



Title	Estimation of the camera pose from image point correspondences through the essential matrix and convex optimization
Author(s)	Chesi, G
Citation	IEEE International Conference on Robotics and Automation (ICRA), Kobe, Japan, 12 -17 May 2009. In I E E International Conference on Robotics and Automation Proceedings, 2009, p. 35-40
Issued Date	2009
URL	http://hdl.handle.net/10722/62039
Rights	I E E E International Conference on Robotics and Automation Proceedings. Copyright © I E E E, Computer Society.

Estimation of the camera pose from image point correspondences through the essential matrix and convex optimization

Graziano Chesi

Abstract—Estimating the camera pose in stereo vision systems is an important issue in computer vision and robotics. One popular way to handle this problem consists of determining the essential matrix which minimizes the algebraic error obtained from image point correspondences. Unfortunately, this search amounts to solving a nonconvex optimization, and the existing methods either rely on some approximations in order to get rid of the non-convexity or provide a solution that may be affected by the presence of local minima. This paper proposes a new approach to address this search without presenting such problems. In particular, we show that the sought essential matrix can be obtained by solving a convex optimization built through a suitable reformulation of the considered minimization via appropriate techniques for representing polynomials. Numerical results show the proposed approach compares favorably with some standard methods in both cases of synthetic data and real data.

Index Terms—Point correspondences, Essential matrix, Camera pose, Convex optimization.

I. INTRODUCTION

Numerous tasks in computer vision and robotics necessitate to estimate the camera pose in stereo vision systems, i.e. the relative motion existing between two cameras observing a common set of object features. For instance, the camera pose is needed in visual servoing to determine the robot control law, as in position-based visual servoing (see e.g. [1], [2]) where the camera pose is required to define the feedback error, image-based visual servoing (see e.g. [3], [4]) where the camera pose is used to estimate the depth of the points employed in the definition of the image jacobian, hybrid methods such as 2 1/2 D visual servoing (see e.g. [5], [6]) where the camera pose is required to define some parts of the feedback error, and path-planning techniques (see e.g. [7], [8]) where the camera pose is needed to calculate the camera trajectory in the 3D space. See also [9] and references therein for further uses of the camera pose.

Typically the camera pose is computed from the available estimates of some point correspondences between two camera views and the camera intrinsic parameters. A simple technique for performing this computation consists of estimating the essential matrix through the eight point algorithm and then estimating the camera pose from the found essential matrix [10], [11]. This technique estimates the essential matrix through the minimization of an algebraic error, and amounts to solving linear least-squares problems which can be solved via SVD. An analogous technique exploits the homography matrix relative to a virtual plane attached to

the object, which is estimated through the minimization of an algebraic error, and then decomposed in order to recover the camera pose [6]. More accurate techniques such as [12]–[15] involve the minimization of algebraic/geometric errors directly parameterized by the camera pose, for instance by minimizing these errors over the essential matrices manifold. However, these techniques either rely on some approximations in order to get rid of nonlinear terms, or provide a solution that may be affected by the presence of local minima since the optimization is nonconvex.

In this paper we propose a new approach to estimate the camera pose, in particular we propose a new way to solve the minimization of the algebraic error over the essential matrices manifold. The proposed approach exploits the square matrix representation (SMR) introduced in [16] to solve some control problems. By using the SMR we find the sought camera pose by solving a convex optimization formulated as minimization of a linear cost subject to a linear matrix inequality (LMI) constraint. Numerical investigations are carried out with both synthetic and real data, in the former case the ideal data are corrupted by introducing both image noise and calibration errors. The results show that the proposed approach compares favorably with linear methods as well as searches over the essential matrices manifold via gradient algorithm.

The paper is organized as follows. Section II introduces some preliminaries and the problem formulation. Section III describes the proposed idea based on a reformulation of the problem via the SMR and its solution through LMI optimizations. Section IV reports the numerical results. Lastly, Section V concludes the paper with some final discussions.

II. PROBLEM FORMULATION

The notation exploited in this paper is as follows:

- $SO(3)$: set of all rotation matrices of size 3×3 ;
- \mathbf{e}_i : null vector of size 3×1 with the i th component set to 1;
- $\mathbf{0}_{m \times n}$: null matrix of size $m \times n$;
- \mathbf{I}_n : identity matrix of size $n \times n$;
- $tr(\mathbf{X})$: trace of a square matrix \mathbf{X} ;
- \mathbf{X}^T : transpose of a matrix \mathbf{X} ;
- $\|\mathbf{x}\|$: euclidean norm of vector \mathbf{x} ;
- $[\mathbf{x}]_{\times}$: skew-symmetric matrix of a vector \mathbf{x} of size 3×1 , i.e.

$$[\mathbf{x}]_{\times} = \begin{pmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{pmatrix} \quad (1)$$

where x_1, x_2, x_3 are the components of \mathbf{x} ;

G. Chesi is with the Department of Electrical and Electronic Engineering, University of Hong Kong. Contact information: please see <http://www.eee.hku.hk/~chesi>

- s.t.: subject to.

