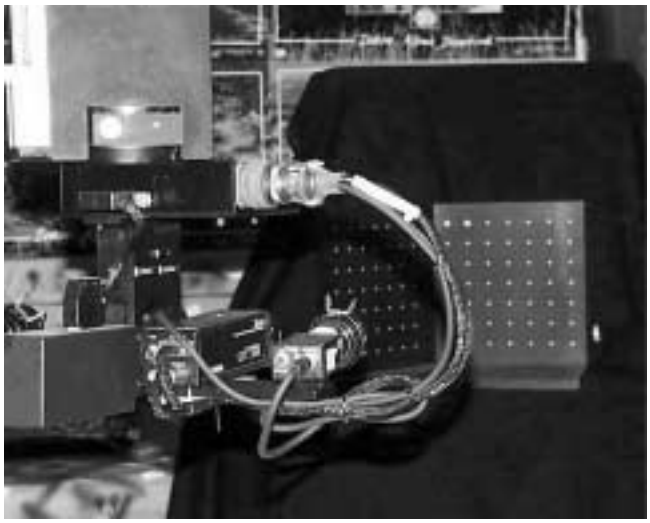

Fig. 12. In Experiment 1, the camera observes a calibration grid.

Table 2. Comparison with a Reference Estimation of the Hand-Eye Transformation

Method	Rotation Error	Translation Error
M1	$1.10.10^{-5}$	0.018
M2	$1.61.10^{-5}$	0.096
M3	$9.77.10^{-5}$	0.149
M4	$0.06.10^{-5}$	0.023
M5	$1.99.10^{-5}$	0.322

In this experiment, we defined the trajectory by ordering the 33 reference positions so that they were as far as possible from each other according to the advice given in Tsai and Lenz (1989). The RMS errors between $\mathbf{A}_i\mathbf{X}$ and $\mathbf{X}\mathbf{B}_i$ are displayed in Figure 13. It can be seen that (**M4**) gives the smallest error in rotation due to the numerical efficiency of the SVD and thus obtains also a reduced error in translation. As for (**M5**), it gives larger errors, as expected since the 3D model is not used. However, the degradation is rather small and can be explained by a rough estimation of the intrinsic parameters.

Then, we compared the estimations obtained above to the reference estimation. We gathered the errors in Table 2. It confirms that the linear method is numerically very efficient, especially as far as rotation is concerned. Moreover, the self-calibration method yields a lower accuracy, which nevertheless remains acceptable in the context of visual servoing (Espiau 1993).

5.3.2. Experiment 2

In a second experiment, we tested (**M5**) with more realistic images. Four positions were defined where the images shown in Figure 14 were taken. In the first image, points were extracted and then tracked during the motion between each position of the camera. Then, hand-eye self-calibration was performed on the tracked points.

Table 3. Comparison with a Reference Estimation of the Hand-Eye Transformation

Method	Rotation Error	Translation Error
M1	$2.7.10^{-5}$	0.18
M2	$2.8.10^{-5}$	0.22
M3	1.82	1.01
M4	$2.3.10^{-5}$	0.17
M5	$2.8.10^{-4}$	0.20

In a goal of comparison, the blocks were replaced by the calibration grid and the robot was moved anew to the four predefined positions. Then, hand-eye calibration was performed with the images taken there.

The RMS errors between $\mathbf{A}_i\mathbf{X}$ and $\mathbf{X}\mathbf{B}_i$ in this experiment are given in Figure 15. They show an awful behavior of the nonlinear minimization method, probably due to the small number of data. They also show a slightly higher degradation of the performance of (**M5**) compared to the others. Nevertheless, it remains in an acceptable ratio since the relative error in translation is close to 3%.

We also compared the results obtained in this experiment to the reference estimation (Table 3). This comparison confirms the accuracy of both the linear method and the self-calibration scheme.

6. Conclusion

We proposed a hand-eye self-calibration method that reduces the human supervision compared with classical calibration methods. The cost of releasing the human constraint is a small degradation of the numerical accuracy. However, the obtained precision is good enough in the context of visual servoing.

