UNIVERSIDADE DE COIMBRA
INSTITUTO DE SISTEMAS E ROBÓTICA

ARTICLE

Improving the Resolution of the Generic Camera Model By Means of a Parametric Representation

Author:
Pedro MIRALDO
miraldo@isr.uc.pt

Co–Author:
Prof. Dr. Helder ARAUJO
helder@isr.uc.pt

Contents

Improving the Resolution of the Generic Camera Model By Means of a Parametric Representation

PEDRO MIRALDO AND HELDER ARAUJO
Institute for Systems and Robotics
Department of Electrical and Computer Engineering–Polo II
University of Coimbra
3030-290 Coimbra, Portugal
{miraldo, helder}@isr.uc.pt

Abstract

Generic camera models can be used to represent any camera. These models are specially suited for non-central cameras for which closed-form models do not exist. These models are discrete and define a mapping between each pixel in the image and a straight line in 3D space. The discrete nature of the model is a disadvantage because the resolution of the model is constrained by the calibration data. In this paper we show that the use of thin-plate splines to interpolate the coordinates of the 3D lines allow both an increase in resolution (due to their continuous nature) and a more compact representation. In addition we compare several representations for 3D lines in terms of calibration accuracy.

1 Introduction

Most cameras can be modeled by a perspective projection, which implies that all projecting rays intersect at a single point. These cameras where all projecting rays intersect at a single point are usually called central cameras. However, in the last few years, cameras whose projecting rays do not satisfy the constraint of intersecting at a single effective viewpoint started to be used due essentially to the large fields of view that can be obtained. These cameras are called non-central and in many cases are obtained by combining a perspective camera with a curved mirror–the so-called catadioptric cameras. They are used in several applications ranging from robotics to visualization. Central cameras have parametric models and their calibration consists in the estimation of the parameters of those models. Non-central cameras do not have, in general, parametric models. In this article we study the calibration of generic camera models.

Calibration of a generic camera model was first discussed in Grossberg and Nayar [3, 4]. In these articles a non-parametric discrete imaging model was defined consisting in associating projecting rays in 3D space with pixels in the image. To each pixel a set of parameters called *raxel* is associated. The set of all *raxels* (representing all pixels) constitutes the complete generic imaging model.

A *raxel* is a set of parameters including image pixel coordinates, the coordinates of the associated ray in the world (position and direction) and radiometric parameters.

Grossberg and Nayar also propose a method for finding the parameters of the general imaging model. Their approach requires the acquisition of, at least, two images of a calibration object with known structure and also requires the knowledge of the object motion between the images.

Sturm and Ramalingam [8] proposed a method based on the non-parametric imaging model suggested by Grossberg and Nayar. However, they excluded the radiometry entities of the *raxel*. Their methods assumes that the camera is fully described by the coordinates of rays and mapping between rays and pixels.

Instead of using two images, Sturm and Ramalingam developed a method that requires three or more images of calibration objects, acquired from arbitrary and unknown viewing positions. If three points of the calibration object are seen in the same pixel the collinearity constraint allows the computation of the motion and as as results of the direction of the ray in 3D space.

In Barreto et al. [1] presented a generic camera calibration method to calibrate of a rigid medical endoscope based on Sturm and Ramalingam [8]. They proposed the estimation of the calibration parameters using the incidence relation of points in lines in terms of the *Plücker* coordinates, using three test planes.

All the methods described are discrete and non-parametric, using mapping arrays (rays in 3D space to pixels) to calibrate the imaging model. Image pixels have associated a set of parameters that are independent from their neighbors. Therefore, performing a complete camera calibration requires setting the mapping parameters for every pixel.

However, most of the non-central cameras that can be useful have in general pixel-ray relations that vary smoothly along the image. That is the case for non-central catadioptric systems with quadric mirrors, fisheye lens or pushbroom cameras. Smoothness in the imaging model can allow a significant reduction in the number of the unknown parameters in the calibration process and also decrease error due to noise.