Let us consider a stereo vision system, i.e. a pair of cameras observing a common object, and let F^{abs} be an absolute frame in the 3D space. We denote with $F = (\mathbf{O}, \mathbf{c})$ the frame of one of the two cameras of the stereo vision system expressed with respect to the absolute frame F^{abs} , where $\mathbf{O} \in SO(3)$ is a rotation matrix which defines the orientation, and $\mathbf{c} \in \mathbb{R}^3$ is a vector which defines the translation. Similarly we denote with $F^* = (\mathbf{O}^*, \mathbf{c}^*)$ the frame of the other camera of the stereo vision system.

Let $\mathbf{q}_1, \dots, \mathbf{q}_N \in \mathbb{R}^3$ be a set of 3D points expressed with respect to the absolute frame F^{abs} . The i th 3D point \mathbf{q}_i projects onto the camera frame F at the point $\mathbf{p}_i = (p_{i,1}, p_{i,2}, 1)^T \in \mathbb{R}^3$ expressed in homogeneous coordinates and given by

$$d_i \mathbf{p}_i = \mathbf{A} \mathbf{O}^T (\mathbf{q}_i - \mathbf{c}) \quad (2)$$

where d_i is the depth of the point with respect to F , and $\mathbf{A} \in \mathbb{R}^{3 \times 3}$ is the upper-triangular matrix containing the camera intrinsic parameters:

$$\mathbf{A} = \begin{pmatrix} f_x & s & u_x \\ 0 & f_y & u_y \\ 0 & 0 & 1 \end{pmatrix}, \quad (3)$$

being $f_x, f_y \in \mathbb{R}$ the scaling parameters, $u_x, u_y \in \mathbb{R}$ the coordinates of the principal point, and $s \in \mathbb{R}$ the skew parameter. Similarly, the i th 3D point \mathbf{q}_i projects onto the camera frame F^* at the point $\mathbf{p}_i^* = (p_{i,1}^*, p_{i,2}^*, 1)^T \in \mathbb{R}^3$ expressed in homogeneous coordinates and given by

$$d_i^* \mathbf{p}_i^* = \mathbf{A}^* \mathbf{O}^{*T} (\mathbf{q}_i - \mathbf{c}^*) \quad (4)$$

where d_i^* is the depth of the point with respect to F^* , and $\mathbf{A}^* \in \mathbb{R}^{3 \times 3}$ is the upper-triangular matrix containing the intrinsic parameters of F^* analogously to (3).

In this paper we address the problem of estimating the camera pose of the considered stereo vision system, i.e. the pair

$$(\mathbf{R}_{true}, \mathbf{t}_{true}) \in SO(3) \times \mathbb{R}^3 \quad (5)$$

where \mathbf{R}_{true} is the rotation matrix which expresses the relative orientation between F and F^* , and \mathbf{t}_{true} is the unit-norm vector which expresses the normalized translation between F and F^* (in the sequel we will assume that we consider camera pose with non-zero translational components, for which \mathbf{t}_{true} can always be defined). In particular, we consider that the camera pose (5) is expressed with respect to the camera frame F , and hence \mathbf{R}_{true} is given by the relationship

$$\mathbf{R}_{true} = \mathbf{O}^T \mathbf{O}^* \quad (6)$$

while \mathbf{t}_{true} is given by

$$\mathbf{t}_{true} = \frac{\mathbf{O}^T (\mathbf{c}^* - \mathbf{c})}{\|\mathbf{O}^T (\mathbf{c}^* - \mathbf{c})\|}. \quad (7)$$

Therefore, the problem consists on estimating \mathbf{R}_{true} and \mathbf{t}_{true} by exploiting only the following data:

- 1) the image projections of the 3D points $\mathbf{q}_1, \dots, \mathbf{q}_N$ onto F and F^* , i.e. the image points $\mathbf{p}_1, \mathbf{p}_1^*, \dots, \mathbf{p}_N, \mathbf{p}_N^*$;

- 2) the camera intrinsic parameters in \mathbf{A} and \mathbf{A}^* .

In the sequel we will consider, for ease of notation, that $\mathbf{p}_1, \mathbf{p}_1^*, \dots, \mathbf{p}_N, \mathbf{p}_N^*$, \mathbf{A} and \mathbf{A}^* represent the available estimate of their respective quantities, which are corrupted by image noise and calibration error respectively.

Let us observe that we are interested in estimating the normalized translation rather than the translation itself because, by exploiting only the image projections $\mathbf{p}_1, \mathbf{p}_1^*, \dots, \mathbf{p}_N, \mathbf{p}_N^*$ the translation can be estimated only up to a scale factor, which stands for the unknown distance between the origins of the camera frames F and F^* .

It is assumed that the 3D points \mathbf{q}_i are non-coplanar, and that the camera pose is not purely rotational (because in this case the essential matrix is identically zero, and hence cannot be used to estimate the camera pose).

