Euclidean Reconstruction and Affine Camera Calibration Using Controlled Robot Motions

Radu Horaud, Stéphane Christy, Roger Mohr

Euclidean Reconstruction and Affine Camera Calibration Using Controlled Robot Motions *

Radu Horaud, Stéphane Christy, and Roger Mohr

GRAVIR–IMAG & INRIA Rhône-Alpes
655, avenue de l'Europe
38330 Montbonnot Saint-Martin FRANCE

Abstract

In this paper we are addressing the problem of Euclidean reconstruction with an uncalibrated affine camera and calibration of this camera. Since efficiency is an important issue in robotics and computer vision, we investigate constraints under which the Euclidean shape and motion problem becomes linear. The theoretical study that is described in this paper leads us to impose some practical constraints, namely we require that the camera is mounted onto a robot arm and that the robot is executing controlled motions whose parameters are known. The affine camera model considered here is certainly just an approximation of the true projective mapping. Nevertheless, there is a large number of applications for which the camera is allowed to be at some distance from the scene and under these circumstances the affine model is a good approximation. The fact that we deal with an uncalibrated camera is an advantage over previous methods because we do not rely anymore on the tedious task of camera calibration. The experiments that are described at the end of the paper show that the method described herein compares favorably with other similar methods.

1 Introduction and background

One of the most challenging tasks in robotics is the task of recovering three-dimensional Euclidean structure of a scene from a sequence of images gathered with a moving camera. Roughly speaking, there are two possible approaches to this problem. The first class of approaches uses a calibrated camera in which case the task of recovering shape and motion is non-linear if a perspective camera model is considered [2, 10], and linear if an affine camera model is considered [8, 11, 13]. The second class of approaches uses an uncalibrated camera in which case the task of recovering Euclidean shape and motion is a non-linear problem both with projective [3, 4, 12], and affine [9] camera models.

*This work has been supported by "Société Aérospatiale" and by DGA/DRET.

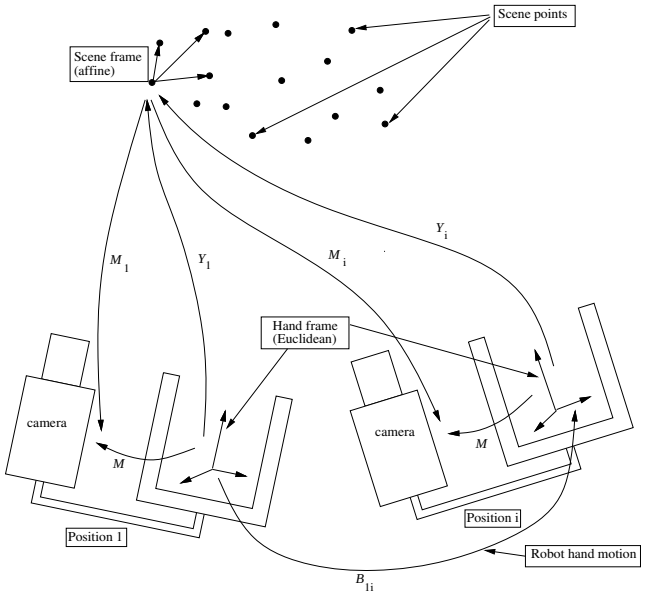

Figure 1: This figure shows the camera rigidly mounted onto a robot hand. The hand undergoes a rigid motion (and so does the camera) and the camera observes a 3-D scene materialized by a number of points.

The standard way to solve a non linear set of equations is to cast the problem into a least squares minimization problem and to use numerical methods in order to find a solution. Such non-linear optimization methods are appealing because they can deal with noise and, to a certain extent, with outliers, but have two drawbacks: they require some form of initialization which is not always an obvious task and they are iterative in nature. The cost of each iteration is at least equal to the cost of inverting a symmetric positive semi-definite matrix ($J^T J$ where J is the Jacobian of the sum-of-squares error function to be minimized). The size of this matrix is directly related to the number of images in the sequence and to the number of points to be reconstructed [4] and hence matrix inversion can

be a time-consuming process.

In this paper we are addressing the problem of Euclidean reconstruction with an uncalibrated affine camera. Since efficiency is an important issue in computer vision we investigate constraints under which the Euclidean shape and motion problem becomes linear. The theoretical study that is described in this paper leads us to impose some practical constraints, namely we require that the camera is mounted onto a robot arm and that the robot is executing controlled motions whose parameters are known. Recent work has shown that the combination of robot controlled motions with a projective camera does not make the problem any simpler [7]. The affine camera model considered here is certainly just an approximation of the true projective mapping. Nevertheless, there is a large number of applications for which the camera is allowed to be at some distance from the scene and under these circumstances the affine model is a good approximation. The fact that we deal with an uncalibrated camera is an advantage over previous methods because we do not rely anymore on the tedious task of camera calibration.

Since the camera is mounted onto a robot hand there is a fixed rigid transformation between the camera frame and the hand frame. This transformation is not known in advance and hence the camera motion is known only up to a rotation and translation.

More formally, the problem that we attempt to solve in this paper can be stated as follows: Given an uncalibrated affine camera that is mounted onto a robot hand, given a number of point-to-point correspondences between images gathered with this camera and given known robot motions, then (i) determine the Euclidean structure of the scene, (ii) calibrate the affine camera, and (iii) estimate the camera-to-hand transformation.