In this article, we describe a parametric approximation of the general imaging model based on pixel ray relation that Grossberg and Nayar first introduced. As Sturm and Ramalingam, we consider only geometric parameters. We also present an error analysis for the case of a stereo system, a catadioptric spherical and elliptical imaging system and a central camera.

1.1 Notation

Matrices are represented as bold capital letters (e.g $\mathbf{A}$ for some matrix $\mathcal{M}_{i,j}$, i rows and j columns). Vectors are represented as bold small letters (e.g $\mathbf{a}$ for some vector $\mathcal{M}_{i,1}$, i elements) By default, a vector is considered a column. Small letters (e.g. a) represent one dimensional elements. By default, the jth column vector of $\mathbf{A}$ is specified as $\mathbf{a}_j$. The jth element of a vector $\mathbf{a}$ is written as a_j. The element of $\mathbf{A}$ in the line i and column j is represented as $a_{i,j}$.

Projective space is represented as $\mathcal{P}^n$ (in euclidean n-space). A point $\mathbf{x}$ in $\mathcal{P}^n$ can be written in homogeneous coordinates in $\mathbb{R}^{n+1}$ as $\mathbf{x} = (x_0, x_1, x_2, \ldots, x_n)$ and we can recover inhomogeneous coordinates with $\mathbf{x}' = (x_1/x_0, x_2/x_0, \ldots, x_n/x_0)$.

2 Background

2.1 *Plücker* Coordinates

Plücker coordinates are a special case of *Grassmann* coordinates. A *Grassmann* manifold is the set of k dimensional subspaces in a n dimensional vector space and is denoted as $\Lambda^k\mathbb{R}^n$. *Plücker* coordinates can be obtained as a result of the application of an extension of the exterior product to four dimensional vectors $\mathbf{x} \wedge \mathbf{y}$. The result of this operation lies in a six dimension vector space $\mathbb{R}^6$ that can represent lines in $\mathcal{P}^3$.

Consider two points ($\mathbf{x}$ and $\mathbf{w}$) in $\mathcal{P}^3$ represented in homogeneous coordinates $\mathbb{R}^4$. Using *Plücker* coordinates, we can define a line in the world as

$$\mathbf{l}\mathbb{R} = \mathbf{x} \wedge \mathbf{w} = (\underbrace{l_{01}, l_{02}, l_{03}}_{\mathbf{d}}, \underbrace{l_{23}, l_{31}, l_{12}}_{\mathbf{m}}) \in \Lambda^2\mathbb{R}^4 \subset \mathbb{R}^6 \tag{1}$$

with $l_{ij} = x_i w_j - x_j w_i$, basis $e_{ij} = e_i \wedge e_j$ (e_i are $\mathbb{R}^4$ basis) and $\mathbf{d}$ and $\mathbf{m}$ are, respectively, the direction and the moment of the line.

Although all elements of the four dimensional exterior product $\Lambda^2\mathbb{R}^4$ belong to $\mathbb{R}^6$, not all elements of $\mathbb{R}^6$ represent lines in 3D space. It can be shown that (1) is the result of a four dimensional space exterior product (and therefore is a line in 3D space) if and only if it belongs to the *Klein* quadric

$$\Omega(\mathbf{l}, \mathbf{l}) = l_{01}l_{23} + l_{02}l_{31} + l_{03}l_{12} = \langle \mathbf{d}, \mathbf{m} \rangle = 0 \tag{2}$$

We call unit *Plücker* coordinates $\bar{\mathbf{l}} = (\mathbf{d}, \mathbf{m})$ if $\langle \mathbf{d}, \mathbf{m} \rangle = 0$ and $\langle \mathbf{d}, \mathbf{d} \rangle = 1$. Note that a line in 3D space, represented as *Plücker* coordinates, is defined up to a scale factor. Thus

$$\bar{\mathbf{l}} = \mathbf{l} / \|d\| \tag{3}$$

which means that lines represented using unit *Plücker* coordinates are not defined up to a scale factor.

