• U  C •
UNIVERSIDADE DE COIMBRA
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES
INSTITUTO DE SISTEMAS E ROBÓTICA
3030-290 COIMBRA, PORTUGAL

ARTICLE: Pose Estimation for General Cameras using Lines

Author:
Pedro MIRALDO
miraldo@isr.uc.pt

Co-Authors:
Helder ARAUJO
helder@isr.uc.pt

Nuno GONÇALVES
nunogon@isr.uc.pt

CONTENTS

Preprint submitted to the IEEE Trans. Cybernetics at 18 November, 2013.

Pose Estimation for General Cameras using Lines

Pedro Miraldo, Helder Araujo, and Nuno Gonçalves

***Abstract*—In this article we address the problem of pose estimation under the framework of generalized camera models. We propose a solution based on the knowledge of the coordinates of 3D straight lines (expressed in the world coordinate frame) and their corresponding image pixels. Previous approaches used the knowledge of the coordinates of 3D points (zero dimensional elements) and their corresponding images (zero dimensional elements). In this approach pixels belonging to the image of 3D lines are used. There is no need to establish correspondences between pixels and 3D points. Correspondences are established between 3D lines and their images. There is no need to identify individual pixels. The use of correspondences between pixels, that belong to the images of the 3D lines, and 3D lines facilitate the correspondence problem when compared to the use of world and image points. This is one of the contributions of the paper. The approach is both evaluated and validated using synthetic data and also real images.**

***Index Terms*—Absolute pose, generalized camera models, 3D straight lines.**

I. Introduction

The problem of pose computation consists on the estimation of six parameters (rotation and translation) that define the pose between the world and camera coordinate systems. A common approach is to use correspondences between 3D points and pixels. Most of the methods for pose estimation described in the literature were derived for the case of central camera models, eg. [7], [8], [9], [10], [11]. However, in the last decade, non-central cameras have been subject of a significant volume of research specially due to their wide field of view, image resolution, and also because they model new imaging systems. Significant part of the research concerns camera systems where reflections and/or refractions occur. Examples of those camera systems are non-central catadioptric cameras and also cameras in a water environment.

When considering non-central cameras, several models exist. For instance, non-central catadioptric cameras using quadric mirrors [6] or multiple planar mirrors [4]. Yet another example is the case of cameras where refraction has to be dealt with [5]. In this paper we address the pose problem for the case of general camera models. The goal of this paper is to describe a solution for pose, within the framework of the generalized camera model (proposed in [2], [3], see Fig. 1). In this framework, the camera model is formed by a set of arrays of parameters called *raxels*. A *raxel* is itself a set of parameters that associate image pixels to 3D straight lines. This model is appropriate for general cases, where an analytical representation of the camera is impossible or unfeasible. Several methods were proposed in the literature for the calibration of these models [3], [12], [13]. In this paper we refer to each 3D straight line, associated to each image pixel (generated from the general camera model), as "projection line".

The authors are with the Institute for Systems and Robotics, Department of Electrical and Computer Engineering, University of Coimbra, Portugal.

As far as we know, the problem of the absolute pose for general camera models was studied by Chen & Chang at [14], [15] and Schweighofer & Pinz at [16]. Both approaches use matching between known coordinates of 3D points (in the world coordinate frame) and their corresponding pixels in the image. Chen & Chang proposed a solution for the minimal case (where only three 3D points and its correspondent pixels are known) and then derived an algorithm that computes the pose in a least-squared-error manner, using the minimal case. Schweighofer & Pinz proposed an iterative globally optimal $O(n)$ solution to the problem. We also note that an alternative solution for the minimal case was proposed by Nís ter & Stewénius at [17]. In this paper, we address the non-minimal case.

The determination of point correspondences (between 3D points and their images) is still a difficult problem and current solutions are error-prone. In this paper, we want to relax this procedure by using coordinates of 3D straight lines (one dimensional elements) defined in the world coordinate system, instead of 3D points, which constitutes one of the advantages of the proposed approach.

Let us consider a 3D straight line with known coordinates. The image of this line is made up by a set of image pixels. Depending on the complexity of the imaging system, this set of pixels may consist on a non-continuous curve in the image plane. On the other hand, if the imaging system is smooth the association between pixels and "projection lines" is smooth [13], and the curve in the image must be continuous. As a result, and since we are considering the general case, we do not take into account any geometric constraints between pixels corresponding to the image of the same 3D straight line (expressed in the world coordinate system). We only assume that we have a set of pixels whose coordinates are known and which correspond to a given 3D straight line. The only requirement is the determination of the image of a 3D line. Then any pixel of that image can be used. The problem becomes, therefore, easier, since there is no need to establish correspondences between pixels and 3D point features.

Using the framework described in the previous paragraph we propose a novel solution to the problem of the pose estimation for general camera models, using lines, Sec. II. There is, a trivial two step solution that can be used to solve this problem. As a matter of fact and for general non-central cameras, 3D straight lines can be recovered using a single image. On the other hand, for more than three 3D straight lines represented in two different coordinate systems, it is possible to recover the rotation and translation parameters that define the rigid transformation between both.

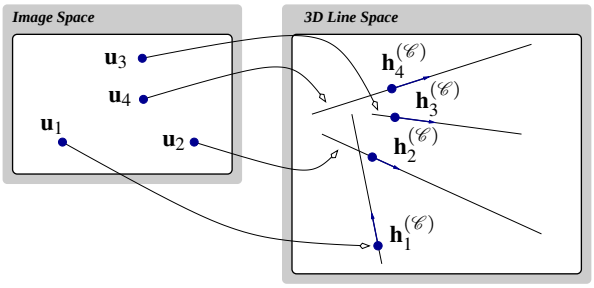


Fig. 1. Depict of the mapping between image pixels and 3D straight lines that constitutes the general camera model. We call these 3D lines as "projection lines".

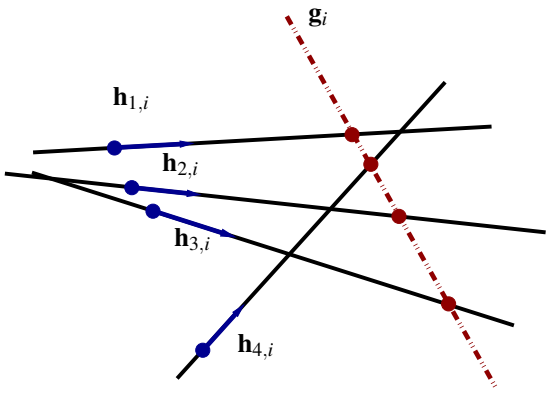


Fig. 2. In this figure we show an example of the intersection between a world line $\mathbf{g}_i$ and four "projection lines" $\mathbf{h}_{j,i}$, for $j = 1,\ldots,4$.