This method is based on the structure-from-motion paradigm, rather than pose estimation, to compute the camera motions, and its derivation includes a new linear formulation of hand-eye calibration. The linearity of the formulation allows a simple algebraic analysis. Thus, we determined the parts of the hand-eye transformation that can be obtained from a reduced number of motions that does not allow a complete calibration. Moreover, the linear formulation provides improved numerical accuracy even in the case where the camera/robot rotations have small amplitude.

However, one difficulty with the Euclidean 3D reconstruction with a moving camera is to be able to find reliable point correspondences between images. The method proposed in Christy and Horaud (1996) solves this problem by tracking points along the motion. However, it requires that the points

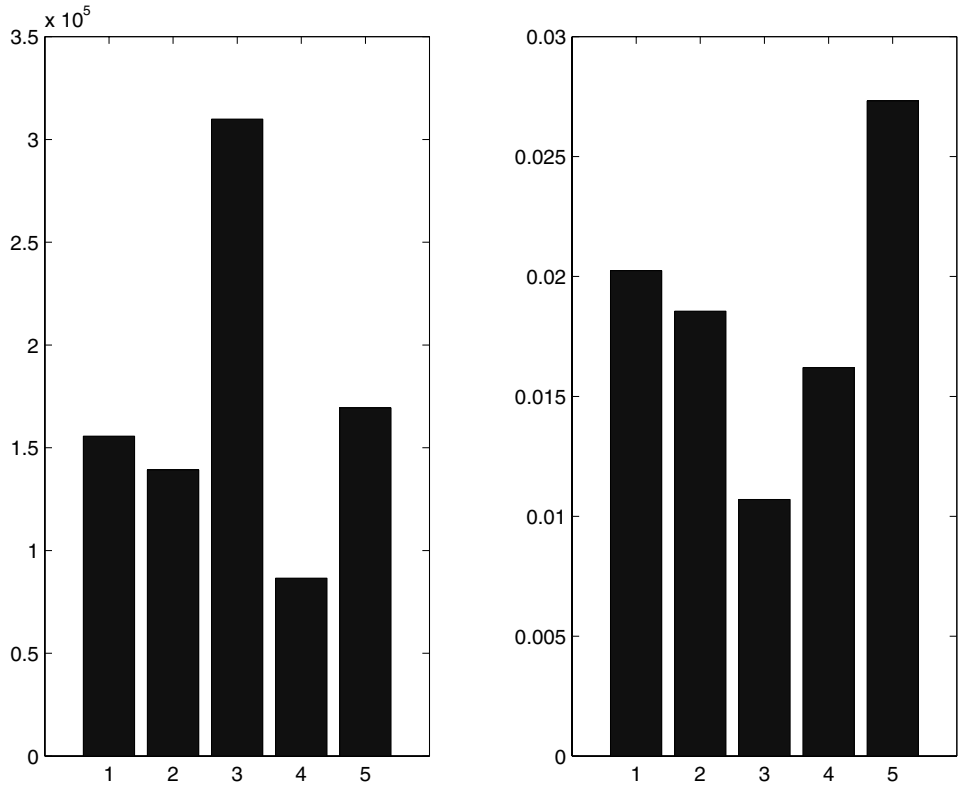

Fig. 13. RMS errors in rotation (left) and translation (right) with 33 images of a calibration grid for each method (see text).

are tracked from the beginning until the end of the robot trajectory. This is a hard constraint since, in practice, one hardly obtains enough points after a long trajectory.

Stereo vision may offer the answer to this problem since it was shown that Euclidean reconstruction can be performed, without any prior knowledge, from two Euclidean motions of a stereo pair (Devernay and Faugeras 1996). This is fully in coherence with our constraints. Moreover, this kind of method releases the constraint on the presence of points along the whole sequence of images. Exploiting this idea, Ruf et al. (1999) proposed a method for projective robot kinematic calibration and its application to visual servoing.

Finally, there is a pending question that was never answered: "What are the motions for hand-eye (self-)calibration that yield the higher numerical accuracy?"