2.2 Interpolation

Suppose that we want to estimate an unknown function $f : \mathbb{R}^d \mapsto \mathbb{R}$ from a set of scattered data points $\mathbf{X} = \{\mathbf{x}_i\} \subset \mathbb{R}^d$ with $d \geq 1$. Let $\mathbf{y} = \{y_i\}$ be a finite set of function outputs where $y_i = f(\mathbf{x}_i)$ and $y_i \subset \mathbb{R}$.

Interpolation requires the computation of an interpolating function $s : \mathbb{R}^d \mapsto \mathbb{R}$ that satisfies

$$s(\mathbf{x}_i) = f(\mathbf{x}_i), \forall i \tag{4}$$

Radial basis functions (RBF) are a usual tool to solve that problem. Let us assume that we have $\{\mathbf{x}_1, \ldots, \mathbf{x}_P\}$. The RBF interpolating function has the form

$$s(\mathbf{x}) = a_0 + \mathbf{a}_{\mathrm{x}}^T \mathbf{x} + \sum_{i=1}^{P} w_i \phi(||\mathbf{x} - \mathbf{x}_i||) \tag{5}$$

where $||.||$ is the euclidean norm for $\mathbb{R}^d$ vectors and $\mathbf{a}_{\mathrm{x}} \in \mathbb{R}^{d-1}$.

Polyharmonic splines are one of the most common classes of RBF. This class includes *thin plate splines* $\phi(r) = r^2 log(r)$, *Gaussian* functions where $\phi(r) = exp(-r^2)$, and *multiquadrics* with $\phi(r) = (1 + r^2)^{-1/2}$. r stands for a distance, in most cases an Euclidean distance.

The interpolation problem (5) reduces to the estimation of the unknown parameters $\mathbf{a} = (a_0, \mathbf{a}_{\mathrm{x}})$ and $\mathbf{w} = (w_1, \ldots, w_p)$. The interpolating function $s(\mathbf{x})$ has $p + d + 1$ degrees of freedom and the data sets $\mathbf{X}$ and $\mathbf{Y}$ only give p equations.

To ensure a number of equations equal to the number of unknowns additional constraints have to be added. The interpolating function must have square integrable derivatives

$$\sum_{i=1}^{p} w_i = 0 \quad \& \quad \sum_{i=1}^{p} w_i x_1^i = \cdots = \sum_{i=1}^{p} w_i x_d^i = 0 \tag{6}$$

Constraints (6) and interpolating problem (5) can be rewritten in matrix form as

$$\underbrace{\begin{pmatrix} \mathbf{K} & \mathbf{P}^T \\ \mathbf{P} & \mathbf{0} \end{pmatrix}}_{\mathbf{L}} \underbrace{\begin{pmatrix} \mathbf{w} \\ \mathbf{a} \end{pmatrix}}_{\mathbf{h}_{\mathbf{w},\mathbf{a}}} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \tag{7}$$

where $\mathbf{K}$ is a symmetric $p \times p$ matrix with $k_{ij} = \phi(||\mathbf{x}_i - \mathbf{x}_j||)$, $\mathbf{P} \in \mathcal{M}_{d+1,p}$ is the stacking of column vectors $(1, \mathbf{x}_i)$. For simplicity, the $(p + d + 1) \times (p + d + 1)$ matrix in (7), is denoted as $\mathbf{L}$ and $\mathbf{h}_{\mathbf{w},\mathbf{a}}$ is the vector of the unknowns with dimension $p + d + 1$.

For the sets of correspondences $\mathbf{X}$ and $\mathbf{Y}$ the unknown coefficients of the interpolating function $s(\mathbf{x})$ can be estimated using the pseudo-inverse $\mathbf{L}^\dagger$

$$\mathbf{h}_{\mathbf{w},\mathbf{a}} = \mathbf{L}^\dagger \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \tag{8}$$

3 Calibration of the General Imaging Model

From the definition of general imaging model introduced by [3], each pixel in the image $\mathbf{x} \in \mathcal{P}^2$ is mapped to a ray in 3D space $\mathbf{l} \in \mathcal{L}^3$. Different points in the image can correspond to the same ray in the world. On the other hand, different rays in the world can not be mapped to the same image point.