However this approach requires the estimation of more parameters than those required by the problem (the absolute pose requires the estimation of six parameters – three for translation and three for rotation). Considering that the coordinates of N 3D straight lines are known, as well as their corresponding image pixels, and also that the lines are represented by four degrees of freedom (minimal representation for the line), it is necessary to estimate $N4+6$ parameters. It is also known that the methods for the estimation of the coordinates of 3D lines are highly sensitive to noise, when the camera models are close to central. The issues related to the estimation of 3D lines were analyzed in several papers. For non-central catadioptric cameras 3D line estimation was addressed in [18], [19], [21], [22]. In this paper, however, we are not assuming any underlying geometry for camera model, i.e., we are not addressing the case of non-central catadioptric cameras. However and to evaluate the advantages of the proposed method, without explicitly estimating the known 3D straight lines in the camera coordinate system, in Sec. III, we briefly describe a two-step algorithm that we will use in the experiments Sec. IV.

A. Notations and Background

1) Notations: By default, we use small bold letters to represent n dimensional vectors (eg. $\mathbf{a} \in \mathbb{R}^n$). To represent matrices, we use capital bold letters (eg. $\mathbf{A} \in \mathbb{R}^{n\times m}$). Regular small letters represent one dimensional elements. We use $\sim$ to express an algebraic relation up to a scale factor.

We use superscripts $(\mathscr{W})$ and $(\mathscr{C})$ to distinguish between the same feature in the world and camera coordinate systems, respectively.

The operator $(\cdot)$ represents the inner product. We use the operator $s(.)$ to represent the 3×3 matrix that linearizes the exterior product, such that $\mathbf{a}\times\mathbf{b} = s(\mathbf{a})\mathbf{b}$ and

$$s(\mathbf{a}) \in \mathbb{R}^{3\times 3} \doteq \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{1}$$

2) Line Representation: 3D straight lines have four degrees of freedom. There are many ways to represent 3D lines. We use the six-tuple *Plücker* coordinates, $\mathbf{g} \sim (\overline{\mathbf{g}}, \breve{\mathbf{g}})$ where $\overline{\mathbf{g}} \in \mathbb{R}^3$ and $\breve{\mathbf{g}} \in \mathbb{R}^3$ are respectively the direction and moment of the line $\mathbf{g} \subset \mathbb{R}^6$. *Plücker* coordinates are a subset of a six-dimensional space because they must verify the constraint known as the *Klein* quadric: $\overline{\mathbf{g}}\cdot\breve{\mathbf{g}} = 0$. One of the main advantages of the use of *Plücker* coordinates is the possibility of dealing with incidence relations [23]. Let us consider two lines $\mathbf{g} \sim (\overline{\mathbf{g}}, \breve{\mathbf{g}})$ and $\mathbf{h} \sim \left(\overline{\mathbf{h}}, \breve{\mathbf{h}}\right)$. They intersect if and only if

$$\Omega(\mathbf{g},\mathbf{h}) = 0 \Leftrightarrow \overline{\mathbf{g}}\cdot\breve{\mathbf{h}} + \breve{\mathbf{g}}\cdot\overline{\mathbf{h}} = 0. \tag{2}$$

3) Rigid Transformation Applied to Lines: Let us consider the rigid transformation between the world and camera coordinate systems defined by the rotation $\mathbf{R} \in \mathscr{SO}(3)$ and translation $\mathbf{t} \in \mathbb{R}^3$. We use lines represented in *Plücker* coordinates as $\mathbf{g}^{(\mathscr{W})} \sim \left(\overline{\mathbf{g}}^{(\mathscr{W})}, \breve{\mathbf{g}}^{(\mathscr{W})}\right)$ and $\mathbf{g}^{(\mathscr{C})} \sim \left(\overline{\mathbf{g}}^{(\mathscr{C})}, \breve{\mathbf{g}}^{(\mathscr{C})}\right)$ (for the same line represented in the world and camera coordinate systems respectively). According to [24], [25], it is possible to derive the linear operator

$$\mathbf{g}^{(\mathscr{C})} \doteq \Psi\left(\mathbf{g}^{(\mathscr{W})}\right), \tag{3}$$

that can be defined as

$$\begin{bmatrix} \overline{\mathbf{g}}^{(\mathscr{C})} \\ \breve{\mathbf{g}}^{(\mathscr{C})} \end{bmatrix} \sim \underbrace{\begin{bmatrix} \mathbf{R} & \mathbf{0}_{3\times 3} \\ \mathbf{E} & \mathbf{R} \end{bmatrix}}_{\mathbf{H}\in\mathbb{R}^{6\times 6}} \begin{bmatrix} \overline{\mathbf{g}}^{(\mathscr{W})} \\ \breve{\mathbf{g}}^{(\mathscr{W})} \end{bmatrix}, \text{ and } \mathbf{E} \doteq s(\mathbf{t})\mathbf{R}. \tag{4}$$

Note that matrix $\mathbf{E} \in \mathbb{R}^{3\times 3}$ is known as the *essential* matrix [1].

II. Pose Using Lines for General Camera Models

Pose estimation consists in finding the rotation and translation, from 3D straight lines whose coordinates are known (in the world coordinate system) and their corresponding images.

To represent lines, we use *Plücker* coordinates (Sec.I-A2). Let us consider the problem of the pose estimation as the determination of the rotation $\mathbf{R} \in \mathbb{R}^{3\times 3}$ and translation $\mathbf{t} \in \mathbb{R}^3$ parameters that define rigid transformation between the world and camera coordinate systems. Moreover, let us consider that we know M coordinates of 3D straigh lines, in the world coordinate system. Since we are considering generalized imaging systems, the image of any 3D straight line is a non-parametric curve in the image plane. Depending on the smoothness of the imaging system, the curve can be continuous or discontinuous.

TABLE I
IN THIS TABLE WE SHOW THE ELEMENTS OF THE VOCTOR $\mathbf{c}_{i,j} \in \mathbb{R}^{18}$. $c_{i,j}^{(k)}$ DENOTES THE k^{th} ELEMENT OF $\mathbf{c}_{i,j}$ AND ${}^{k}\overline{g}_i^{(\mathscr{W})}$ DENOTES THE k^{th} ELEMENT OF THE VECTOR $\overline{\mathbf{g}}_i^{(\mathscr{W})}$. THE SAME DENOMINATION IS USED FOR $\breve{\mathbf{g}}_i^{(\mathscr{W})}$, $\overline{\mathbf{h}}_i^{(\mathscr{W})}$ AND $\breve{\mathbf{h}}_i^{(\mathscr{W})}$