The problem of Euclidean shape and motion estimation with an affine camera has received a lot of attention in the past. In the case of a calibrated camera, Tomasi & Kanade [11] (orthographic projection), Weinshall & Tomasi [14] (weak perspective), and Poelman & Kanade [8] (paraperspective) showed that the problem can be solved using straightforward linear methods. Quan [9] extended these approaches to the case of an uncalibrated camera. He showed that in this case one cannot use linear methods anymore and the problem becomes non-linear. Moreover, he suggested a canonical decomposition of the affine transformation matrix into intrinsic and extrinsic camera parameters. This decomposition treats various affine camera models (orthography, weak perspective, and paraperspective) within a common framework. All the methods mentioned above that use an affine camera model can determine shape only up to a mirror transformation. In this paper we show under which controlled motions this ambiguity is removed.

The work reported herein is also related to the problem of hand-eye calibration. Previous work on hand-eye calibration attempted to solve the problem in two stages: first the robot is moved to some predetermined locations and at each one of these locations the intrinsic and extrinsic camera parameters are estimated using a known 3-D calibration grid, and second a set of homogeneous matrix equations is solved in order to determine the camera-to-hand transformation [6]. Our method may well be viewed as an on-line hand-eye calibration technique which requires neither a known calibration grid nor intrinsic camera parameters. Since in this paper we consider an affine camera, only the rotation associated with the hand-eye transformation can be determined.

To summarize, the main contributions of this paper are the following:

- We show that the problem of Euclidean reconstruction with an uncalibrated affine camera has a simple linear formulation if the camera is mounted onto a robot and if the robot executes controlled motions,
- We show the existence of a sufficient condition under which a unique solution may be obtained, namely we show that two independent robot rotations are sufficient to insure the uniqueness of the solution, and
- We solve simultaneously for Euclidean reconstruction, camera calibration, and hand-eye calibration.

Paper organization. The remainder of this article is organized as follows. Section 2 describes the affine camera model that will be used throughout the paper. Section 3 describes the problem formulation allowing to express reconstruction and calibration into a common framework. Section 4 provides a sufficient condition that insures that the problem has a unique solution. Section 5 describes the problem solution which involves simple linear algebra and shows how recovering camera parameters (camera calibration). Section 6 describes a number of experiments performed with an uncalibrated camera mounted onto a robot arm and Section 7 gives some directions for future work.

2 The affine camera

In its most general form, the affine camera model can be written as:

$$\mathbf{p} = \mathbf{M}\mathbf{P} \tag{1}$$

where $\mathbf{p}$ is an image point expressed in standard image coordinates, $\mathbf{P}$ is 3-D point expressed in some Eu-

clidean frame, and $\mathbf{M}$ is a 3×4 matrix of the form:

$$\mathbf{M} = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2}$$

Eq. (1) can also be written as:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \underbrace{\begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \end{pmatrix}}_{\mathbf{N}} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \underbrace{\begin{pmatrix} m_{14} \\ m_{24} \end{pmatrix}}_{\mathbf{n}} \tag{3}$$

The translation can therefore be "absorbed" by changing the origin of the image frame:

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} u - m_{14} \\ v - m_{24} \end{pmatrix} = \mathbf{N} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{4}$$

The 2-vector $\mathbf{n}$ is the origin of the new image frame and it is the projection of the 3-D frame's origin.

In order to make explicit the intrinsic and extrinsic parameters of the affine camera, one may apply QR-decomposition to matrix $\mathbf{N}$ which provides [9]:

$$\mathbf{N} = \underbrace{\begin{pmatrix} a & 0 \\ b & c \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \end{pmatrix}}_{\mathbf{R}_{2\times 3}} \tag{5}$$

This decomposition is unique if the rank of $\mathbf{N}$ is equal to 2. Matrix $\mathbf{A}$ encodes 3 parameters associated with an affine camera and $\mathbf{R}_{2\times 3}$ contains two rows of a rotation matrix, the third row being easily recovered with:

$$\mathbf{r}_3 = \mathbf{r}_1 \times \mathbf{r}_2$$

Special cases of the affine camera model are: orthography, weak perspective, and paraperspective. However, in this paper we will not make any prior assumption about one or the other of these specific models.

3 Problem formulation

We consider now a camera that is rigidly mounted onto a robot arm and we associate an Euclidean frame with the robot hand — this frame is known in robotics as the tool frame and it is the frame able to perform controlled motions, provided that the robot kinematic model is known. Let ($\mathbf{R}$ $\mathbf{t}$) be the rigid transformation (rotation and translation) between the hand frame and the standard camera Cartesian frame. The elements of $\mathbf{R}$ and $\mathbf{t}$ are the parameters associated with hand-eye calibration. However, with an affine camera, only the rotational parameters can be recovered, that is the row vectors of matrix $\mathbf{R}$.

We assume now that the robot hand frame is the 3-D frame in eq. (1) and therefore the camera projection matrix is (see Figure 1):

$$\mathbf{M} = \begin{pmatrix} \mathbf{A}\mathbf{R}_{2\times 3} & \mathbf{n} \\ \mathbf{0}^T & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{N} & \mathbf{n} \\ \mathbf{0}^T & 1 \end{pmatrix} \tag{6}$$

where $\mathbf{A}$ contains the 3 intrinsic parameters of the affine camera and $\mathbf{R}_{2\times 3}$ contains the first 2 row vectors of the rotation matrix $\mathbf{R}$.

Therefore, Matrix $\mathbf{M}$ encapsulates both the camera calibration problem and the hand-eye calibration problem. Of course $\mathbf{M}$ cannot be directly estimated, and in what follows we will provide a method that first computes a 3-D reconstruction of a scene and second calibrates the camera as well as the hand-eye relationship.