III. PROPOSED APPROACH

A. Essential matrix and algebraic error

A possible way to estimate the camera pose in stereo vision systems consists of minimizing the algebraic error provided by the essential matrix. Specifically, let

$$\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R} \quad (8)$$

be the essential matrix of the considered stereo vision system. Then, the algebraic error is defined as (see for instance [11], [17] and references therein)

$$\eta = \sum_{i=1}^N (\mathbf{m}_i^T \mathbf{E} \mathbf{m}_i^*)^2 \quad (9)$$

where \mathbf{m}_i and \mathbf{m}_i^* are the available estimates of the image points in normalized-camera coordinates given by

$$\begin{aligned} \mathbf{m}_i &= \mathbf{A}^{-1} \mathbf{p}_i \\ \mathbf{m}_i^* &= \mathbf{A}^{*-1} \mathbf{p}_i^*. \end{aligned} \quad (10)$$

The problem we aim to solve in this paper is hence the following:

$$\begin{aligned} &\min_{\mathbf{R}, \mathbf{t}} \eta \\ \text{s.t. } &\mathbf{R} \in SO(3), \|\mathbf{t}\| = 1. \end{aligned} \quad (11)$$

Let us observe that in (11) we take into account the structure of the essential matrix, i.e. its dependence on the rotation \mathbf{R} and translation \mathbf{t} . Let us also observe that the free scale factor of the essential matrix, and hence the free scale factor of the algebraic error, is fixed by our choice to consider that the translation \mathbf{t} is a unit-norm vector.

As it is well known, the minimization (11) is a non-convex optimization problem. In this paper we want to investigate how this minimization can be addressed by using convex optimization tools, in particular we provide an extension of our approach in [18] through relaxations for constrained polynomial programming.

Before proceeding it is worthwhile to explain that more accurate estimates of the camera pose can be found by minimizing the geometric error rather than the algebraic error. In this paper, however, we are interested in the algebraic error, since it can provide satisfactory estimates and since it represents a reasonable trade-off between accuracy and complexity.

B. SMR of polynomials

The approach proposed in this paper is based on the square matrixial representation (SMR) of polynomials introduced in [16] to solve some control problems. Let $p(\mathbf{x})$ be a polynomial of degree $2m$ in the real vector $\mathbf{x} = (x_1, \dots, x_n)^T \in \mathbb{R}^n$, i.e.

$$p(\mathbf{x}) = \sum_{\substack{i_1 + \dots + i_n \leq 2m \\ i_1 \geq 0, \dots, i_n \geq 0}} p_{i_1, \dots, i_n} x_1^{i_1} \dots x_n^{i_n} \quad (12)$$

for some coefficients $p_{i_1, \dots, i_n} \in \mathbb{R}$. Then, according to the SMR, $p(\mathbf{x})$ can be expressed as

$$p(\mathbf{x}) = \mathbf{y}(\mathbf{x})^T \mathbf{P}(\boldsymbol{\alpha}) \mathbf{y}(\mathbf{x}) \quad (13)$$

$$\mathbf{P}(\boldsymbol{\alpha}) = \mathbf{P} + \mathbf{Q}(\boldsymbol{\alpha}) \quad (14)$$

where $\mathbf{y}(\mathbf{x})$ is any vector containing a base for the polynomials of degree m in \mathbf{x} , and hence can be simply chosen as the set of monomials of degree less than or equal to m in \mathbf{x} , for example via

$$\mathbf{y}(\mathbf{x}) = (x_1^m, x_1^{m-1}x_2, x_1^{m-1}x_3, \dots)^T. \quad (15)$$

The matrix \mathbf{P} is any symmetric matrix such that

$$p(\mathbf{x}) = \mathbf{y}(\mathbf{x})^T \mathbf{P} \mathbf{y}(\mathbf{x}) \quad (16)$$

which can be simply obtained via trivial coefficient comparisons. The vector $\boldsymbol{\alpha}$ is a vector of free parameters, and the matrix function $\mathbf{Q}(\boldsymbol{\alpha})$ is a linear parametrization of the set

$$\mathcal{Q} = \{\mathbf{Q} = \mathbf{Q}^T : \mathbf{y}(\mathbf{x})^T \mathbf{Q} \mathbf{y}(\mathbf{x}) = 0 \quad \forall \mathbf{x}\} \quad (17)$$

which can be computed through standard linear algebra techniques for parameterizing linear spaces. The matrices \mathbf{P} and $\mathbf{P}(\boldsymbol{\alpha})$ are known as SMR matrix and complete SMR matrix of $p(\mathbf{x})$. The dimension of $\mathbf{y}(\mathbf{x})$ is given by

$$d_1(n, m) = \frac{(n+m)!}{n!m!} \quad (18)$$

while the dimension of $\boldsymbol{\alpha}$ is

$$d_2(n, m) = \frac{1}{2}d_1(n, m)(d_1(n, m) + 1) - d_1(n, 2m). \quad (19)$$

See also [19]–[21].