$c_{i,j}^{(1)} = {}^1\overline{g}_i^{(\mathscr{W})}\ {}^1\breve{h}_i^{(\mathscr{C})} + {}^1\overline{h}_i^{(\mathscr{C})}\ {}^1\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(2)} = {}^2\overline{g}_i^{(\mathscr{W})}\ {}^1\breve{h}_i^{(\mathscr{C})} + {}^1\overline{h}_i^{(\mathscr{C})}\ {}^2\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(3)} = {}^3\overline{g}_i^{(\mathscr{W})}\ {}^1\breve{h}_i^{(\mathscr{C})} + {}^1\overline{h}_i^{(\mathscr{C})}\ {}^3\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(4)} = {}^1\overline{g}_i^{(\mathscr{W})}\ {}^2\breve{h}_i^{(\mathscr{C})} + {}^2\overline{h}_i^{(\mathscr{C})}\ {}^1\breve{g}_i^{(\mathscr{W})}$
$c_{i,j}^{(5)} = {}^2\overline{g}_i^{(\mathscr{W})}\ {}^2\breve{h}_i^{(\mathscr{C})} + {}^2\overline{h}_i^{(\mathscr{C})}\ {}^2\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(6)} = {}^3\overline{g}_i^{(\mathscr{W})}\ {}^2\breve{h}_i^{(\mathscr{C})} + {}^2\overline{h}_i^{(\mathscr{C})}\ {}^3\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(7)} = {}^1\overline{g}_i^{(\mathscr{W})}\ {}^3\breve{h}_i^{(\mathscr{C})} + {}^3\overline{h}_i^{(\mathscr{C})}\ {}^1\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(8)} = {}^2\overline{g}_i^{(\mathscr{W})}\ {}^3\breve{h}_i^{(\mathscr{C})} + {}^3\overline{h}_i^{(\mathscr{C})}\ {}^2\breve{g}_i^{(\mathscr{W})}$
$c_{i,j}^{(9)} = {}^3\overline{g}_i^{(\mathscr{W})}\ {}^3\breve{h}_i^{(\mathscr{C})} + {}^3\overline{h}_i^{(\mathscr{C})}\ {}^3\breve{g}_i^{(\mathscr{W})}$	$c_{i,j}^{(10)} = {}^1\overline{g}_i^{(\mathscr{W})}\ {}^1\overline{h}_i^{(\mathscr{C})}$	$c_{i,j}^{(11)} = {}^2\overline{g}_i^{(\mathscr{W})}\ {}^1\overline{h}_i^{(\mathscr{C})}$	$c_{i,j}^{(12)} = {}^3\overline{g}_i^{(\mathscr{W})}\ {}^1\overline{h}_i^{(\mathscr{C})}$
$c_{i,j}^{(13)} = {}^1\overline{g}_i^{(\mathscr{W})}\ {}^2\overline{h}_i^{(\mathscr{C})}$	$c_{i,j}^{(14)} = {}^2\overline{g}_i^{(\mathscr{W})}\ {}^2\overline{h}_i^{(\mathscr{C})}$	$c_{i,j}^{(15)} = {}^3\overline{g}_i^{(\mathscr{W})}\ {}^2\overline{h}_i^{(\mathscr{C})}$	$c_{i,j}^{(16)} = {}^1\overline{g}_i^{(\mathscr{W})}\ {}^3\overline{h}_i^{(\mathscr{C})}$
$c_{i,j}^{(17)} = {}^2\overline{g}_i^{(\mathscr{W})}\ {}^3\overline{h}_i^{(\mathscr{C})}$	$c_{i,j}^{(18)} = {}^3\overline{g}_i^{(\mathscr{W})}\ {}^3\overline{h}_i^{(\mathscr{C})}$		

As a result and to maintain the general concept, we consider as known data-set a set of pixels that belong to the images of 3D straight lines. Thus, we know the coordinates of 3D straigth lines in the world coordinate system $\mathbf{g}_i^{(\mathscr{W})}$ and the set of associated image pixels $\mathbf{u}_{i,j}$, for all j, forming the data-set $\mathbf{g}_i^{(\mathscr{W})} \mapsto \{\mathbf{u}_{i,j}\}$ for all i.

Since we consider that the camera is calibrated according to the general camera model Fig. 1, for each and all image pixels, we know the corresponding "projection line", in the camera coordinate system: $\mathbf{u}_{i,j} \mapsto \mathbf{h}_{i,j}^{(\mathscr{C})}$, forming the data-set $\mathbf{g}_i^{(\mathscr{W})} \mapsto \left\{\mathbf{h}_{i,j}^{(\mathscr{C})}\right\}$ for all i. Note that in this formulation, we never use coordinates of 3D points. A scheme of this representation for some i^{th} known 3D line $\mathbf{g}_i^{(\mathscr{W})}$ is shown in Fig. 2.

From Sec. I-A2, for each and all of the M 3D straight lines (i^{th} line), and the associated j^{th} "projection line", the following constraint must be verified

$$\Omega\left(\mathbf{g}_i^{(\mathscr{C})}, \mathbf{h}_{j,i}^{(\mathscr{C})}\right) = 0, \ \forall j. \tag{5}$$

Note that for this constraint, the lines must be represented in the same coordinate system. We consider the camera coordinate system which means that the coordinates of $\mathbf{g}_i^{(\mathscr{C})}$ are unknowns. However, we know the coordinates of the 3D straight lines in the world coordinate system $\mathbf{g}_i^{(\mathscr{W})}$. Applying the rigid transformation as suggested in Sec.I-A3 (this transformation defines the pose), the constraint defined in (5) can be rewritten as

$$\Omega\left(\Psi\left(\mathbf{g}_i^{(\mathscr{W})}\right), \mathbf{h}_{j,i}^{(\mathscr{C})}\right) = 0, \ \forall j. \tag{6}$$

Developing (6) using (2) and (4), for each and all of the i^{th} 3D straight line $\mathbf{g}_i^{(\mathscr{W})}$, we get

$$\left(\mathbf{R}\overline{\mathbf{g}}_i^{(\mathscr{W})}\right) \cdot \breve{\mathbf{h}}_{j,i}^{(\mathscr{C})} + \left(\mathbf{E}\overline{\mathbf{g}}_i^{(\mathscr{W})}\right) \cdot \overline{\mathbf{h}}_{j,i}^{(\mathscr{C})} + \left(\mathbf{R}\breve{\mathbf{g}}_i^{(\mathscr{W})}\right) \cdot \overline{\mathbf{h}}_{j,i}^{(\mathscr{C})} = 0, \ \forall j. \tag{7}$$

Let us consider

$$\mathbf{R} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \text{ and } \mathbf{E} = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix}, \tag{8}$$

developing (7) we get

$$\mathbf{c}_{i,j} \cdot \mathbf{v} = 0, \tag{9}$$

such that $\mathbf{v} \in \mathbb{R}^{18}$ is

$$\mathbf{v} = (r_1, r_2, \ldots, r_9, e_1, e_2, \ldots, e_9) \tag{10}$$

and the elements of vector $\mathbf{c}_{i,j} \in \mathbb{R}^{18}$ are shown in Tab. I.

From (9) and (10), we see that we have eighteen unknowns. However, these unknowns are not linearly independent. From the properties of the rotation matrix $\mathbf{R} \in \mathscr{SO}(3)$, we have

$$\mathbf{R}^T\mathbf{R} = \mathbf{I} \Rightarrow \mathbf{R}^T\mathbf{R} - \mathbf{I} = \mathbf{0}, \tag{11}$$

which corresponds to nine constraints. We note that these constraints are not all linearly independent too, [11]. In the general case there are only six which are linearly independent.

Moreover, additional constraints must be taken into account from the *essential* matrix $\mathbf{E}$. We know that $\mathbf{E} \doteq s(\mathbf{t})\mathbf{R}$, which means that

$$\mathbf{E}\mathbf{R}^T = s(\mathbf{t}) \underbrace{\mathbf{R}\mathbf{R}^T}_{\mathbf{I}} = s(\mathbf{t}). \tag{12}$$

Matrix $s(\mathbf{t})$ must be as (1). Thus, we can define six constraints to the product $\mathbf{E}\mathbf{R}^T$.