Appendix A: Proof of Lemma 1

A.1. Preliminary Results

PRELIMINARY RESULT 1. Given two similar rotation matrices $\mathbf{R}$ and $\mathbf{R}'$ (i.e., there exists a rotation matrix $\mathbf{R}_x$ such that $\mathbf{R}' = \mathbf{R}_x\mathbf{R}\mathbf{R}_x^T$), then

1) if $\mathbf{v}$ is an eigenvector of $\mathbf{R} \otimes \mathbf{R}'$, then $(\mathbf{I} \otimes \mathbf{R}_x^T)\mathbf{v}$ is an eigenvector of $\mathbf{R} \otimes \mathbf{R}$ for the same eigenvalue;

2) if $\mathbf{x}$ is an eigenvector of $\mathbf{R} \otimes \mathbf{R}$, then $(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x}$ is an eigenvector of $\mathbf{R} \otimes \mathbf{R}'$ for the same eigenvalue.

Proof.

1) Let $\mathbf{v}$ be an eigenvector of $\mathbf{R} \otimes \mathbf{R}'$ with eigenvalue λ. Then, $(\mathbf{R} \otimes \mathbf{R}')\mathbf{v} = \lambda\mathbf{v}$. Replacing $\mathbf{R}'$ by $\mathbf{R}_x\mathbf{R}\mathbf{R}_x^T$ in this relation gives

$$(\mathbf{R} \otimes \mathbf{R}_x\mathbf{R}\mathbf{R}_x^T)\mathbf{v} = \lambda\mathbf{v}.$$

From $(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD})$ (Bellman 1960), we obtain

$$(\mathbf{I} \otimes \mathbf{R}_x)(\mathbf{R} \otimes \mathbf{R})(\mathbf{I} \otimes \mathbf{R}_x^T)\mathbf{v} = \lambda\mathbf{v}.$$

As $(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}$ (Bellman 1960), we derive the following relation:

$$(\mathbf{I} \otimes \mathbf{R}_x^T)^{-1}(\mathbf{R} \otimes \mathbf{R})(\mathbf{I} \otimes \mathbf{R}_x^T)\mathbf{v} = \lambda\mathbf{v}.$$

Hence, $(\mathbf{R} \otimes \mathbf{R})(\mathbf{I} \otimes \mathbf{R}_x^T)\mathbf{v} = \lambda(\mathbf{I} \otimes \mathbf{R}_x^T)\mathbf{v}$.

2) Let $\mathbf{x}$ be an eigenvector of $\mathbf{R} \otimes \mathbf{R}$ with eigenvalue α. Then,

$$(\mathbf{R} \otimes \mathbf{R})\mathbf{x} = \alpha\mathbf{x}.$$

As $(\mathbf{I} \otimes \mathbf{R}_x^T)(\mathbf{I} \otimes \mathbf{R}_x) = \mathbf{I}$, we can insert it on both sides:

$$(\mathbf{R} \otimes \mathbf{R})(\mathbf{I} \otimes \mathbf{R}_x^T)(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x} = \alpha(\mathbf{I} \otimes \mathbf{R}_x^T)(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x},$$