The usefulness of the SMR is to allow one to establish positivity of polynomials via convex optimizations. Indeed, $p(\mathbf{x})$ is positive if there exists $\boldsymbol{\alpha}$ such that

$$\mathbf{P}(\boldsymbol{\alpha}) \geq 0 \quad (20)$$

which is a linear matrix inequality (LMI), and whose feasible solutions can be found via convex optimizations, see for instance [22].

C. Estimate via convex optimization

In order to solve (11), we parameterize the rotation \mathbf{R} through polynomials. This can be done, for instance, by using Euler parameter as follows:

$$\begin{cases} \mathbf{R} = \boldsymbol{\Omega}(\mathbf{r}) \\ \|\mathbf{r}\| = 1 \end{cases} \quad (21)$$

where $\mathbf{r} = (r_1, \dots, r_4)^T \in \mathbb{R}^4$ is a unit-norm vector which represents the Euler parameter of \mathbf{R} , and $\boldsymbol{\Omega}(\mathbf{r})$ is quadratic matrix function, see for example [23] for details. This parametrization is complete, in the sense that:

- 1) for any unit-norm vector $\mathbf{r} \in \mathbb{R}^4$ it follows that $\boldsymbol{\Omega}(\mathbf{r})$ is a rotation matrix in $SO(3)$;
- 2) for any rotation matrix $\mathbf{R} \in SO(3)$ there exists a vector $\mathbf{r} \in \mathbb{R}^4$ with unit-norm such that $\mathbf{R} = \boldsymbol{\Omega}(\mathbf{r})$.

This allows us to reformulate the problem (11) by using polynomials. Indeed, let us define the new variable

$$\mathbf{x} = \begin{pmatrix} \mathbf{r} \\ \mathbf{t} \end{pmatrix} \quad (22)$$

and let us rewrite the term $\mathbf{m}_i^T \mathbf{E} \mathbf{m}_i^*$ as

$$\begin{aligned} \mathbf{m}_i^T \mathbf{E} \mathbf{m}_i^* &= \mathbf{m}_i^T [\mathbf{t}]_{\times} \mathbf{R} \mathbf{m}_i^* \\ &= \mathbf{m}_i^T [\mathbf{t}]_{\times} \boldsymbol{\Omega}(\mathbf{r}) \mathbf{m}_i^* \\ &= f_i(\mathbf{x}). \end{aligned} \quad (23)$$

The quantity $f_i(\mathbf{x})$ is a cubic function of the new variable \mathbf{x} . In fact, $\mathbf{m}_i^T \mathbf{E} \mathbf{m}_i^*$ is a bilinear function of \mathbf{R} and \mathbf{t} , i.e. a linear function of \mathbf{R} for fixed \mathbf{t} and a linear function of \mathbf{t} for fixed \mathbf{R} , moreover \mathbf{R} is a quadratic function of \mathbf{r} .

Let us consider now the cost function of (11). We have that η can be rewritten as

$$\begin{aligned} \eta &= \sum_{i=1}^N (\mathbf{m}_i^T \mathbf{E} \mathbf{m}_i^*)^2 \\ &= \sum_{i=1}^N f_i(\mathbf{x})^2 \end{aligned} \quad (24)$$

where each term $f_i(\mathbf{x})^2$ is a polynomial of degree six in \mathbf{x} (specifically, a polynomial of degree four in \mathbf{r} and a quadratic function of \mathbf{t}).

Therefore, problem (11) can be reformulated as

$$\begin{aligned} \min_{\mathbf{x}} \quad & \sum_{i=1}^N f_i(\mathbf{x})^2 \\ \text{s.t.} \quad & \|\mathbf{C}_1 \mathbf{x}\| = 1, \|\mathbf{C}_2 \mathbf{x}\| = 1 \end{aligned} \quad (25)$$

where the matrices \mathbf{C}_1 and \mathbf{C}_2 are given by

$$\mathbf{C}_1 = (\mathbf{I}_4 \mathbf{0}_{4 \times 3}), \quad \mathbf{C}_2 = (\mathbf{0}_{3 \times 4} \mathbf{I}_3). \quad (26)$$

Now, let us observe that (25) can be rewritten as

$$\begin{aligned} & - \min_s s \\ \text{s.t.} \quad & \sum_{i=1}^N f_i(\mathbf{x})^2 + s \geq 0 \quad \forall \mathbf{x} : \begin{cases} \|\mathbf{C}_1 \mathbf{x}\| = 1 \\ \|\mathbf{C}_2 \mathbf{x}\| = 1 \end{cases} \end{aligned} \quad (27)$$

where $s \in \mathbb{R}$ is an additional variable.