Adding the six constraints derived from (12) to the previous six linearly independent constraints derived from (11), we get twelve constraints. Thus, the problem defined by (9) subject to the constraints derived in (11) and (12) has six degrees of freedom (that correspond to the six degrees of freedom of the pose problem).

The constraints defined by (11) and (12) are quadratic constraints and can be represented as

$$\mathbf{v}^T\mathbf{K}_l\mathbf{v} = k_l, \ \text{for } l = 1, \ldots, 15, \tag{13}$$

where k_l is zero or one, depending on the constraint equation.

To conclude, stacking all the $\mathbf{c}_{i,j}$ (for all 3D straight lines and their associated "projection lines") into matrix $\mathbf{C}$, the general pose using lines can be obtained by solving the problem

$$\begin{aligned} \min_{\mathbf{v}} \quad & ||\mathbf{C}\mathbf{v}||^2 \\ \text{s.t} \quad & \mathbf{v}^T\mathbf{K}_l\mathbf{v} = k_l, \ l = 1, \ldots, 15. \end{aligned} \tag{14}$$

Using *Lagrange multipliers* λ_l, we formulate the *Lagrangian* $\mathscr{L}(\mathbf{v}, \lambda)$ of this problem as

$$\mathscr{L}(\mathbf{v}, \lambda) = \mathbf{v}^T\mathbf{C}^T\mathbf{C}\mathbf{v} + \sum_{l=1}^{15} \lambda_l \left(\mathbf{v}^T\mathbf{K}_l\mathbf{v} - k_l\right) \tag{15}$$

and the problem defined in (14) can be rewritten as

$$\max_{\lambda} \min_{\mathbf{v}} \mathscr{L}(\mathbf{v}, \lambda) \tag{16}$$

This problem is well studied in the literature. In the experiments, we used the `Matlab` optimization toolbox.

III. The Two-Step Solution

In this paper we address the pose estimation using known coordinates of 3D straight lines in the world coordinate system. The proposed solution was described in Sec. II. As far as we know, this problem was not addressed before. However and as described in Sec. I, there is a trivial two step solution that can be easily derived from the two following assumptions:

- For general non-central cameras, we can determine the 3D coordinates of a straight line from the coordinates of four or more incident skewed lines [26], [18], [19], [21], [20], [22].
- Using the estimated 3D lines in the camera coordinate system and since we know the coordinates of the same lines in the world reference frame, the rotation and translation that define the transformation of the lines from the world to the camera coordinate system can be estimated.

From the constraint that ensures that two lines intersect (2), and for five or more "projection lines" ($\mathbf{h}_{j,i}^{(\mathscr{C})}$ for $j=1,\ldots,N_i$, where $N_i \geqslant 5$) that are incident with the 3D straight line $\mathbf{g}_i^{(\mathscr{C})}$ in the camera coordinate system (see Fig. 2), we define the following algebraic relation

$$\underbrace{\begin{bmatrix} \breve{\mathbf{h}}_{1,i}^{(\mathscr{C})\,T} & \mathbf{h}_{1,i}^{(\mathscr{C})\,T} \\ \vdots & \vdots \\ \breve{\mathbf{h}}_{N_i,i}^{(\mathscr{C})\,T} & \mathbf{h}_{N_i,i}^{(\mathscr{C})\,T} \end{bmatrix}}_{\mathbf{A}_i} \begin{bmatrix} \mathbf{g}_i^{(\mathscr{C})} \\ \breve{\mathbf{g}}_i^{(\mathscr{C})} \end{bmatrix} = \mathbf{0}_{N_i}, \tag{17}$$

where $\mathbf{A}_i \in \mathbb{R}^{N_i \times 6}$. Thus, we can conclude that

$$\left(\mathbf{g}_i^{(\mathscr{C})},\, \breve{\mathbf{g}}_i^{(\mathscr{C})}\right) \sim \operatorname{null}(\mathbf{A}_i). \tag{18}$$

For data with noise, matrices $\mathbf{A}_i$ will generally have $\operatorname{rank}(\mathbf{A}_i) = 6$ which means that there will be no direct solution for (18). However, a solution can be computed in the least-squares sense. Note that line $\mathbf{g}_i^{(\mathscr{C})}$ can be represented up to a scale factor. As a result, it is easy to see that the solution for the coordinates of this line can be computed non-iteratively by computing an inverse of a 5×5 matrix.

We also note that for noisy data, the solution for $\mathbf{g}_i^{(\mathscr{C})}$ may not verify the *Klein* quadric constraint: $\mathbf{g}_i^{(\mathscr{C})} \cdot \breve{\mathbf{g}}_i^{(\mathscr{C})} = 0$. However, we note that there exist *Plücker* correction algorithms to recover the coordinates meeting the Klein quadric constraint [27].

Let us now assume that we have M 3D straight lines in the world coordinate system. For each one of them, we have five or more "projection lines" ($N_i \geqslant 5$), for $i=1,\ldots,M$. We can compute line coordinates $\mathbf{g}_i^{(\mathscr{C})}$ using (18), for all i.

Note that for all the estimated coordinates of the 3D straight lines $\mathbf{g}_i^{(\mathscr{C})}$, we know the correspondent coordinates in the world coordinate system $\mathbf{g}_i^{(\mathscr{W})}$. As a result, the rotation and translation parameters can be estimated such that all the set $\left\{\mathbf{g}_i^{(\mathscr{C})} \leftrightarrow \mathbf{g}_i^{(\mathscr{W})}\right\}$ verifies the transformation defined in (3).

Let us consider the matching between directions $\left\{\overline{\mathbf{g}}_i^{(\mathscr{C})} \leftrightarrow \overline{\mathbf{g}}_i^{(\mathscr{W})}\right\}$. From (4), we can see that the rotation matrix must verify

$$\underbrace{\begin{bmatrix} \overline{\mathbf{g}}_1^{(\mathscr{C})} & \cdots & \overline{\mathbf{g}}_M^{(\mathscr{C})} \end{bmatrix}}_{\mathbf{P}_1 \in \mathbb{R}^{3\times M}} \sim \mathbf{R} \underbrace{\begin{bmatrix} \overline{\mathbf{g}}_1^{(\mathscr{W})} & \cdots & \overline{\mathbf{g}}_M^{(\mathscr{W})} \end{bmatrix}}_{\mathbf{P}_2 \in \mathbb{R}^{3\times M}} \tag{19}$$

The rotation matrix $\mathbf{R}$ can be estimated solving the *procrustes* problem [28], which can be solved by computing a *Singular Value Decomposition* of a 3×3 matrix. We note that, since we are dealing with 3×3 matrices, there exists analytical solution for the SVD. Moreover, we also ensure that the solution for the rotation matrix $\mathbf{R}$ belongs to the space of orthonormal matrices.