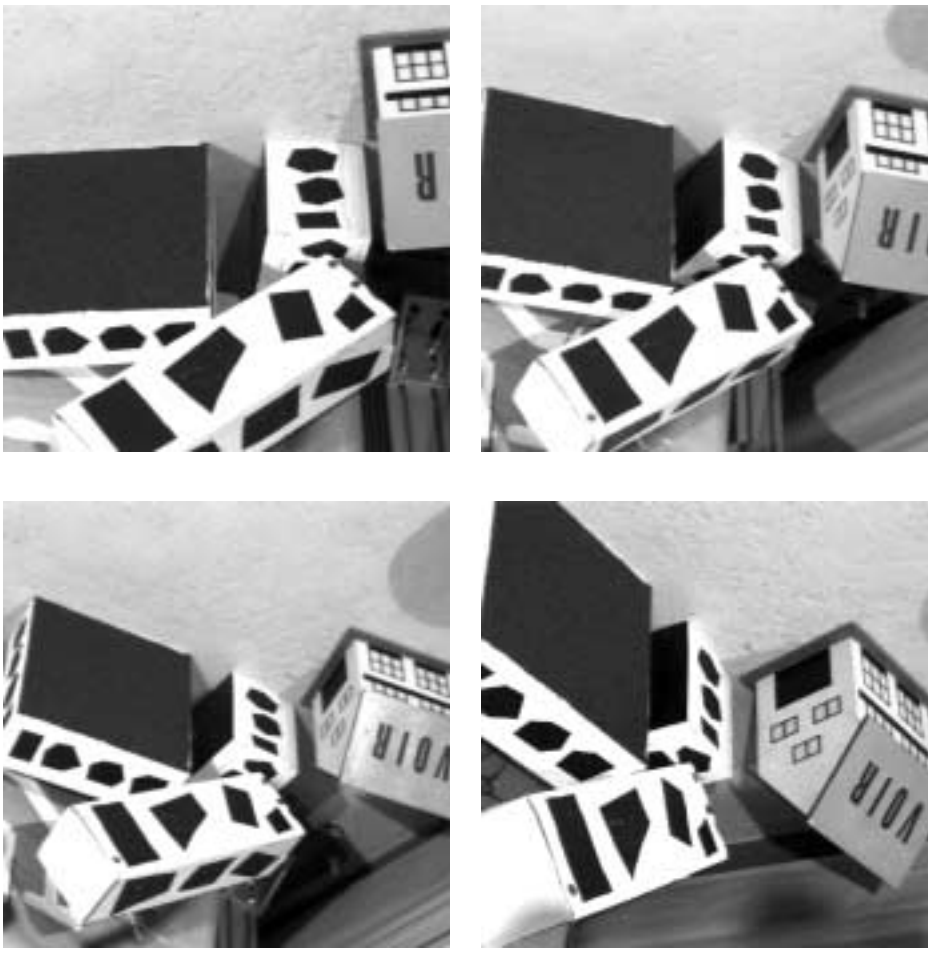

Fig. 14. A sequence of four images used for hand-eye self-calibration in Experiment 2.

which rewrites as

$$(\mathbf{I} \otimes \mathbf{R}_x)(\mathbf{R} \otimes \mathbf{R})(\mathbf{I} \otimes \mathbf{R}_x^T)(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x} = \alpha(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x}.$$

Hence,

$$(\mathbf{R} \otimes \mathbf{R}')(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x} = \alpha(\mathbf{I} \otimes \mathbf{R}_x)\mathbf{x}.$$

□

PRELIMINARY RESULT 2. Let $\mathbf{R}_1$ and $\mathbf{R}_2$ be two rotation matrices with nonparallel axes and $\mathbf{R}_i, 2 < i \leq n$, any rotation matrices. Let $\mathbf{R}$ be another rotation matrix. Then,

$$\left.\begin{array}{l} \mathbf{R}_1 \otimes \mathbf{R}_1 \; vec(\mathbf{R}) = vec(\mathbf{R}) \\ \mathbf{R}_2 \otimes \mathbf{R}_2 \; vec(\mathbf{R}) = vec(\mathbf{R}) \\ \vdots \\ \mathbf{R}_n \otimes \mathbf{R}_n \; vec(\mathbf{R}) = vec(\mathbf{R}) \end{array}\right\} \Rightarrow \mathbf{R} = \mathbf{I}_3.$$

Proof. The first two equations of the previous system are equivalent to

$$\begin{array}{l} \mathbf{R}_1\mathbf{R} = \mathbf{R}\mathbf{R}_1 \\ \mathbf{R}_2\mathbf{R} = \mathbf{R}\mathbf{R}_2. \end{array}$$

If $\mathbf{R}$ satisfies the first equation, then either $\mathbf{R}$ is the identity or it has the same rotation axis as $\mathbf{R}_1$. Similarly, it is either the identity or has the same rotation axis as $\mathbf{R}_2$. As $\mathbf{R}_1$ and $\mathbf{R}_2$ have different rotation axes, it must be the identity.

With the same logic, adding other rotations $\mathbf{R}_i, 2 < i \leq n$ such that $\mathbf{R}_i \otimes \mathbf{R}_i \; vec(\mathbf{R}) = vec(\mathbf{R})$ does not change the conclusion of the above result concerning $\mathbf{R}$. □

PRELIMINARY RESULT 3. Let $\mathbf{R}_1$ and $\mathbf{R}_2$ be two rotation matrices with nonparallel rotation axes and $\mathbf{R}_i, 2 < i \leq n$, any rotation matrices. Let $\mathbf{M} \neq 0$ be a matrix such that

$$\begin{array}{c} \mathbf{R}_1 \otimes \mathbf{R}_1 \; vec(\mathbf{M}) = vec(\mathbf{M}) \\ \mathbf{R}_2 \otimes \mathbf{R}_2 \; vec(\mathbf{M}) = vec(\mathbf{M}) \\ \vdots \\ \mathbf{R}_n \otimes \mathbf{R}_n \; vec(\mathbf{M}) = vec(\mathbf{M}). \end{array}$$