At this point, let us introduce auxiliary polynomials $g_0(\mathbf{x})$, $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$. We select $g_0(\mathbf{x})$ in order to satisfy the condition

$$g_0(\mathbf{x}) \geq 1 \quad \forall \mathbf{x}. \quad (28)$$

The constrained minimization in (27) can be addressed via

$$\begin{aligned} & - \min_{s, g_1(\mathbf{x}), g_2(\mathbf{x})} s \\ \text{s.t. } & u(\mathbf{x}, s, g_1(\mathbf{x}), g_2(\mathbf{x})) \geq 0 \quad \forall \mathbf{x} \end{aligned} \quad (29)$$

where

$$\begin{aligned} u(\mathbf{x}, s, g_1(\mathbf{x}), g_2(\mathbf{x})) &= g_0(\mathbf{x}) \left(\sum_{i=1}^N f_i(\mathbf{x})^2 + s \right) \\ &+ \sum_{i=1}^2 g_i(\mathbf{x}) (\|\mathbf{C}_i \mathbf{x}\|^2 - 1) \end{aligned} \quad (30)$$

which is a polynomial in \mathbf{x} and an affine linear function of $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$. In fact, let us observe that $u(\mathbf{x}, s, g_1(\mathbf{x}), g_2(\mathbf{x})) \geq 0$ reduces to $\sum_{i=1}^N f_i(\mathbf{x})^2 + s \geq 0$ for all \mathbf{x} satisfying $\|\mathbf{C}_1 \mathbf{x}\| = 1$ and $\|\mathbf{C}_2 \mathbf{x}\| = 1$.

The next step consists of exploiting the SMR of polynomials introduced in Section III-B. Let us express the polynomials $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$ as

$$g_i(\mathbf{x}) = \mathbf{g}_i^T \mathbf{y}_i(\mathbf{x}) \quad \forall i = 1, 2 \quad (31)$$

where $\mathbf{y}_1(\mathbf{x}), \mathbf{y}_2(\mathbf{x})$ are vectors containing suitable polynomial bases, and $\mathbf{g}_1, \mathbf{g}_2$ are the vectors containing the coefficients of $g_1(\mathbf{x})$ and $g_2(\mathbf{x})$ with respect to these bases. Then, let $\mathbf{U}(\boldsymbol{\alpha}, s, \mathbf{g}_1, \mathbf{g}_2)$ be a complete SMR matrix of $u(\mathbf{x}, s, g_1(\mathbf{x}), g_2(\mathbf{x}))$ built with respect to a vector $\mathbf{y}(\mathbf{x})$ containing a suitable polynomial base, i.e. such that

$$u(\mathbf{x}, s, g_1(\mathbf{x}), g_2(\mathbf{x})) = \mathbf{y}(\mathbf{x})^T \mathbf{U}(\boldsymbol{\alpha}, s, \mathbf{g}_1, \mathbf{g}_2) \mathbf{y}(\mathbf{x}). \quad (32)$$

The vector $\boldsymbol{\alpha}$ is a free vector of suitable dimension which parameterizes all possible SMR matrices with respect to $\mathbf{y}(\mathbf{x})$ describing $u(\mathbf{x}, s, g_1(\mathbf{x}), g_2(\mathbf{x}))$. Such a matrix $\mathbf{U}(\boldsymbol{\alpha}, s, g_1(\mathbf{x}), g_2(\mathbf{x}))$ is an affine linear function of $\boldsymbol{\alpha}$, s , $g_1(\mathbf{x})$, and $g_2(\mathbf{x})$, which can be simply obtained via trivial coefficient comparisons and standard linear algebra procedures for parameterizing linear spaces, see for instance [19], [21] for details. Problem (29) becomes

$$\begin{aligned} & - \min_{\boldsymbol{\alpha}, s, \mathbf{g}_1, \mathbf{g}_2} s \\ \text{s.t. } & \mathbf{y}(\mathbf{x})^T \mathbf{U}(\boldsymbol{\alpha}, s, \mathbf{g}_1, \mathbf{g}_2) \mathbf{y}(\mathbf{x}) \geq 0 \quad \forall \mathbf{x}. \end{aligned} \quad (33)$$

Let us observe that the positivity of the constraint in (33) is ensured by the positivity of the inner matrix, hence obtaining

$$\begin{aligned} & - \min_{\boldsymbol{\alpha}, s, \mathbf{g}_1, \mathbf{g}_2} s \\ \text{s.t. } & \mathbf{U}(\boldsymbol{\alpha}, s, \mathbf{g}_1, \mathbf{g}_2) \geq 0. \end{aligned} \quad (34)$$

The minimization in (34) belongs to the class of convex optimizations. In fact, both cost function and constraint set in (34) are convex since the former is a linear function and the latter is the feasible set of an LMI, see for instance [22]. Therefore, the advantage of the new formulation (34) over the original formulation (25) is hence that (34) is a convex optimization whereas (25) is a non-convex one. Moreover, the solution of (34) is guaranteed to be a lower bound of the solution of (27), and the conservatism of this lower bound can be regulated by increasing the degree of the introduced polynomials.