From the second row of (4) and since $\breve{\mathbf{g}}_i^{(\mathscr{C})} \times \breve{\mathbf{g}}_i^{(\mathscr{C})} = \mathbf{0}_3$, we can define the following algebraic relation:

$$\breve{\mathbf{g}}_i^{(\mathscr{C})} \times \breve{\mathbf{g}}_i^{(\mathscr{C})} = \left(-s(\mathbf{t})\,\mathbf{R}\overline{\mathbf{g}}_i^{(\mathscr{W})} + \mathbf{R}\breve{\mathbf{g}}_i^{(\mathscr{W})}\right) \times \breve{\mathbf{g}}_i^{(\mathscr{C})} = \mathbf{0}_3. \tag{20}$$

Developing this equation and taking into account that $\mathbf{a}\times\mathbf{b} = -\mathbf{b}\times\mathbf{a}$ which implies $s(\mathbf{a})\,\mathbf{b} = -s(\mathbf{b})\,\mathbf{a}$, we get

$$\begin{aligned} &-s\left(\breve{\mathbf{g}}_i^{(\mathscr{C})}\right)\left(s\left(\mathbf{R}\overline{\mathbf{g}}_i^{(\mathscr{W})}\right)\mathbf{t} + \mathbf{R}\breve{\mathbf{g}}_i^{(\mathscr{W})}\right) = 0 \\ &\quad\Rightarrow \underbrace{-s\left(\breve{\mathbf{g}}_i^{(\mathscr{C})}\right) s\left(\mathbf{R}\overline{\mathbf{g}}_i^{(\mathscr{W})}\right)}_{\mathbf{B}_i}\mathbf{t} = \underbrace{s\left(\breve{\mathbf{g}}_i^{(\mathscr{C})}\right)\mathbf{R}\breve{\mathbf{g}}_i^{(\mathscr{W})}}_{\mathbf{b}_i}, \end{aligned} \tag{21}$$

where $\mathbf{B}_i \in \mathbb{R}^{3\times 3}$ and $\mathbf{b}_i \in \mathbb{R}^3$.

For $i=1,\ldots,M$, we have

$$\underbrace{\begin{bmatrix} \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_M \end{bmatrix}}_{\mathbf{B}} \mathbf{t} = \underbrace{\begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_M \end{bmatrix}}_{\mathbf{b}} \tag{22}$$

where $\mathbf{B} \in \mathbb{R}^{3M\times 3}$ and $\mathbf{b} \in \mathbb{R}^{3M}$. The solution for $\mathbf{t}$ can be computed using $\mathbf{t} = \mathbf{B}^{\dagger}\mathbf{b}$, where $\mathbf{B}^{\dagger}$ represents the *pseudo-inverse* of $\mathbf{B}$. Note that also $\mathbf{B}^{\dagger}$ can be computed in closed-form, which means that $\mathbf{t}$ can be solved analytically too.

To have a single solution for the translation parameters $\mathbf{t}$, matrix $\mathbf{B}$, in (22), must have rank equal to three. Otherwise, the solution will be degenerate. It can be proved that, independently from $\mathbf{R}$, $\breve{\mathbf{g}}_i^{(\mathscr{C})}$ or $\overline{\mathbf{g}}_i^{(\mathscr{W})}$, the rank of $\mathbf{B}_i$, in (21), will be equal to one. As a result, a matching of, at least, three 3D lines is required both in the world and camera coordinate systems ($M \geqslant 3$).

IV. Experiments with Synthetic Data

To evaluate and validate the method proposed in Section II against the trivial two step solution proposed in Section III, experiments with both synthetic and real data are performed. The `Matlab` code, used to compute the experimental results, will be available on the author's page.

For synthetic data, we consider the following procedure. We randomly generate M 3D straight lines $\mathbf{g}_i^{(\mathscr{C})}$. To get these lines we randomly generate 3D points $\widetilde{\mathbf{g}}_i^{(\mathscr{C})} \in \mathbb{R}^3$ (in a cube with 200 units of side length) and random directions $\overline{\mathbf{g}}_i^{(\mathscr{C})}$ (with norm equal to one). In this representation, any point incident with the line can be expressed as $\widetilde{\mathbf{g}}_i^{(\mathscr{C})} + \mu_i \overline{\mathbf{g}}_i^{(\mathscr{C})}$ for some $\mu_i \in \mathbb{R}$.

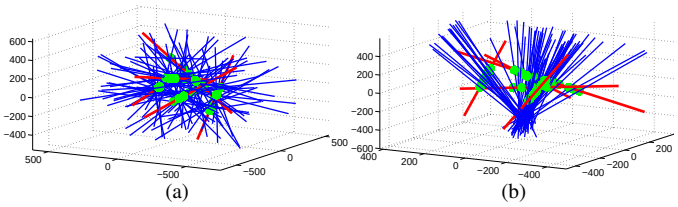


Fig. 3. In this figure we show two examples of the generation of the 3D lines and "projection lines" in the camera coordinate system $\mathbf{h}_{j,i}^{(\mathscr{C})}$ (blue lines) and $\mathbf{g}_i^{(\mathscr{C})}$ (red lines). As green, we show the intersection points. In Fig. (a), we show an example of the general case. In the case of Fig. (b) the directions of the "projection lines" ensure that they pass close to each other (close to central).

Using the 3D point and respective direction coordinates, we get the *Plücker* coordinates such that

$$\mathbf{g}_i^{(\mathscr{C})} \doteq \left(\overline{\mathbf{g}}_i^{(\mathscr{C})}, \widetilde{\mathbf{g}}_i^{(\mathscr{C})} \times \overline{\mathbf{g}}_i^{(\mathscr{C})}\right). \tag{23}$$

For more information see [23].

For each 3D line, we compute N_i "projection lines" $\mathbf{h}_{j,i}^{(\mathscr{C})}$. We randomly choose N_i parameters $\widetilde{\mu}_j$ (for $j = 1,\ldots,N_i$) and compute the coordinates of the 3D points $\widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})} = \widetilde{\mathbf{g}}_i^{(\mathscr{C})} + \widetilde{\mu}_j \overline{\mathbf{g}}_i^{(\mathscr{C})}$. Note that $\widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})}$ belong to the line $\mathbf{g}_i^{(\mathscr{C})}$. The set of parameters $\widetilde{\mu}_j$ is randomly chosen, from -100 to 100. The set of directions $\overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}$ is randomly computed too. The *Plücker* coordinates of the "projection lines" are then computed as

$$\mathbf{h}_{j,i}^{(\mathscr{C})} \doteq \left(\overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}, \widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})} \times \overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}\right). \tag{24}$$

An example of the generation of both $\mathbf{g}_i^{(\mathscr{C})}$ and $\mathbf{h}_{j,i}^{(\mathscr{C})}$ is shown in Fig. 3(a).

Random ground truth rotations and translation parameters are computed ($\mathbf{R}_{\text{gt}} \in \mathscr{SO}(3)$ and $\mathbf{t}_{\text{gt}} \in \mathbb{R}^3$). We generate $\mathbf{t}_{\text{gt}}$ in a cube with 200 units of side length. Using these parameters, we get the *Plücker* coordinates of the 3D lines in the world coordinate system as $\mathbf{g}_i^{(\mathscr{W})} = \Psi^{-1}\left(\mathbf{g}_i^{(\mathscr{C})}\right)$, (4).

To conclude, pose is computed using the association between $\mathbf{g}_i^{(\mathscr{W})} \leftrightarrow \left\{\mathbf{h}_{1,i}^{(\mathscr{C})}, \ldots, \mathbf{h}_{N_i,i}^{(\mathscr{C})}\right\}$, for all i. For each estimate $\{\mathbf{R},\mathbf{t}\}$ and ground truth data $\{\mathbf{R}_{gt},\mathbf{t}_{gt}\}$, we compute the six parameters that define the pose: three angles for the rotation (radians) and three coordinates for the translation. For all the pose parameters, we compute the distance between the ground truth and the estimated parameters.