Then,

$$\exists \mu \neq 0, \mathbf{M} = \mu \mathbf{I}_3.$$

Proof. To write

$$\mathbf{R}_1 \otimes \mathbf{R}_1 \; vec(\mathbf{M}) = vec(\mathbf{M})$$

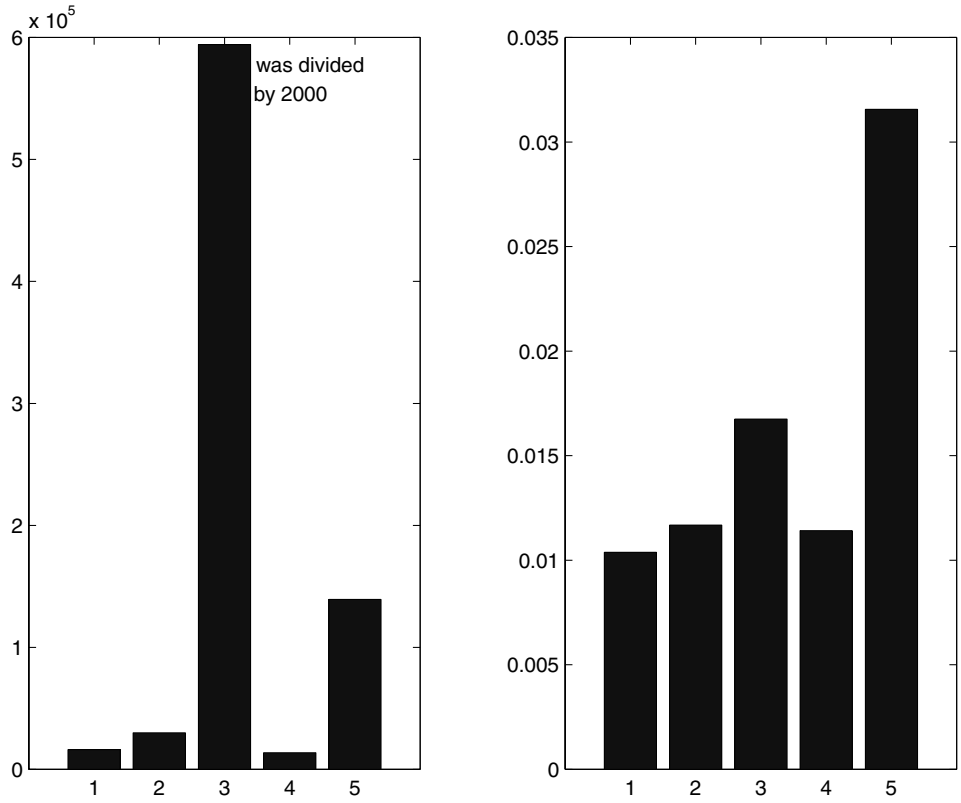

Fig. 15. RMS errors in rotation (left) and translation (right) with four images (see text).

is equivalent to say that $\mathbf{R}_1$ and $\mathbf{M}$ commute. Therefore, $\mathbf{M}$ is of the form $\mu\mathbf{R}$ where $\mu \neq 0$ and $\mathbf{R}$ is a rotation matrix that commutes with $\mathbf{R}_1$. This can be easily seen by replacing $\mathbf{M}$ by its SVD.