We assume now that both the camera and the robot hand undergo a sequence of rigid motions and that neither $\mathbf{A}$ nor $\mathbf{R}$ and $\mathbf{t}$ vary during these motions: hence, matrix $\mathbf{M}$ is assumed to be fixed during these motions. The camera observes a 3-D scene and image-to-image correspondences can therefore be established and a affine scene reconstruction can be performed. Indeed, such methods as factorization [11] or affine-invariant methods [13] allow one to compute 3-D affine coordinates of the observed scene in some scene centered reference frame. Hence there is an affine frame associated with the 3-D scene. Let $\mathbf{S}_j$ be the affine coordinates of the scene points expressed in this frame and let $\mathbf{M}_i$ be the transformation from this frame to the frame associated with the i^{th} image, e.g., Figure 1. Therefore $\mathbf{P}_j$ and $\mathbf{S}_j$ denote the *same physical 3-D point* expressed in two different coordinate frames: a scene centered affine frame and the hand frame which is an Euclidean frame.

Let $\mathbf{Y}_i$ be a 4×4 invertible matrix associated with a transformation from the hand Euclidean frame (in position i) to the scene affine frame. Matrix $\mathbf{Y}_i$ describes a 3-D affine transformation. Therefore, the matrix $\mathbf{M}$ can be written as a combination of two affine transformations, e.g., Figure 1:

$$\begin{aligned} \mathbf{M} &= \mathbf{M}_1\mathbf{Y}_1 \\ &= \dots \\ &= \mathbf{M}_i\mathbf{Y}_i \\ &= \dots \\ &= \mathbf{M}_n\mathbf{Y}_n \end{aligned} \tag{7}$$

In these equations $\mathbf{M}_1, \dots, \mathbf{M}_i, \dots, \mathbf{M}_n$ are 3×4 matrices describing affine transformations from the 3-D affine space to each one of the images associated with the camera in motion. Moreover, let $\mathbf{B}_{1i}$ be a 4×4 matrix that expresses the rigid hand motion between the first position – 1 – and any other position – i. The relationship between $\mathbf{Y}_1$, $\mathbf{Y}_i$, and $\mathbf{B}_{1i}$ is simply:

$$\mathbf{Y}_i = \mathbf{Y}_1\mathbf{B}_{1i}^{-1}$$

By substituting the $\mathbf{Y}_i$'s in eq. (7) we obtain the following set of $n-1$ matrix equations:

$$\left\{\begin{array}{rcl} \mathbf{M}_1\mathbf{Y}_1\mathbf{B}_{12} &=& \mathbf{M}_2\mathbf{Y}_1 \\ & \vdots & \\ \mathbf{M}_1\mathbf{Y}_1\mathbf{B}_{1n} &=& \mathbf{M}_n\mathbf{Y}_1 \end{array}\right. \tag{8}$$

Each one of these matrix equations can be written as:

$$\underbrace{\begin{pmatrix} \mathbf{N}_1 & \mathbf{n}_1 \\ \mathbf{0}^T & 1 \end{pmatrix}}_{3\times4}\underbrace{\begin{pmatrix} \mathbf{X}_1 & \mathbf{x}_1 \\ \mathbf{0}^T & 1 \end{pmatrix}}_{4\times4}\underbrace{\begin{pmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0}^T & 1 \end{pmatrix}}_{4\times4} = \underbrace{\begin{pmatrix} \mathbf{N}_i & \mathbf{n}_i \\ \mathbf{0}^T & 1 \end{pmatrix}}_{3\times4}\underbrace{\begin{pmatrix} \mathbf{X}_1 & \mathbf{x}_1 \\ \mathbf{0}^T & 1 \end{pmatrix}}_{4\times4}$$

This matrix equation can further be decomposed as:

$$\mathbf{N}_1\mathbf{X}_1\mathbf{R}_i = \mathbf{N}_i\mathbf{X}_1 \tag{9}$$

$$\mathbf{N}_1\mathbf{X}_1\mathbf{t}_i + (\mathbf{N}_1 - \mathbf{N}_i)\mathbf{x}_1 = \mathbf{n}_i - \mathbf{n}_1 \tag{10}$$

The first one of these equations is composed of 2×3 matrices and it is homogeneous in the elements of the 3×3 unknown matrix $\mathbf{X}_1$. The second one is composed of 2-vectors and the unknowns are the matrix $\mathbf{X}_1$ and the 3-vector $\mathbf{x}_1$:

$$\mathbf{Y}_1 = \begin{pmatrix} \mathbf{X}_1 & \mathbf{x}_1 \\ \mathbf{0}^T & 1 \end{pmatrix}$$

Therefore there are 12 unknowns and each camera motion provides $6+2$ constraints. For n camera positions there are $n-1$ motions which provide $8(n-1)$ linear equations. Therefore, n must at least be equal to 3 in order to solve for the elements of $\mathbf{Y}_1$.

Before we proceed further and develop a solution to the problem, let's recapitulate the main stages of the method:

1. Perform a number (at least 2) of *controlled motions* with the robot on which the camera is mounted. These robot motions provide the elements of the matrices $\mathbf{B}_{1i} = (\ \mathbf{R}_i \quad \mathbf{t}_i\ )$. Grab an image at each position, extract interest points from each image and establish point correspondences between images.