Once (34) has been solved, we build the estimate of the camera pose as follows. Let $\bar{\mathbf{x}}$ be a non-zero vector satisfying

$$\mathbf{U}(\bar{\boldsymbol{\alpha}}, \bar{s}, \bar{\mathbf{g}}_1, \bar{\mathbf{g}}_2) \mathbf{y}(\bar{\mathbf{x}}) = \mathbf{0}_l \quad (35)$$

where l is the size of the matrix $\mathbf{U}(\bar{\boldsymbol{\alpha}}, \bar{s}, \bar{\mathbf{g}}_1, \bar{\mathbf{g}}_2)$, and $\bar{\boldsymbol{\alpha}}, \bar{s}, \bar{\mathbf{g}}_1, \bar{\mathbf{g}}_2$ are the minimizers returned by the solver used for (34). Then, the sought estimate is given by

$$\begin{aligned} \hat{\mathbf{R}} &= \Omega \left(\frac{\mathbf{C}_1 \bar{\mathbf{x}}}{\|\mathbf{C}_1 \bar{\mathbf{x}}\|} \right) \\ \hat{\mathbf{t}} &= \frac{\mathbf{C}_2 \bar{\mathbf{x}}}{\|\mathbf{C}_2 \bar{\mathbf{x}}\|}. \end{aligned} \quad (36)$$

The vectors $\bar{\mathbf{x}}$ satisfying (35) can be computed via linear algebra operations as explained in [24]. In case there exist at least two linearly independent vectors $\bar{\mathbf{x}}$ satisfying (35), the best one is selected as the vector which provide the camera pose for which all object points are kept in front of the cameras. Let us observe that the scale factor of the vector $\bar{\mathbf{x}}$, which is undefined by the relation (35), does not affect the estimate (36).

IV. NUMERICAL RESULTS

In this section we present some results obtained with both synthetic and real data.

A. Synthetic data

We have generated 400 camera-object configurations by randomly locating the camera frames F and F^* in the 3D space and by randomly generating some object points inside a sphere of radius 150 mm centered along the optical axis of F^* at a distance of 400 mm from its origin under the constraint that the object points lie in the field of view of F and F^* . 100 of these configurations have $N = 7$ point-correspondences, other 100 have $N = 8$, other 100 have $N = 9$, and the remaining 100 have $N = 12$. The screen size is 800×600 pixels, and the intrinsic parameters in \mathbf{A} and \mathbf{A}^* are $f_x = 800$, $f_y = 600$, $s = 20$, $u_x = 400$, $u_y = 300$. Figure 1 shows one of these configurations and the corresponding camera view.

Then, we have introduced uncertainty on the available data of each configuration as follows:

- 1) we have introduced image noise by adding to each component of all image points $\mathbf{p}_1, \dots, \mathbf{p}_N$ random variables with uniform distribution in the interval $[-1, 1]$ pixels;
- 2) we have introduced calibration error by multiplying the scaling parameters, principal point, and skew parameter in (3) by random variables with uniform distribution in the interval $[0.9, 1.1]$.

Hence, we have evaluated the estimation errors

$$\begin{aligned} \epsilon_r &= \arccos \left(\frac{\text{tr}(\mathbf{R}_{true}^T \mathbf{R}) - 1}{2} \right) \\ \epsilon_t &= \arccos(\mathbf{t}_{true}^T \mathbf{t}) \end{aligned} \quad (37)$$

which are, respectively, the angle in the exponential coordinates representation of the rotation matrix $\mathbf{R}_{true}^T \mathbf{R}$ and the angle between the translation vectors \mathbf{t}_{true} and \mathbf{t} . These errors range from 0 (perfect estimation) to π (worst-case estimation).

The proposed approach is compared with the following algorithms:

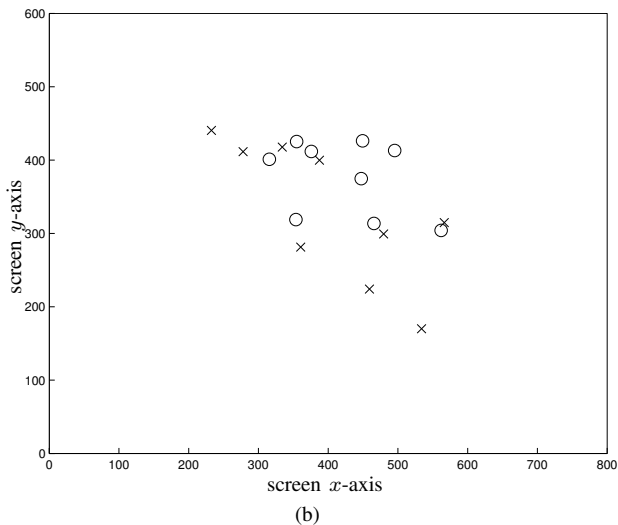
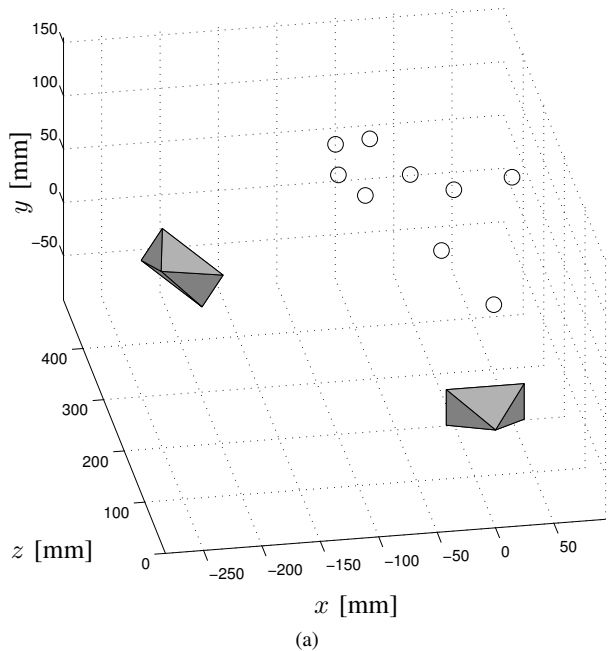