Thus, $\mathbf{M} = \mu\mathbf{R}$ where $\mathbf{R}$ is such that

$$\begin{aligned} \mathbf{R}_1\mathbf{R} &= \mathbf{R}\mathbf{R}_1 \\ \mathbf{R}_2\mathbf{R} &= \mathbf{R}\mathbf{R}_2 \\ &\vdots \\ \mathbf{R}_n\mathbf{R} &= \mathbf{R}\mathbf{R}_n. \end{aligned} \tag{43}$$

From Preliminary result 2, we obtain $\mathbf{R} = \mathbf{I}_3$ and $\mathbf{M} = \mu\mathbf{I}_3$. □

A.2. Proof of Lemma 1

System (26) is equivalent to

$$\begin{aligned} \mathbf{R}_{a1} \otimes \mathbf{R}_{b1}\mathbf{v} &= \mathbf{v} \\ \mathbf{R}_{a2} \otimes \mathbf{R}_{b2}\mathbf{v} &= \mathbf{v} \\ &\vdots \\ \mathbf{R}_{an} \otimes \mathbf{R}_{bn}\mathbf{v} &= \mathbf{v}. \end{aligned} \tag{44}$$

Under the assumption that the camera motions and the robot motions are rigidly linked by a fixed hand-eye transformation $(\mathbf{R}_x, \mathbf{t}_x)$ and from Preliminary result 1, this system becomes

$$\begin{aligned} \mathbf{R}_{b1} \otimes \mathbf{R}_{b1}\mathbf{v}' &= \mathbf{v}' \\ \mathbf{R}_{b2} \otimes \mathbf{R}_{b2}\mathbf{v}' &= \mathbf{v}' \\ &\vdots \\ \mathbf{R}_{bn} \otimes \mathbf{R}_{bn}\mathbf{v}' &= \mathbf{v}', \end{aligned} \tag{45}$$

where $\mathbf{v}' = (\mathbf{I} \otimes \mathbf{R}_x^T)\mathbf{v}$. Applying the result of Preliminary result 3, we obtain that $vec^{-1}(\mathbf{v}') = \mu\mathbf{I}_3$. Using the definition of $\mathbf{v}'$ and the properties of the Kronecker product, we end up with

$$vec^{-1}(\mathbf{v})\mathbf{R}_x^T = \mu\mathbf{I}_3,$$

where $\mathbf{V} = vec^{-1}(\mathbf{v})$. Hence,

$$\mathbf{V} = \mu\mathbf{R}_x.$$

Consequently, the matrix $\mathbf{V}$ extracted from the null space of (26) is proportional to the hand-eye rotation. The coefficient μ is obtained from the orthogonality constraint: $det(\mathbf{R}_x) = 1$. The latter becomes $det(\mathbf{V}) = \mu^3$, which finally gives

$$\mu = sgn(det(\mathbf{V}))\ |det(\mathbf{V})|^{1/3}.$$

□

Acknowledgments

This work was supported by the European Community through the Esprit-IV reactive LTR project number 26247 (VIGOR). During this work, Nicolas Andreff was staying at INRIA Rhône-Alpes as a Ph.D. student.

References

Andreff, N. 1999. *Asservissement visuel à partir de droites et auto-étalonnage pince-caméra*. Thèse de Doctorat, Institut National Polytechnique de Grenoble, Grenoble, November. Available: http://www.inrialpes.fr/bip/people/andreff/docs/these.ps.gz.

Bellman, R. 1960. *Introduction to Matrix Analysis*. New York: McGraw-Hill.

Brewer, J. W. 1978. Kronecker products and matrix calculus in system theory. *IEEE Trans. on Circuits and Systems* CAS-25(9):772–781.

Chen, H. H. 1991. A screw motion approach to uniqueness analysis of head-eye geometry. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. 145–151.

Chou, J.C.K., and Kamel, M. 1991. Finding the position and orientation of a sensor on a robot manipulator using quaternions. *Int. J. Robotics Res.* 10(3):240–254.

Christy, S., and Horaud, R. 1996. Euclidean shape and motion from multiple perspective views by affine iterations. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 18(11):1098–1104.

Cui, N., Weng, J. J., and Cohen, P. 1994. Recursive-batch estimation of motion and structure from monocular image sequences. *Computer Vision and Image Understanding* 59(2):154–170.

Daniilidis, K., and Bayro-Corrochano, E. 1996. The dual quaternion approach to hand-eye calibration. *Proc. IAPR International Conference on Pattern Recognition*, pp. 318–322.

Deif, A. S., Seif, N. P., and Hussein, S. A. 1995. Sylvester's equation: Accuracy and computational stability. *J. Computational and Applied Mathematics* 61:1–11.

Devernay, F., and Faugeras, O. 1996. From projective to Euclidean reconstruction. *Proc. Conf. on Computer Vision and Pattern Recognition*, June, pp. 264–269.