2. Determine *affine structure* and *affine motion* from the extracted 2-D points, that is, determine 3-D affine coordinates $\mathbf{S}_j$ in a scene centered frame as well as the projection matrices $\mathbf{M}_i$ which transform these points into image points. Thus, we have $\mathbf{N}_i$ and $\mathbf{n}_i$ in equations (9) and (10).

3. Convert affine structure into *Euclidean structure*. This is equivalent to solving the overconstrained set of linear equations as described above (eq. (9) and (10)). We deduce $\mathbf{X}_1$ and $\mathbf{x}_1$, i.e. matrix $\mathbf{Y}_1$. Once $\mathbf{Y}_1$ is determined one can easily convert affine coordinates into Euclidean coordinates:

$$\forall j \ \ \mathbf{P}_j = \mathbf{Y}_1^{-1}\mathbf{S}_j \tag{11}$$

4. *Affine camera and hand-eye calibration* are performed by computing matrix $\mathbf{M} = (\ \mathbf{N} \quad \mathbf{n}\ )$ using any of the expressions in eq. (7). As already explained in section 2, the QR-decomposition of the 2×3 matrix $\mathbf{N}$ provides the intrinsic and extrinsic parameters of the affine camera. The extrinsic parameters (a 3×3 rotation matrix) amount for the hand-eye calibration.

4 A sufficient condition for uniqueness

As it has been outlined in the previous section, the problem to be solved is the resolution of a system of matrix equalities defined by eq. (8). One such equality can be written as:

$$\mathbf{M}_1 = \mathbf{M}_i\mathbf{Y}_1\mathbf{B}_{1i}^{-1}\mathbf{Y}_1^{-1}$$

Let $\mathbf{U}$ be a 4×4 invertible matrix that belongs to the subgroup of matrices of the affine group that commutes with $\mathbf{B}_{1i}^{-1}$, or with $\mathbf{B}_{1i}$, and let $\mathcal{C}(\mathbf{B}_{1i})$ denote this subgroup. We have:

$$\begin{aligned} \mathbf{M}_1 &= \mathbf{M}_i\mathbf{Y}_1\mathbf{B}_{1i}^{-1}\mathbf{Y}_1^{-1} \\ &= \mathbf{M}_i\mathbf{Y}_1\ \mathbf{U}\mathbf{U}^{-1}\ \mathbf{B}_{1i}^{-1}\mathbf{Y}_1^{-1} \\ &= \mathbf{M}_i\mathbf{Y}_1\ \mathbf{U}\mathbf{B}_{1i}^{-1}\ \mathbf{U}^{-1}\mathbf{Y}_1^{-1} \\ &= \mathbf{M}_i\ (\mathbf{Y}_1\mathbf{U})\ \mathbf{B}_{1i}^{-1}\ (\mathbf{Y}_1\mathbf{U})^{-1} \end{aligned}$$

Therefore, if $\mathbf{Y}_1$ is a solution, $\mathbf{Y}_1\mathbf{U}$ is a solution as well and $\mathbf{Y}_1$ will not be uniquely determined. As a consequence, one has to choose a sequence of hand motions B_{1i} such that the only possible choice for $\mathbf{U}$ is the identity matrix:

$$\bigcap_{i>1} \mathcal{C}(\mathbf{B}_{1i}) = \{\mathbf{I}\}$$

In what follows we derive a *sufficient* condition for obtaining the result above. We show that at least 2 motions (or 3 robot positions) are necessary, that there should be at least 2 distinct axes of rotation, and that one among the motions should be such that its translation vector is not orthogonal to its axis of rotation.

Indeed, let $\mathbf{U}$ and $\mathbf{B}_{1i}$ be written as:

$$\mathbf{U} = \begin{pmatrix} \mathbf{W} & \mathbf{u} \\ \mathbf{0}^T & 1 \end{pmatrix} \qquad \mathbf{B}_{1i} = \begin{pmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{0}^T & 1 \end{pmatrix}$$

We seek a matrix $\mathbf{U}$ satisfying:

$$\mathbf{B}_{1i}\mathbf{U} = \mathbf{U}\mathbf{B}_{1i}$$

By developing the matrix products on both sides of the equation above we obtain:

$$\mathbf{W}\mathbf{R}_i = \mathbf{R}_i\mathbf{W} \quad (12)$$

$$\mathbf{W}\mathbf{t}_i + \mathbf{u} = \mathbf{R}_i\mathbf{u} + \mathbf{t}_i \quad (13)$$

Let $\mathbf{n}_1$, $\mathbf{n}_2$, and $\mathbf{n}_3$ be the eigenvectors of the rotation matrix $\mathbf{R}_i$. It is well known that a rotation matrix has three distinct eigenvalues: $\mu_1 = 1$, $\mu_2 = e^{i\theta}$, and $\mu_3 = e^{-i\theta}$. From eq. (12) we easily obtain that $\mathbf{W}$ and $\mathbf{R}_i$ must have *the same eigenvectors* but they can have distinct eigenvalues. Indeed, with $\mathbf{R}_i\mathbf{n}_j = \mu_j\mathbf{n}_j$ and by substitution in eq. (12):

$$\mathbf{W}\mathbf{R}_i\mathbf{n}_j = \mathbf{W}\mu_j\mathbf{n}_j = \mu_j\mathbf{W}\mathbf{n}_j = \mathbf{R}_i\mathbf{W}\mathbf{n}_j$$

Hence, $\mathbf{W}\mathbf{n}_j = \nu_j\mathbf{n}_j$ with $j = 1,2,3$ and one may notice that the eigenvalues of $\mathbf{W}$, ν_1, ν_2, and ν_3 cannot be null because $\mathbf{W}$ is not singular.