Fig. 1. Synthetic data. Figure (a) shows a camera-object configuration with $N = 9$, while figure (b) shows the image points in F (“o” marks) and F^* (“x” marks).

- 1) LIN (for the case of $N \geq 8$): linear least-squares where the unknown variables are the entries of \mathbf{E} . From the found estimate, \mathbf{R} and \mathbf{t} are computed as the rotation matrix and the unitary-norm vector that better approximate $\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R}$ and keep the object points in front of the cameras, see [10], [11], [17];
- 2) LIN (for the case of $N = 7$): similar to the previous case, but this time the minimizer of the linear least-squares is a matrix pencil, and by imposing the rank constraint there are up to three solutions. The one that better approximates the singular values property of the essential matrix is selected as explained in [13], [25];
- 3) EMMG: minimization of the algebraic error over the essential matrices manifold via gradient algorithm using as initialization the solution provided by LIN (in



Fig. 2. Real data. There are $N = 8$ point correspondences in the images pair (a)–(b), and $N = 12$ in the images pair (c)–(d).

both cases of $N \geq 8$ and $N = 7$).

Table I shows the average of these estimation errors. As we can see, the proposed approach favorably compares with the other methods. The reason is that the LIN methods do not search on the manifold of the essential matrices, indeed the structure of the essential matrix is imposed after solving the linear least-squares, while the EMMG method is affected by the presence of local minima.

N	method	ϵ_r	ϵ_t
7	LIN	0.391	0.353
7	EMMG	0.382	0.285
7	this paper	0.235	0.150
8	LIN	0.324	0.282
8	EMMG	0.293	0.253
8	this paper	0.152	0.097
9	LIN	0.192	0.153
9	EMMG	0.186	0.137
9	this paper	0.114	0.073
12	LIN	0.122	0.083
12	EMMG	0.106	0.067
12	this paper	0.085	0.057

TABLE I
AVERAGE ESTIMATION ERRORS WITH SYNTHETIC DATA (EXPRESSED IN RADIANS).

Regarding the computational time, the averages for the performed estimations are 0.01 seconds for LIN, 0.23 seconds for EMMG, and 1.15 seconds for our approach. Hence, our approach is the slowest among the ones considered, however its estimates are the most accurate as shown in Table I, moreover faster implementations of the proposed approach can be obtained by using C instead of Matlab.

B. Results with real data

In this section we report some results obtained with real data. We have considered the general scenarios shown in Figure 2. In particular, for the pictures in Figure 2a–b we have used a Canon digital camera with resolution 1600×1200 pixels, and we have considered $N = 8$ point correspondences. For the pictures in Figure 2c–d we have used a Logitech webcam 960×720 pixels, and we have considered $N = 12$ point correspondences. In both cases an estimate of the corresponding intrinsic parameters has been obtained by using standard calibration techniques.

Contrary to the case of synthetic data, the true camera displacement is unknown with real data, and therefore we have compared the various methods by using the found minimum of the algebraic error, in particular we have compared the normalized error

$$\zeta = \sqrt{\frac{1}{N} \sum_{i=1}^N (\mathbf{m}_i^T [\mathbf{t}]_{\times} \mathbf{R} \mathbf{m}_i^*)^2} \quad (38)$$

where \mathbf{R} and \mathbf{t} are the estimates found by each method. Table II shows the results obtained with our approach and with the methods described in Section IV-A. As we can see, the proposed approach provides significantly smaller algebraic errors than the the other methods.

scenario	method	ζ
Figures 2a–b	LIN	0.0220
Figures 2a–b	EMMG	0.0020
Figures 2a–b	this paper	0.0010
Figures 2c–d	LIN	0.0930
Figures 2c–d	EMMG	0.0062
Figures 2c–d	this paper	0.0053

TABLE II
RESULTS WITH REAL DATA.