A complete and highly general imaging model is represented by a non-parametric discrete array that contains all possible pixels in an image. This means that we need mappings for all the pixels, independently of the image resolution or of the smoothness on the variation of the parameters corresponding to the 3D lines associated to neighboring pixels.

Our assumption is that the pixel-ray mapping can be smoothly represented by a non-injective function $\mathbf{f} : \mathcal{P}^2 \mapsto \mathcal{L}^3$ that maps a point in the image plane to a line in 3D space. This function can be called the parametric inverse projection of general imaging model. A schematic representation of this model is shown in the figure 1.

In general imaging models, a direct projection model does not always exist since one point in the world can be mapped into more than one point in the image plane.

For a set of scattered data $\mathbf{X} = \{\mathbf{x}_i\}$ and $\mathbf{Y} = \{\mathbf{l}_i\}$, where $\mathbf{l}_i = \mathbf{f}(\mathbf{x}_i)$, the aim of this work is to find an interpolating function $\mathbf{s} : \mathcal{P}^2 \mapsto \mathcal{L}^3$ that satisfies

$$\mathbf{s}(\mathbf{x}_i) = \mathbf{f}(\mathbf{x}_i), \forall i \tag{9}$$

4 Interpolation Using *Plücker* Coordinates

Plücker coordinates, for line representation were described in §2.1. A line in 3D space is therefore specified

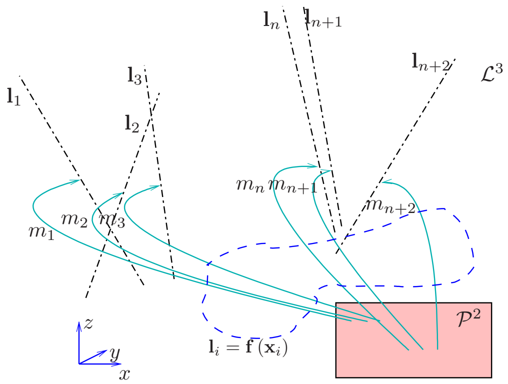


Figure 1: General imaging model with parametric modeling. m_i represents the individual mapping between pixels in $\mathcal{P}^2$ and line space in $\mathcal{L}^3$ (that represents geometric parameters of raxels). Our assumption it that this mapping is smooth and that it can be represented by a function $\mathbf{f}(\mathbf{x}): \mathcal{P}^2 \mapsto \mathcal{L}^3$

by a six dimensional vector, up to a scale factor $\mathbf{l}\mathbb{R}$. In this section we describe the estimation of smooth interpolating functions $\mathbf{s}(\mathbf{x})$, based on *scattered* data, where the output is the six dimensional *Plücker* coordinates vector.

Despite the fact that *Plücker* coordinates belong to $\mathbb{R}^6$, not every element of $\mathbb{R}^6$ define *Plücker* coordinates. In the first part of this section, we ignore that constraint, considering the interpolating function as $\mathbf{s} : \mathbb{R}^2 \mapsto \mathbb{R}^6$. In the end, we describe a method that estimates a *Plücker* vector from $\mathbb{R}^6$, in the least square sense.

The interpolating function is

$$\mathbf{s}(\mathbf{x}) = (s_1(\mathbf{x}), s_2(\mathbf{x}), \ldots, s_6(\mathbf{x}))^T \tag{10}$$

where each s_i is a RBF interpolating function §2.2. The problem is similar to (7) but the output is an euclidean six dimensional vector. Instead of considering the estimation of each $\mathbf{h}^i_{\mathbf{w},\mathbf{a}}$, as in (8), we can take into account the fact that $\mathbf{L}$ does not depend on the output element.