We consider now a second motion with its associated rotation matrix $\mathbf{R}_k$ whose eigenvectors are denoted by $\mathbf{m}_1$, $\mathbf{m}_2$, and $\mathbf{m}_3$. Let $\mathbf{m}_1$ be the axis of rotation of $\mathbf{R}_k$ and let us impose that this vector is neither parallel to $\mathbf{n}_1$ (the axis of rotation of $\mathbf{R}_i$) nor in the plane defined by $\mathbf{n}_2$ and $\mathbf{n}_3$. $\mathbf{m}_1$ can therefore be written as a linear combination of the eigenvectors of $\mathbf{R}_i$:

$$\mathbf{m}_1 = \sum_{j=1}^{3} \lambda_j\mathbf{n}_j \quad (14)$$

We obtain:

$$\mathbf{W}\mathbf{m}_1 = \mathbf{W}\sum_{j=1}^{3}\lambda_j\mathbf{n}_j = \sum_{j=1}^{3}\lambda_j\nu_j\mathbf{n}_j \quad (15)$$

However, the eigenvectors of $\mathbf{W}$ and $\mathbf{R}_k$ should also be the same:

$$\mathbf{W}\mathbf{m}_1 = \nu_1\mathbf{m}_1 = \nu_1\sum_{j=1}^{3}\lambda_j\mathbf{n}_j = \sum_{j=1}^{3}\nu_1\lambda_j\mathbf{n}_j \quad (16)$$

By identifying eq. (16) with eq. (15) we obtain:

$$\nu_1 = \nu_2 = \nu_3 = \nu$$

Since the matrix $\mathbf{W}$ has three identical eigenvalues it is necessarily of the form:

$$\mathbf{W} = \nu\mathbf{I}$$

As a consequence, eq. (13) becomes:

$$(\nu - 1)\mathbf{t}_i = (\mathbf{R}_i - \mathbf{I})\mathbf{u}$$

In this expression, $(\mathbf{R}_i - \mathbf{I})\mathbf{u}$ is a vector perpendicular to the axis of rotation of $\mathbf{R}_i$ – the eigenvector $\mathbf{n}_1$.[1]

[1] In order to be convinced of this orthogonality, consider the dot product between vector $\mathbf{u}$ and the axis of rotation of $\mathbf{R}_i$, $\mathbf{n}_1$. This dot product is invariant with respect to rotation: $\mathbf{u}\cdot\mathbf{n}_1 = (\mathbf{R}_i\mathbf{u})\cdot(\mathbf{R}_i\mathbf{n}_1) = (\mathbf{R}_i\mathbf{u})\cdot\mathbf{n}_1$. It follows that $(\mathbf{R}_i\mathbf{u} - \mathbf{u})\cdot\mathbf{n}_1 = 0$.

Therefore, unless the translation vector $\mathbf{t}_i$ lies in a plane perpendicular to $\mathbf{n}_1$ and if we eliminate the trivial solution ($\mathbf{R}_i = \mathbf{I}$ and $\mathbf{t}_i = 0$), we have:

$$\nu = 1$$

and

$$\mathbf{u} = 0$$

Finally, we obtain that the unique solution for $\mathbf{U}$ is:

$$\mathbf{U} = \mathbf{I}$$

To conclude we proved the following sufficient condition which insures the uniqueness of the solution of eq. (8): *among all robot motions $\mathbf{B}_{1i}$, (i) at least two motions must have distinct axes of rotation and (ii) at least one motion must have its translation vector non orthogonal to its rotation axis.* Notice that the former condition is identical to the uniqueness condition associated with Euclidean hand-eye calibration. If the latter condition is not satisfied, Euclidean shape is recovered only up to a scale factor.

5 Problem solution

Let $\mathbf{P}_0, \mathbf{P}_1, \ldots \mathbf{P}_k$ be the set of scene points for which we seek 3-D Euclidean coordinates. As already explained, we first determine their affine coordinates, and second we convert these coordinates into Euclidean ones. Let $\mathbf{P}_0$ be the origin of the world frame. We define an affine basis of the 3-D space with the points $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$, and $\mathbf{P}_3$ which are supposed to be non coplanar. Vector $\mathbf{S}_j$ is the vector from $\mathbf{P}_0$ to $\mathbf{P}_j$. Let $\mathbf{S}_1$, $\mathbf{S}_2$, and $\mathbf{S}_3$ be an affine basis — they are three unit vectors: $\mathbf{S}_1 = (\ 1\ \ 0\ \ 0\ )^T$, $\mathbf{S}_2 = (\ 0\ \ 1\ \ 0\ )^T$, and $\mathbf{S}_3 = (\ 0\ \ 0\ \ 1\ )^T$.

Moreover, the scalar triplet $(\alpha_j, \beta_j, \gamma_j)$ denotes the affine coordinates of a vector $\mathbf{S}_j$. We have:

$$\mathbf{S}_j = \alpha_j\mathbf{S}_1 + \beta_j\mathbf{S}_2 + \gamma_j\mathbf{S}_3 \quad (17)$$

Let $\mathbf{p}_{ji}$ be the projection of $\mathbf{P}_j$ onto the i^{th} image. Hence $\mathbf{p}_{j1}, \mathbf{p}_{j2}, \ldots \mathbf{p}_{jn}$ are projections of $\mathbf{P}_j$ in the n images. Since we choose $\mathbf{P}_0$ to be the origin of the space frame, $\mathbf{p}_{01}, \mathbf{p}_{02}, \ldots \mathbf{p}_{0n}$ are the origins of the image frames that absorb the translational component. Similarly, the image vector $\mathbf{s}_{ji}$ denotes the vector from point $\mathbf{p}_{0i}$ to point $\mathbf{p}_{ji}$.