A. Evaluation of the Noise and Deviation from Central Case

In many non-central cameras (such as non-central catadioptric cameras with quadric mirrors), and despite the fact that they are non-central, the "projection lines" pass close to each other. Therefore it is important that the proposed method be evaluated for those configurations. Instead of considering random directions for $\overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}$, we constrain those directions in order to ensure that the "projection lines" pass close to each other. The following procedure is applied: since we already have the 3D coordinates of a point $\widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})}$ that is incident on the line $\mathbf{h}_{j,i}^{(\mathscr{C})}$, an additional point is enough to compute the coordinates of the direction; since in addition we want that all lines pass close to each other, for each "projection line" we compute a new point $\overline{\mathbf{h}}_{j,i}^{(\mathscr{C})} \in \mathbb{R}^3$ randomly chosen in a cube whose side length is defined by the variable `Deviation from Central Case`. The directions are thus computed as

$$\overline{\mathbf{h}}_{j,i}^{(\mathscr{C})} = \left(\widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})} - \overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}\right) \Big/ \left|\widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})} - \overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}\right|. \tag{25}$$

Note that when the variable `Deviation from Central Case` tends to zero, the camera model tends to be central. When it gets higher, it tends to the general case. An example with `Deviation from Central Case` equal to twenty is shown in Fig. 3(b).

For the first evaluation, we consider data with noise. Instead of considering "projection lines" as described in (24), we use

$$\mathbf{h}_{j,i}^{(\mathscr{C})} \doteq \left(\overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}, \left(\widetilde{\mathbf{h}}_{j,i}^{(\mathscr{C})} + e_{j,i}^{(\mathscr{C})}\right) \times \overline{\mathbf{h}}_{j,i}^{(\mathscr{C})}\right). \tag{26}$$

The vector $e_{j,i}^{(\mathscr{C})}$ has random direction and random norm, with standard deviation equal to the `Noise` variable. We vary the `Noise` variable from one to ten and the results are shown in Figs. 4(a) and 4(c) for the general case and for the case where `Deviation from Central Case` is equal to twenty, respectively.

In Fig. 4(b) we evaluate the proposed method by varying the number of 3D lines used to compute the pose. We consider the general case and `Noise` variable equal to 20.

In addition, we evaluate the proposed approach as a function of the `Deviation from Central Case` variable – we vary the evaluation variable from 80 to 10. For that purpose, we consider the `Noise` variable as 10. The results are shown in Fig. 4(d).

B. Convergence of the Method

For the method of Sec. II, it is required to analyse the presence of local minima in the non-linear optimization approach (defined by (16). For that purpose, we test the method by using inital values that are randomly generated. We consider noiseless synthetic data, such that the data-set is generated as described in the previous section. For initial values, we

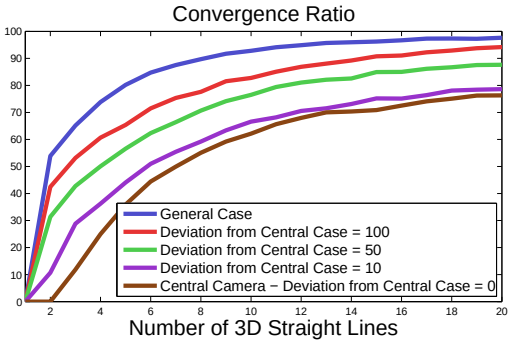
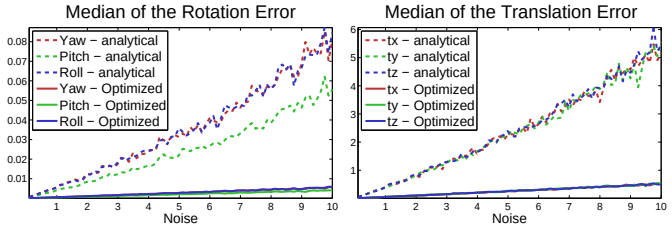
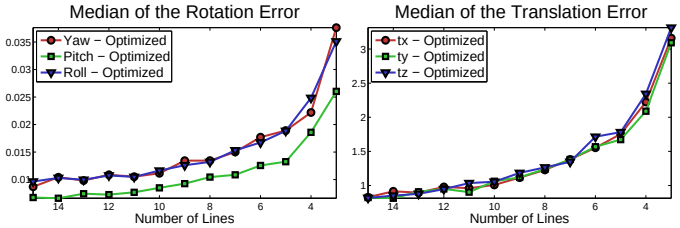


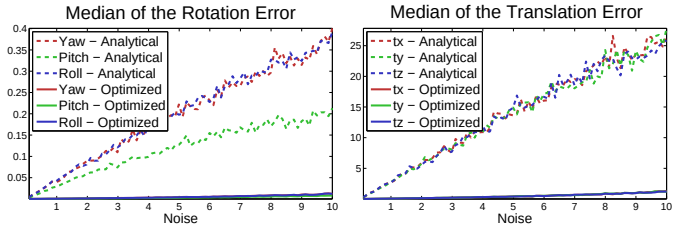
Fig. 5. In this figure we evaluate the convergence ratio of the non-linear method proposed in Sec.II. For this purpose, we vary the number of known 3D straight lines. In addition, we also consider different values for `Deviation from Central Case` variable. Note that we consider the case of central camera – `Deviation from Central Case` equal to zero.



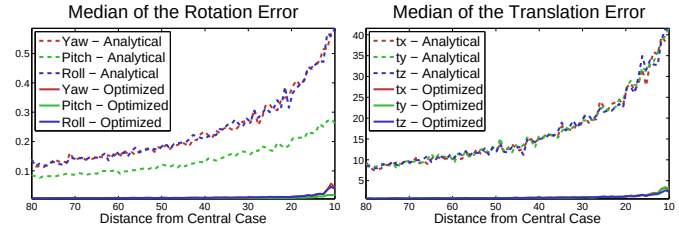
(a) Evaluation of the proposed approach as a function of the `Noise` variable. We consider the general case – Fig. 3(a). For that purpose, we consider $N_i = 40$, $\forall i$ and $M = 10$.



(b) Evaluation of the proposed approach as a function of the number of 3D lines used. We use $N_i = 40$, $\forall i$ and `Noise` variable with value equals to 20. Note that three 3D lines is the minimal case.



(c) Similar to the case of Fig. (a) but, instead of the general case, we consider `Deviation from Central Case` variable equals to 20 – Fig. 3(b).



(d) Evaluation of the proposed approach as a function of the `Deviation from Central Case`. We consider the `Noise` variable as 10 and use $N_i = 40$, $\forall i$ and $M = 10$.

Fig. 4. In this figure we show the evaluation of the proposed method using synthetic data. We consider the error for the six parameters that define the pose: three rotation angles – radians; and three for the translation. In Figs. (a) and (c) we show the evaluation of the algorithm as a function of the noise. In Fig. (b) we show the results for different number of lines. In Fig. (d) we study the variation of the error as a function of the `Deviation from Central Case`. For each value of the evaluation variable, we consider 10^3 trials as described in the text.

consider $\mathbf{t}_0 = \mathbf{0}_3$ and rotation matrix $\mathbf{R}_0$ computed using random rotation angles.