The fitting problem defined in(10) is reduced to the estimation of $6(p+3)$ parameters, where p is the number points used to estimate the interpolating function.

$$\mathbf{L} \underbrace{\begin{pmatrix} \mathbf{h}^1_{\mathbf{w},\mathbf{a}} & \mathbf{h}^2_{\mathbf{w},\mathbf{a}} & \ldots & \mathbf{h}^6_{\mathbf{w},\mathbf{a}} \end{pmatrix}}_{\mathbf{H}_{\mathbf{w},\mathbf{a}}} = \begin{pmatrix} \mathbf{Y}^T \\ \mathbf{0}_{3,6} \end{pmatrix} \tag{11}$$

Algorithm 1 Algorithm to correct *Plücker* coordinates based on the SVD. This algorithm seeks to determine a *Plücker* coordinate vector $\hat{\mathbf{l}} \in \Lambda^2\mathbb{R}^4$ for a given vector $\mathbf{l} \in \mathbb{R}^6$ in the least squares sense $min_{\hat{\mathbf{l}},\Omega(\hat{\mathbf{l}})=0} \left|\left|\hat{\mathbf{l}} - \mathbf{l}\right|\right|^2$

Compute the *singular value decomposition* $\begin{pmatrix} \mathbf{d} & \mathbf{m} \end{pmatrix}_{3,2} = \bar{\mathbf{U}}\bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^T$.

Let $\bar{\mathbf{Z}} = \bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^T$, and the matrix $\mathbf{T} = \begin{pmatrix} z_{21} & z_{22} \\ z_{12} & -z_{11} \end{pmatrix}$.

Compute the *singular vector* $\hat{\mathbf{v}}$ associated to the smallest singular value of matrix $\mathbf{T}$.

Define $\hat{\mathbf{V}} = \begin{pmatrix} \hat{v}_1 & -\hat{v}_2 \\ \hat{v}_2 & \hat{v}_1 \end{pmatrix}$ and set $\begin{pmatrix} \hat{\mathbf{d}} & \hat{\mathbf{m}} \end{pmatrix}_{3,2} = \bar{\mathbf{U}}\hat{\mathbf{V}} diag\left(\hat{\mathbf{V}}^T\bar{\mathbf{\Sigma}}\bar{\mathbf{V}}^T\right)$.

The closest *Plücker* coordinates of $\mathbf{l}$ are $\hat{\mathbf{l}}\mathbb{R} = \left(\hat{\mathbf{d}}, \hat{\mathbf{m}}\right)$.

The unknown elements of the interpolating function $\mathbf{H}_{\mathbf{w},\mathbf{a}} \in \mathcal{M}_{(p+3),6}$ can be computed using the pseudo-inverse $\mathbf{L}^\dagger$

$$\mathbf{H}_{\mathbf{w},\mathbf{a}} = \mathbf{L}^\dagger \begin{pmatrix} \mathbf{Y}^T \\ \mathbf{0}_{3,6} \end{pmatrix} \tag{12}$$

As described in §2.1, a point in $\mathbb{R}^6$ corresponds to the *Plücker* coordinates of a 3D line if and only if it belongs to the *Klein* conic (2). Thus, the result of $\mathbf{s}(\mathbf{x}) \in \mathbb{R}^6$ does not, in general, satisfy the Klein quadric constraint. Therefore it does not belong to the space of 3D lines. However, it is possible to estimate the closest *Plücker* coordinates $\hat{\mathbf{l}} \in \Lambda^2\mathbb{R}^4$ for a given vector $\mathbf{l} \in \mathbb{R}^6$. The method is based on SVD, and is described in algorithm 1.

4.1 Interpolation Using Line Representations with Four Coordinates

Plücker coordinates are the most general representation for 3D lines. However, this representation is not minimal. It is known that 3D lines have four degrees of freedom and there are several representations for lines in 3D space that use four independent variables.

In this section we describe three line representations using four parameters. These representations are described in [6], [5] and [7].

4.1.1 Stereographic Representation - Two Parallel Planes

The stereographic representation of the line [6] is based on the stereographic projection of the *Klein* quadric (2). This representation can be seen as defined by the coordinates of the intersection points of the 3D space line with two parallel planes $z = 0$ with $(x_1, x_2, 0)$, and $z = 1$ with $(x_3, x_4, 1)$.

$$\mathbf{l}_s = (x_1, x_2, x_3, x_4) \in \mathbb{R}^4 \tag{13}$$

Lines that do not intersect both planes can not be represented by this. This means that lines parallel to z can not be represented.