Therefore, for all j, and for all i ($i = 1 \ldots n$, $j = 1 \ldots k$) eq. (3) writes:

$$\mathbf{s}_{ji} = \mathbf{N}_i\mathbf{S}_j \quad (18)$$

$$\mathbf{p}_{0i} = \mathbf{n}_i \quad (19)$$

5.1 Affine shape and motion

By combining eq. (17) with eq. (18) we obtain:

$$\mathbf{s}_{ji} = \alpha_j\mathbf{N}_i\mathbf{S}_1 + \beta_j\mathbf{N}_i\mathbf{S}_2 + \gamma_j\mathbf{N}_i\mathbf{S}_3$$
$$= \alpha_j\mathbf{s}_{1i} + \beta_j\mathbf{s}_{2i} + \gamma_j\mathbf{s}_{3i}$$

For n images and for every point correspondence j, with $j > 3$, we obtain the following matrix equation:

$$\begin{pmatrix} \mathbf{s}_{11} & \mathbf{s}_{21} & \mathbf{s}_{31} \\ \vdots & \vdots & \vdots \\ \mathbf{s}_{1n} & \mathbf{s}_{2n} & \mathbf{s}_{3n} \end{pmatrix} \begin{pmatrix} \alpha_j \\ \beta_j \\ \gamma_j \end{pmatrix} = \begin{pmatrix} \mathbf{s}_{j1} \\ \vdots \\ \mathbf{s}_{jn} \end{pmatrix}$$

This can be written more compactly as:

$$\underbrace{\mathbf{B}}_{2n\times 3}\underbrace{\mathbf{S}_j}_{3\times 1} = \underbrace{\sigma_j}_{2n\times 1}$$

The pseudo-inverse of $\mathbf{B}$ can be computed provided that the rank of $\mathbf{B}$ is equal to 3. This rank condition is satisfied if:

- n (the number of views) is greater or equal to 2;
- The basis image vectors are not collinear. This is a necessary but not sufficient condition for insuring that the four 3-D basis points are not coplanar, and
- the camera motion is neither a pure translation nor a rotation around the optical axis.

With these conditions satisfied, one may compute the affine coordinates of any space point j, with $j > 3$:

$$\mathbf{S}_j = \mathbf{B}^\dagger \sigma_j$$

The overscript $\dagger$ stands for the pseudo-inverse of a matrix. This expression can be written for all j, $j > 3$:

$$\mathbf{S}^\star = \mathbf{B}^\dagger \sigma^\star \tag{20}$$

with:

$$\begin{aligned} \mathbf{S}^\star &= (\mathbf{S}_4 \dots \mathbf{S}_k) \\ \sigma^\star &= (\sigma_4 \dots \sigma_k) \end{aligned}$$

Therefore, the *affine shape* just determined is described by the $3 \times k$ matrix $\mathbf{S}$:

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}_1 & \mathbf{S}_2 & \mathbf{S}_3 & \mathbf{S}^\star \end{pmatrix}$$

We may now determine the affine transformations $\mathbf{N}_i$ as follows. Eq. (18) can be written for n views and for k points in the following form:

$$\begin{pmatrix} \mathbf{s}_{11} & \dots & \mathbf{s}_{k1} \\ \vdots & & \vdots \\ \mathbf{s}_{1n} & \dots & \mathbf{s}_{kn} \end{pmatrix} = \begin{pmatrix} \mathbf{N}_1 \\ \vdots \\ \mathbf{N}_n \end{pmatrix} \begin{pmatrix} \mathbf{S}_1 & \dots & \mathbf{S}_k \end{pmatrix}$$

More compactly this can be written as [11]:

$$\sigma = \mathbf{A}\mathbf{S} \tag{21}$$

We refer to this formula as the *affine shape and motion equation.*

While the affine-invariant method allows a more direct analysis of the problem, the factorization method [1, 8, 11] is more convenient from a practical point of view. This method avoid the problem of finding four non coplanar object point to perform the reconstruction. The factorization method computes affine shape and motion simultaneously by performing a singular value decomposition of the $2n \times k$ matrix σ (equation (21)).

5.2 Euclidean shape and motion

We are now able to solve for $\mathbf{Y}_1$ in eq. (8), where each matrix constraint can be decomposed into eq. (9) and eq. (10):

$$\begin{aligned} \mathbf{N}_1\mathbf{X}_1\mathbf{R}_i &= \mathbf{N}_i\mathbf{X}_1 \\ \mathbf{N}_1\mathbf{X}_1\mathbf{t}_i + (\mathbf{N}_1 - \mathbf{N}_i)\mathbf{x}_1 &= \mathbf{n}_i - \mathbf{n}_1 \end{aligned}$$

The first one of these equations has $\mathbf{X}_1$ as unknown. By combining $n-1$ such equations one obtains an overconstrained homogeneous set of linear equations of the form ($\mathcal{X}_1$ is a 9-vector formed with the elements of $\mathbf{X}_1$):

$$\underbrace{\mathcal{A}}_{6(n-1)\times 9}\underbrace{\mathcal{X}_1}_{9\times 1} = \mathbf{0} \tag{22}$$

The optimal solution for $\mathcal{X}_1$ can be found by solving:

$$\min_{\mathcal{X}_1} \left(\|\mathcal{A}\mathcal{X}_1\|^2 + \lambda(1 - \|\mathcal{X}_1\|^2) \right)$$

The solution is the eigenvector associated with the smallest eigenvalue of the symmetric semi-definite positive matrix $\mathcal{A}^T\mathcal{A}$. Therefore the elements of $\mathbf{X}_1$ are defined up to a scale factor: $\mu\mathbf{X}_1$.