We tested five different values for the variable `Distance from Central Case`. The first case is thus the general case. We also consider values of 100, 50, 10 and 0. Note that when this variable is 0 we have a central camera model. In addition, we consider $N_i = 40$, for all i.

For each one of these different configurations, we evaluate pose for different number of known 3D straight lines. The convergence ratio was computed considering 10^4 trials for each number of lines. We consider that a solution converges if the norm of the errors for the translation and rotation parameters is smaller then 10^{-5}.

V. Experiments with Real Data

For the experiments with real data, we consider a non-central calibrated catadioptric camera, made up by a perspective camera and a spherical mirror. Since the trivial two step method, proposed in Sec. III, gave very bad results (specially due to bad estimations of the coordinates of the 3D lines in the camera coordinate system), we will ignore its results in these experiments. Note that the analysis of the errors, as a function of the noise, is evaluated in Sec. IV.

We consider a sequence of images taken by moving the camera through a path in the lab. In each image, we determine eight curves that correspond to known straight lines in the world coordinate system. Five examples of these images are shown in Fig. 6(a). The coordinates of the eight known 3D straight lines in the world coordinate system are shown by the correspondent colors in the 3D Fig. 7. Using the described data set and the method proposed in this article, we estimate the pose for the sequence. The results are shown in Fig. 7. In this figure, we show the position and orientation of the perspective camera of the non-central catadioptric system.

The estimation of the pose allows the determination of the localization of the camera in the world coordinate system. Therefore and since we are using a non-central catadioptric camera with a spherical mirror, points in the world can be mapped into the image [30], [29]. To further evaluate the proposed solution for the pose, we consider an application of augmented reality, with the following objects:

- Extra 3D straight lines on the lab's floor – represented as red lines in Fig. 7;
- Two 3D rectangular parallelepipeds on the chess-boards pattern, located on the floor of the lab. These two parallelepipeds are represented in green and blue – see Fig. 7.

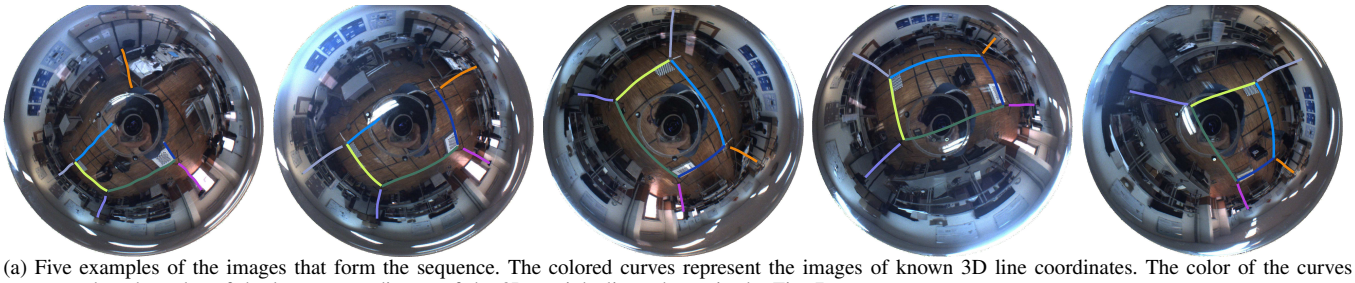
For each object, we project the points that make up the objects into the images that form the sequence. Two examples of the application of the augmented reality are shown in Fig. 6(b).

A movie with the results for the sequence is provided in the supplementary material.

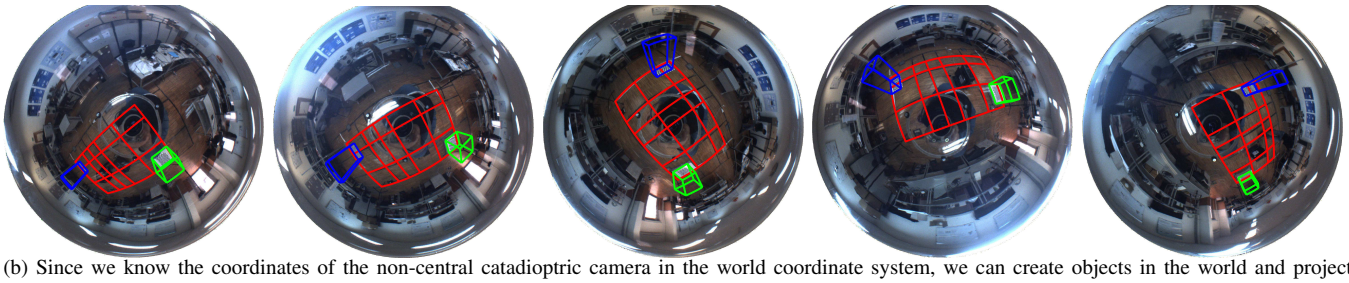
VI. Conclusions

A. Discussion of the Experimental Results

Let us first consider the trivial two steps solution, derived in Section III. When considering general non-central cameras and from Fig. 4(a), we see that this method has an acceptable performance even in the case of noisy data. However, it is well known that, for the case of central cameras, the solution degenerate (it is not possible to estimate the coordinates of the 3D lines in the camera coordinate system). As expected, the results deteriorate significantly for configurations close to the central camera models – see Figs. 4(c)-(d). Taking into account that many of the non-central imaging devices are



(a) Five examples of the images that form the sequence. The colored curves represent the images of known 3D line coordinates. The color of the curves correspond to the color of the known coordinates of the 3D straight lines shown in the Fig. 7.



(b) Since we know the coordinates of the non-central catadioptric camera in the world coordinate system, we can create objects in the world and project them to the image [30], [29]. We test the proposed pose estimation using an application of augmented reality. The 3D generated objects are shown in Fig. 7. In this figure, we show an example of the proposed augmented reality for the same two images of (a).

Fig. 6. In Fig. (a) we show five examples of images taken from the non-central catadioptric camera. The image curves marked in the image correspond to the data set used for the computation of the pose. In Fig. (b) we test the proposed method using an application of augmented reality. The created objects are shown in Fig. 7.

relatively close to central camera models (for example the non-central catadioptric camera), this deterioration on the results may significantly affect the estimation of the pose.

From the results presented in Figs. 4 and contrarily to the trivial two step solution, one can conclude that our approach, Sec. II – without explicitly estimating the known 3D straight lines in the camera coordinate system, significantly decreases the errors due to both noise and to configurations close to the central case. To evaluate the convergence of the non-linear optimization, represented in (16), random values for the rotation and zero values for the translation were used as initial estimates. Convergence ratios of up to 95% for the case of general non-central cameras were obtained–see Fig. 5.