The relationship between *Plücker* coordinates and the stereographic representation array is given by

$$\mathbf{l}\mathbb{R} = (x_3 - x_1, x_4 - x_2, 1, x_2, -x_1, x_1x_4 - x_2x_3)\,\mathbb{R} \tag{14}$$

Any point in $\mathbf{l}_s \in \mathbb{R}^4$ will represent a 3D line. After converting a set of data from *Plücker* coordinates to this representation, we will have a problem of interpolating four independent functions. RBF (5) can be used as the interpolating function

$$\mathbf{l}_s(\mathbf{x}) = (s_1(\mathbf{x}), s_2(\mathbf{x}), s_3(\mathbf{x}), s_4(\mathbf{x}))^T \tag{15}$$

and the unknowns of the interpolating function can be estimated by using the following matrix representation

$$\mathbf{L}\underbrace{\begin{pmatrix} \mathbf{h}^1_{\mathbf{w,a}} & \mathbf{h}^2_{\mathbf{w,a}} & \mathbf{h}^3_{\mathbf{w,a}} & \mathbf{h}^4_{\mathbf{w,a}} \end{pmatrix}}_{\mathbf{H}_{\mathbf{w,a}}} = \begin{pmatrix} \mathbf{Y}^T \\ \mathbf{0}_{3,4} \end{pmatrix} \tag{16}$$

and therefore $\mathbf{H}_{\mathbf{w,a}} = \mathbf{L}^\dagger\left(\mathbf{Y}^T, \mathbf{0}_{3,4}\right)$, where $\mathbf{Y}$ are a set of line coordinates defined in the stereographic representation (13).

4.1.2 Representation based on Two Orthogonal Planes

Another model to represent 3D lines can be obtained by computing the intersection of each 3D line with two orthogonal planes [5]. Bartoli and Sturm [2] used this representation to define a line by means of two projections. The intersections yield two vectors in $\mathbb{R}^2$ that are the coordinates of the intersections between the line and the corresponding planes.

To define this representation we consider the intersections of the lines with the planes defined by $x = 0$ and $y = 0$. As a result this representation has a singularity: 3D lines orthogonal to the plane $z = 0$ cannot be represented. Each 3D line is represented by a four-dimensional vector

$$\mathbf{l}_p = (x_1, x_2, x_3, x_4) \in \mathbb{R}^4 \tag{17}$$

The relationship between $\mathbf{l}_p$ and the *Plücker* coordinates is given by

$$\mathbf{l}\mathbb{R} = (x_4, -x_2, x_2x_3 - x_4x_1, x_1, x_3, 1)\,\mathbb{R} \tag{18}$$

The interpolating function is similar to (15) and the unknown parameters can be estimated as in (16) where $\mathbf{Y}$ is the set of line coordinates in this representation.

4.1.3 Roberts Representation

In this section we describe a representation for 3D lines which also uses four parameters. This representation was introduced by Roberts [7]. These four parameters code two orientations and two positions.

Consider a line represented by unit *Plücker* coordinates $\bar{\mathbf{l}}$ and by the closest point to the origin $\mathbf{p} = (p_1, p_2, p_3) = \mathbf{m} \times \mathbf{d}$. Roberts uses the azimuth ϕ and elevation θ angles of the direction vector represented in spherical coordinates. The other two parameters (a and b) are the coordinates of the intersection of the line with a plane perpendicular to the line and passing through the origin. These coordinates can be written as

$$a = \left(1 - \frac{d_1^2}{1 + d_3}\right) p_1 - \frac{d_1 d_2}{1 + d_3} p_2 - d_1 p_3 \tag{19}$$

$$b = -\frac{d_1 d_2}{1 + d_3} p_1 + \left(1 - \frac{d_2^2}{1 + d_3}\right) p_2 - d_2 p_3 \tag{20}$$

where d_i is the ith element of the direction vector $\mathbf{d}$.