By substituting $\mu\mathbf{X}_1$ in the second equation we obtain:

$$\mu\mathbf{N}_1\mathbf{X}_1\mathbf{t}_i + (\mathbf{N}_1 - \mathbf{N}_i)\mathbf{x}_1 = \mathbf{n}_i - \mathbf{n}_1$$

With n camera positions ($n-1$ motions) we obtain a linear set of equations of the form:

$$\underbrace{\mathcal{B}}_{2(n-1)\times 4}\underbrace{\begin{pmatrix} \mu \\ \mathbf{x}_1 \end{pmatrix}}_{4\times 1} = \mathbf{b}$$

This linear system has an obvious solution if the rank of $\mathcal{B}$ is equal to 4. The Euclidean coordinates of the scene points can now be recovered using eq. (11):

$$\mathbf{P}_j = \mathbf{Y}_1^{-1}\mathbf{S}_j$$

5.3 Camera and hand-eye calibration

Camera calibration and hand-eye calibration are both embedded in the projection matrix $\mathbf{M}$, eq. (6), which is obtained by multiplying two matrices that have just been determined:

$$\mathbf{M} = \mathbf{M}_1\mathbf{Y}_1$$

which is the first equation in eq. (7). Therefore the intrinsic camera parameters and the rotation matrix between the camera and the robot hand are obtained by performing a QR-decomposition of $\mathbf{N}$ — the 2×3 sub-matrix of $\mathbf{M}$.

The intrinsic parameters are recovered under the form of a 2×2 low-diagonal matrix $\mathbf{A}$ as in eq.(5). This is a very general affine camera model with three parameters (a, b, and c) that can be specialized as follows:

- if the off-diagonal element b is small compared to the diagonal elements, then it can be neglected and the camera model is equivalent to weak perspective, and
- if the off-diagonal term is not small, then the camera model is equivalent to paraperspective.

6 Experiments

In order to validate the method we performed several similar experiments. A camera mounted onto a robot observes a 3-D object from 12 different positions. Figure 2 shows the first (a) and the tenth (b) image in this sequence. The object consists of a parallelepiped with black rectangles on its faces. The exact position and size of these black rectangles are not known. The size of this parallelepiped is of $5 \times 4 \times 4$ cm and it is at approximatively 50 cm away from the camera. Therefore, the perspective effect is weak and the camera model can be described by an affine projection.

Interest points were detected and tracked over the image sequence: There are 132 tracked points belonging to three different faces of the parallelepiped. In this sequence, the matching process has been done by hand because of the very repetitive motif, but for other real sequences we have used the method described in [15].

Figure 2 shows the results of reconstructing the 3-D shape of this object with two methods: the method described in this paper and a method that uses a perspective camera model [1]. For each method Figure 2 shows a top view and a lateral view: (c) and (d) correspond to the method described in this paper and (e) and (f) correspond to the method described in [1]. There are two important differences in between these two methods: (i) the former uses an uncalibrated camera while the latter uses a calibrated camera and (ii) the former method assumes an affine camera model while the latter method considers a perspective camera model.

It is not easy to compare the two reconstruction results because the 3-D reference frame used by the two methods are different. In the case of the method described in this paper the 3-D frame is the Euclidean frame associated with the robot hand. For the second method, the 3-D frame is a object centered frame whose orientation is aligned with the camera frame in its first position.

The camera parameters that are obtained with the above experiment are the followings:

$$\begin{aligned} \mathbf{N} &= k\mathbf{A}\mathbf{R}_{2\times 3} \\ &= 1.36 \begin{pmatrix} 0.58 & 0 \\ 0.06 & 1 \end{pmatrix} \begin{pmatrix} -0.97 & -0.12 & +0.18 \\ +0.12 & -0.99 & -0.02 \end{pmatrix} \end{aligned}$$

The scale factor k includes the effect of both average depth and the focal real focal length of the camera lens. Since the off diagonal element of matrix A can be neglected, the camera model can be approximated in this case by weak perspective projection. The non null diagonal element represents the aspect ratio between the horizontal and vertical pixel size.

7 Discussion

In this paper we described a method for recovering the 3-D Euclidean structure of a scene using an uncalibrated affine camera mounted onto a robot arm. The Euclidean information is provided by controlled robot motions with known parameters. We proved a sufficient condition for uniqueness (there must be at least two robot motions with non identical rotation axes) and we provided a very simple linear algebraic method for recovering 3-D shape and for determining — as a side effect — the intrinsic and extrinsic camera parameters. The intrinsic parameters allow one to determine whether it is a weak perspective or a paraperspective camera. The extrinsic parameters (a rotation matrix) correspond to the relative orientation between the robot hand frame and the camera.

The method that we described in this paper may be used whenever controlled motions are possible. This is the case with cameras mounted on robots but also with active stereo heads. Similarly a number of authors have shown that the camera calibration problem is simpler if some special motions can be performed in space [5]. It may be a problem, however, to rotate an uncalibrated camera around its optical axis. The method described in this paper provides some insights for the case where the actual axis of rotation is not perfectly aligned with the optical axis.