Espiau, B. 1993. Effect of camera calibration errors on visual servoing in robotics. *Third International Symposium on Experimental Robotics*, October.

Espiau, B., Chaumette, F., and Rives, P. 1992. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation* 8(3):313–326.

Gruen, A. W. 1985. Adaptative least squares correlation: A powerful image matching technique. *S. Afr. Journal of Photogrammetry, Remote Sensing and Cartography* 14(3):175–187.

Hager, G., and Toyama, K. 1998. X vision: A portable substrate for real-time vision applications. *Computer Vision and Image Understanding* 69(1):23–37.

Horaud, R., and Dornaika, F. 1995. Hand-eye calibration. *Int. J. Robotics Res.* 14(3):195–210.

Horn, B.K.P., Hilden, H. M., and Negahdaripour, S. 1988. Closed form solution of absolute orientation using orthonormal matrices. *Journal of the Optical Society A* 5(7):1127–1135.

Hu, D. Y., and Reichel, L. 1992. Krylov-subspace methods for the Sylvester equation. *Linear Algebra and Its Applications* 172:283–313.

Koenderink, J., and van Doorn, A. 1991. Affine structure from motion. *Journal of the Optical Society of America A* 8(2):377–385.

Li, M. 1998. Kinematic calibration of an active head-eye system. *IEEE Trans. on Robotics and Automation* 14(1):153–158.

Murray, R. M., Li, Z., and Sastry, S. S. 1994. *A Mathematical Introduction to Robotic Manipulation*. CRC.

Neudecker, H. 1969. A note on Kronecker matrix product and matrix equation systems. *SIAM J. Appl. Math.* 17(3):603–606.

Oliensis, J. 2000. A critique of structure from motion algorithms. *Computer Vision and Image Understanding* 80:172–214.

Poelman, C. J., and Kanade, T. 1994. A paraperspective factorization method for shape and motion recovery. In *Proc. 3rd European Conference on Computer Vision*, ed. J. O. Eklundh, Stockholm, Sweden, May, pp. 97–108. New York: Springer-Verlag.

Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. 1992. *Numerical Recipes in C—The Art of Scientific Computing*. Cambridge, UK: Cambridge University Press.

Rémy, S., Dhome, M., Lavest, J. M., and Daucher, N. 1997. Hand-eye calibration. *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1057–1065.

Rotella, F., and Borne, P. 1989. Explicit solution of Sylvester and Lyapunov equations. *Mathematics and Computers in Simulation* 31:271–281.

Ruf, A., Martin, F., Lamiroy, B., and Horaud, R. 1999. Visual control using projective kinematics. *Ninth International Symposium of Robotics Research*, Snowbird, UT, October, pp. 66–73.

Samson, C., Le Borgne, M., and Espiau, B. 1991. *Robot Control: The Task Function Approach*. Oxford, UK: Clarendon.

Shiu, Y. C., and Ahmad, S. 1989. Calibration of wrist mounted robotic sensors by solving homogeneous transform equations of the form AX=XB. *IEEE Trans. on Robotics and Automation* 5(1):16–29.

Taylor, C. T., Kriegman, D. J., and Anandan, P. 1991. Structure and motion in two dimensions from multiple images: A least squares approach. *Proc. IEEE Workshop on Visual Motion*, October, pp. 242–248. IEEE Computer Society Press.

Tomasi, C., and Kanade, T. 1991. Factoring image sequences into shape and motion. *Proc. IEEE Workshop on Visual Motion*, pp. 21–28. IEEE Computer Society Press.

Tsai, R. Y., and Lenz, R. K. 1989. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Trans. on Robotics and Automation* 5(3):345–358.

Wang, C.-C. 1992. Extrinsic calibration of a robot sensor mounted on a robot. *IEEE Trans. on Robotics and Automation* 8(2):161–175.

Wei, G.-Q., Arbter, K., and Hirzinger, G. 1998. Active self-calibration of robotic eyes and hand-eye relationships with model identification. *IEEE Trans. on Robotics and Automation* 14(1):158–165.

Zhuang, H. 1998. A note on using only position equations for robotic hand/eye calibration. *IEEE Trans. on Systems, Man and Cybernetics* 28(3):426–427.