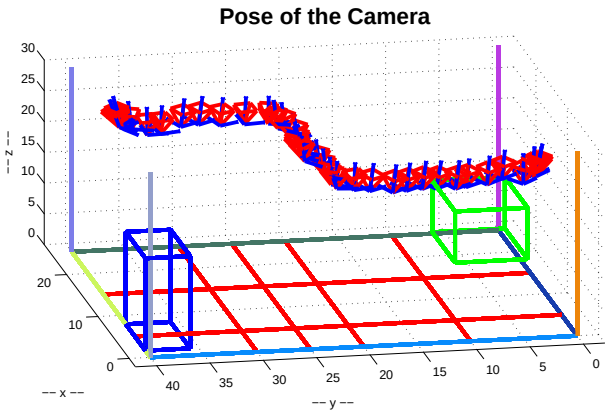


Fig. 7. In this figure we show the reconstruction of the motion obtained from a sequence of images taken from the non-central catadioptric camera – formed with a perspective camera and a spherical mirror. In the graphic, we show the recovered position of the perspective camera.

When the geometry of the imaging system approaches central projection, the convergence ratio decreases. However, for the central case, the convergence ratios of up to 75% are obtained. Nevertheless and also contrarily to the trivial two step solution, it was proven that our method works also for the cases where the imaging device can be modeled by the perspective cameras (central camera models).

In the experiments with real data, Sec.V, we reconstructed the 3D motion from a sequence of images taken from a non-central imaging system. In addition, we also considered a simple example of an augmented reality application that validates the solution for the pose. Note that we are using a non-central catadioptric camera that uses a spherical mirror. This imaging device is relatively close to a central camera model which means that it will be very difficult to compute the coordinates of the 3D lines in the camera coordinate system. As result and since the estimation of the pose using the two step approach gave poor results in the experiments using real data, we only display results using our approach, Sec. II. From the results of figs. 6 and 7, it can be seen that the proposed method yields results with good quality.

B. Closure

In this article, we present a novel approach for pose estimation for general camera models. Instead of the correspondence between known image and 3D points, this method requires the correspondence between known 3D straight lines and pixels that correspond to their images. As far as we known, this problem was not addressed before. Our approach significantly reduces the difficulties in the acquisition of the data set in particular because it does not require the determination of correspondences between 3D points and their images. In

addition the method proved to be substantially robust against noise for a wide variety of configurations, in the difficult setting of highly non-linear and non-central imaging systems.

REFERENCES

[1] R. Hartley and A. Zisserman, Multiple View Geometry. Cambridge University Press, 2000.

[2] M. D. Grossberg and S. K. Nayar, "A General Imaging Model and a Method for Finding its Parameters," IEEE Proc. Int'l Conf. Computer Vision (ICCV), 2001.

[3] ——, "The Raxel Imaging Model and Ray–Based Calibration," Int'l J. of Computer Vision, 2005.

[4] J. Gluckman and S. K. Nayar, "Planar Catadioptric Stereo: Geometry and Calibration," IEEE Proc. Computer Vision and Pattern Recognition (CVPR), 1999.

[5] T. Treibitz, Y. Y. Schechner, C. Kunz, and H. Singh, "Flat Refractive Geometry," IEEE Trans. Pattern Annalysis and Machine Intelligence, 2012.

[6] R. Swaminathan, M. D. Grossberg, and S. K. Nayar, "Non-Single Viewpoint Catadioptric Cameras: Geometry and Analysis," Int'l J. of Computer Vision, 2006.

[7] H. Araujo, R. L. Carceroni, and C. M. Brown, "A Fully Projective Formulation to Improve the Accuracy of Lowe's Pose–Estimation Algorithm," Computer Vision and Image Understanding, 1998.

[8] F. Moreno Noguer, V. Lepetit, and P. Fua, "Accurate Non–Iteractive $O(n)$ Solution to the PnP Problem," IEEE Proc. Int'l Conf. Computer Vision (ICCV), 2007.

[9] J. Hesch and S. Roumeliotis, "A Direct Least-Squares (DLS) Method for PnP," IEEE Proc. Int'l Conf. Computer Vision (ICCV), 2011.

[10] C.-P. Lu, G. D. Hager, and E. Mjolsness, "Fast and Globally Convergent Pose Estimation from Video Images," IEEE Trans. Pattern Analysis and Machine Intelligence, 2000.

[11] A. Ansar and K. Daniilidis, "Linear Pose Estimation from Points or Lines," IEEE Trans. Pattern Annalysis and Machine Intelligence, 2003.

[12] P. Sturm and S. Ramalingam, "A Generic Concept for Camera Calibration," Proc. European Conf. Computer Vision (ECCV), 2004.

[13] P. Miraldo and H. Araujo, "Calibration of Smooth Camera Models," IEEE Trans. Pattern Annalysis and Machine Intelligence, 2013.

[14] C.-S. Chen and W.-Y. Chang, "Pose Estimation for Generalized Imaging Device via Solving Non-Perspective N Point Problem," IEEE Proc. Int'l Conf. Robotics & Automation (ICRA), 2002.

[15] ——, "On Pose Recovery for Generalized Visual Sensors," IEEE Trans. Pattern Annalysis and Machine Intelligence, 2004.

[16] G. Schweighofer and A. Pinz, "Globally Optimal O(n) Solution to the PnP Problem for General Camera Models," Proc. British Machine Vision Conf. (BMVC), 2008.

[17] D. Níster and H. Stewénius, "A Minimal Solution to the Generalised 3-Point Pose Problem," J. Math Imaging Vis, 2007.

[18] S. Gasparini and V. Caglioti, "Line Localization from Single Catadioptric Images," Int'l J. of Computer Vision, 2011.

[19] V. Caglioti and S. Gasparini, "On the localization of straight lines in 3d space from single 2d images," IEEE Proc. Computer Vision and Pattern Recognition (CVPR), 2005.

[20] R. Swaminathan, A. Wu and H. Dong, "Depth From Distortions," Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS), 2008.

[21] D. Lanman, M. Wachs, G. Taubin, and F. Cukierman, "Reconstructing a 3D Line from a Single Catadioptric Image," Int'l Symposium on 3D Data Processing, Visualization and Transmission (3DPVT), 2006.

[22] L. Perdigoto and H. Araujo, "Reconstruction of 3D Lines from a Single Axial Catadioptric Image U]sing Cross-Ratio," IEEE Proc. Int'l Conf. Pattern Recognition (ICPR), 2012.

[23] H. Pottmann and J. Wallner, Computational Line Geometry. Springer-Verlag, 2001.

[24] R. Pless, "Using Many Cameras as One," IEEE Proc. Computer Vision and Pattern Recognition (CVPR), 2003.

[25] P. Sturm, "Multi-View Geometry for General Camera Models," IEEE Proc. Computer Vision and Pattern Recognition (CVPR), 2005.

[26] S. Teller and Michael Hohmeyer, "Determining the Lines Through Four Lines," J. Graphics Tools, 1999.

[27] A. Bartoli and P. Sturm, "Structure–from–motion using lines: Representation, triangulation, and bundle adjustment," Computer Vision and Image Understanding, 2005.

[28] P. Schönemann, "A Generalized Solution of the Orthogonal Procrustes Problem," Psychometrika, 1966.

[29] A. Agrawal, Y. Taguchi, and S. Ramalingam, "Beyond Alhazens Problem: Analytical Projection Model for Non-Central Catadioptric Cameras with Quadric Mirrors," IEEE Proc. Computer Vision and Pattern Recognition (CVPR), 2011.

[30] N. Gonçalves, "On the Reflection Point Where Light Reflects to a Known Destination on Quadratic Surfaces". Optical Letters, 2010.