References

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

[2] N. Cui, J. Weng, and P. Cohen. Extended structure and motion analysis from monocular image sequences. In *Proc. Third International Conference on Computer Vision*, pages 222–229, Osaka, Japan, December 1990.

[3] O. D. Faugeras, Q. T. Luong, and S. J. Maybank. Camera self-calibration: Theory and experiments. In

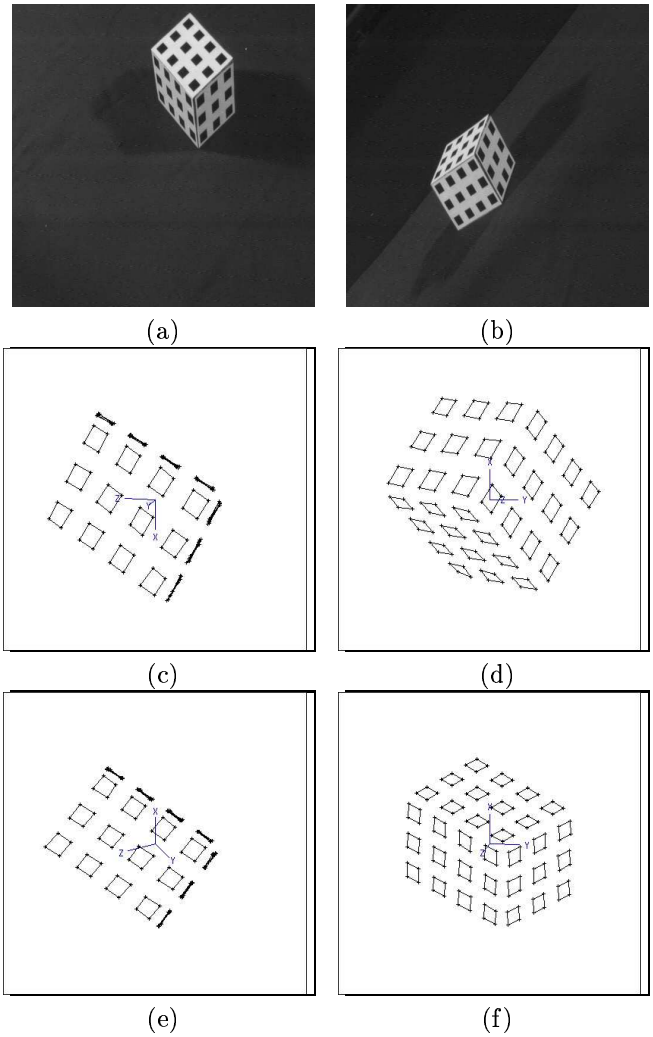

Figure 2: This figure shows two images (a) and (b), out of a sequence of 12 images grabbed with a moving camera mounted onto a robot arm, and the result of reconstruction: top view (c) and general (d) of the 3-D shape obtained with the method described in this paper, and top view (e) and general view (f) of the same 3-D shape using a perspective camera model.

G. Sandini, editor, *Computer Vision – ECCV 92, Proceedings Second European Conference on Computer Vision, Santa Margherita Ligure, May 1992*, pages 321–334. Springer Verlag, May 1992.

[4] R. I. Hartley. Euclidean reconstruction from uncalibrated views. In Mundy Zisserman Forsyth, editor, *Applications of Invariance in Computer Vision*, pages 237–256. Springer Verlag, Berlin Heidelberg, 1994.

[5] R. I. Hartley. Self-calibration from multiple views with a rotating camera. In *Proc. Third European Conference on Computer Vision*, pages 471–478, Stockholm, Sweden, May 1994.

[6] R. Horaud and F. Dornaika. Hand-eye calibration. *International Journal of Robotics Research*, 14(3):195–210, June 1995.

[7] R. Horaud, R. Mohr, F. Dornaika, and B. Boufama. The advantage of mounting a camera onto a robot arm. In *Proceedings of Europe-China Workshop on Geometrical Modelling and Invariants for Computer Vision*, pages 206–213, Xi'an, China, April 1995. Xidan University Press.

[8] C. J. Poelman and T. Kanade. A paraperspective factorization method for shape and motion recovery. In Jan-Olof Eklundh, editor, *Computer Vision – ECCV 94, Proceedings Third European Conference on Computer Vision*, volume 2, pages 97–108. Springer Verlag, Stockholm, Sweden, May 1994.

[9] L. Quan. Self-calibration of an affine camera. In *Proc. of Sixth International Conference on Computer Analysis of Images and Patterns*, pages 448–455, Prague, Czech Republic, September 1995.

[10] R. Szelinski and S. B. Kang. Recovering 3-D shape and motion from image streams using non-linear least squares. Technical Report CRL 93/3, Digital – Cambridge Research Laboratory, March 1993.

[11] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 9(2):137–154, November 1992.

[12] T. Viéville and Q-T. Luong. Computing motion and stucture in image sequences without calibration. In *Proc. of the Twelfth International Conference on Pattern Recognition*, pages 420–425, Jerusalem, Israel, October 1994. IEEE Computer Society Press.

[13] D. Weinshall. Model-based invariants for 3-d vision. *International Journal of Computer Vision*, 10(1):27–42, February 1993.

[14] D. Weinshall and C. Tomasi. Linear and incremental acquisition of invariant shape models from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(5):512–517, May 1995.

[15] Z. Zhang, R. Deriche, O. D. Faugeras, and Q-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78(1–2):87–119, October 1995.