V. CONCLUSION

We have proposed a new approach for estimating the camera pose. This approach determines the essential matrix by minimizing the algebraic error through a convex optimization with LMI constraints, and is obtained by exploiting the SMR for representing polynomials. The advantages with respect to other methods for minimizing the algebraic error consist of not presenting local minima and not introducing approximations of nonlinear terms. Future work will investigate the possibility of reducing the numerical complexity, which presently represents the main drawback.

ACKNOWLEDGEMENT

The author would like to thank the Associate Editor and Reviewers for their time and comments.

REFERENCES

[1] B. Thuilot, P. Martinet, L. Cordesses, and J. Gallice, "Position based visual servoing: keeping the object in the field of vision," in *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington, D.C., 2002, pp. 1624–1629.

[2] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the field of view in eye-in-hand visual servoing: a switching approach," *IEEE Trans. on Robotics*, vol. 20, no. 5, pp. 908–913, 2004.

[3] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura, "Manipulator control with image-based visual servo," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 1991, pp. 2267–2272.

[4] F. Chaumette, "Potential problems of stability and convergence in image-based and position-based visual servoing," in *The confluence of vision and control*, G. H. D. Kriegman and A. Morse, Eds. Springer-Verlag, 1998, pp. 66–78.

[5] E. Malis, F. Chaumette, and S. Boudet, "2 1/2 D visual servoing," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 2, pp. 238–250, 1999.

[6] E. Malis and F. Chaumette, "2 1/2 D visual servoing with respect to unknown objects through a new estimation scheme of camera displacement," *Int. Journal of Computer Vision*, vol. 37, no. 1, pp. 79–97, 2000.

[7] Y. Mezouar and F. Chaumette, "Path planning for robust image-based control," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 4, pp. 534–549, 2002.

[8] G. Chesi and Y. S. Hung, "Global path-planning for constrained and optimal visual servoing," *IEEE Trans. on Robotics*, vol. 23, no. 5, pp. 1050–1060, 2007.

[9] G. Chesi and K. Hashimoto, "Camera pose estimation from less than eight points in visual servoing," in *IEEE Int. Conf. on Robotics and Automation*, New Orleans, Louisiana, 2004, pp. 733–738.

[10] J. Weng, T. Huang, and N. Ahuja, "Motion and structure from two perspective views: Algorithms, error analysis, and error estimation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, no. 5, pp. 451–476, 1989.

[11] R. Hartley and A. Zisserman, *Multiple view in computer vision*. Cambridge University Press, 2000.

[12] R. Deriche, Z. Zhang, Q.-T. Luong, and O. Faugeras, "Robust recovery of the epipolar geometry for an uncalibrated stereo rig," in *Proc. European Conf. on Computer Vision*, Stockholm, Sweden, 1994.

[13] Z. Zhang, "Determining the epipolar geometry and its uncertainty - a review," *Int. Journal of Computer Vision*, vol. 27, no. 2, pp. 161–195, 1998.

[14] U. Helmke, K. Huper, P.-Y. Lee, and J. Moore, "Essential matrix estimation using gauss-newton iterations on a manifold," *Int. Journal of Computer Vision*, vol. 74, no. 2, pp. 117–136, 2007.

[15] R. Hartley and F. Kahl, "Global optimization through searching rotation space and optimal estimation of the essential matrix," in *Proc. 11th IEEE Int. Conf. on Computer Vision*, Rio de Janeiro, Brazil, 2007.

[16] G. Chesi, A. Tesi, A. Vicino, and R. Genesio, "On convexification of some minimum distance problems," in *5th European Control Conf.*, Karlsruhe, Germany, 1999.

[17] O. Faugeras, *Three-Dimensional Computer Vision: A Geometric Viewpoint*. Cambridge: MIT Press, 1993.

[18] G. Chesi, "Camera displacement via constrained minimization of the algebraic error," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 370–375, 2009.

[19] G. Chesi, A. Garulli, A. Tesi, and A. Vicino, "Solving quadratic distance problems: an LMI-based approach," *IEEE Trans. on Automatic Control*, vol. 48, no. 2, pp. 200–212, 2003.

[20] G. Chesi, A. Tesi, A. Vicino, and R. Genesio, "A convex approach to a class of minimum norm problems," in *Robustness in Identification and Control*, A. Garulli, A. Tesi, and A. Vicino, Eds. London: Springer-Verlag, 1999, pp. 359–372.

[21] G. Chesi, A. Garulli, A. Tesi, and A. Vicino, *Homogeneous Polynomial Forms for Robustness Analysis of Uncertain Systems*. Springer (in press).

[22] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*. Philadelphia: SIAM, 1994.

[23] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton: CRC Press, 1994.

[24] G. Chesi, A. Garulli, A. Tesi, and A. Vicino, "Characterizing the solution set of polynomial systems in terms of homogeneous forms: an LMI approach," *Int. Journal of Robust and Nonlinear Control*, vol. 13, no. 13, pp. 1239–1257, 2003.

[25] T. Huang and A. Netravali, "Motion and structure from feature correspondences: A review," in *Proc. IEEE*, vol. 82, no. 2, 1994, pp. 252–268.