The line is represented with by the four-dimensional space vector

$$\mathbf{l}_r = (\phi, \theta, a, b) \in \mathbb{R}^4 \tag{21}$$

Similarly to the representation described in §4.1.1 this model for the 3D lines is given by four independent variables. Therefore, the interpolating function is similar to (15) and the unknown parameters can be estimated as in (16) where $\mathbf{Y}$ is a set of line coordinates in the Roberts representation.

5 Error Analysis

Given the four representations for 3D lines previously described it is necessary to evaluate them when used with the parametric model based on radial basis functions. For that purpose an error analysis was performed. Plücker coordinates are the most general line representation but they use six coordinates which must satisfy a non-linear constraint. The other three representations considered in this paper have four independent coordinates and therefore they constitute minimal representations for 3D lines. Their disadvantage is that not all possible 3D lines (including lines at infinity) can be represented. All four representations have singularities.

We evaluated these four 3D line representations using thin-plate splines $\phi(r) = r^2 log(r)$ as radial basis functions. Thin-plate splines were used because they generated better results in previous tests.

The experiments were performed with synthetic data generated from parametric models of non-central cameras. Specifically we considered non-central catadioptric systems made up of spherical and elliptical mirrors.

To evaluate the 3D line representations parameterized by means of thin-plate splines we used errors defined for the *Plücker* coordinates. In these coordinates a 3D line is represented by two vectors, one corresponding to the direction and the other one to the moment. For example in the case unit *Plücker* coordinates (3), the error in direction corresponds to the angular error between ground truth direction and the estimated direction. For the case of the vector of the moment the error has two components: one component corresponding to the angular error and the second component corresponding to the length or modulus of the vector.

The line representations were compared by estimating the thin-plate splines parameters with different numbers of calibration data p. The calibration data is made up of pairs of vectors. One of the vectors contains the pixel coordinates and the associated vector contains the coordinates of the 3D line representation. In the results presented in this paper we varied the number of elements (pairs of vectors) used in estimating the thin-plate splines parameters. In addition, and for each number of calibration vectors, we changed the calibration data which allowed the estimation of both the average error and the standard deviation. The errors were measured in the angles of the direction vectors, in the angles of the moment vector and also in the length of the moment vector. These results are presented in Figure 2.

6 Conclusions

In this paper we show that a parametric model for the generic camera model can be advantageously used, allowing an improvement on the resolution of the calibration model. Thin-plate splines allow for a smooth interpolation of the calibration data.

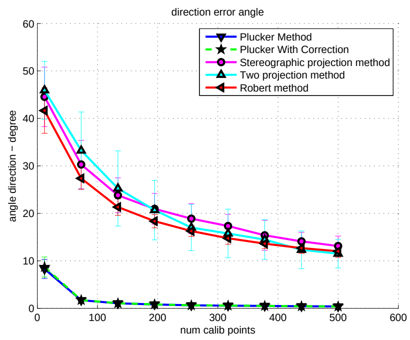
In addition we also show that the Plücker coordinates provide the best results despite the fact that this representation is not minimal in the number of coordinates.

The use of this interpolation model has also the advantage of minimizing the number of parameters in the calibration model. Even if all the calibration data ($p = N$) is used to estimate the splines parameters, the total number of parameters in this model is smaller than their number in the case of the original generalized camera model. As a matter of fact, the total number of parameters in this case (using Plücker coordinates) is $6(N+3) = 6N+18$. The original generic camera model would need at least $N(2+3+2) = 7N$ parameters (Grossberg and Nayar and Sturm and Ramalingam). Therefore in all cases for which $N > 13$ this new parametric model requires less parameters.

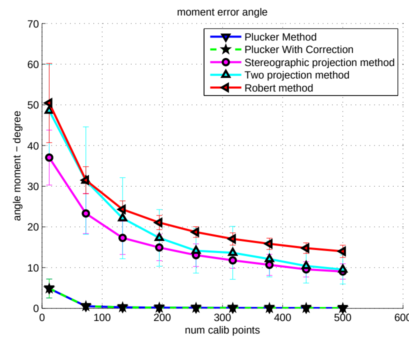
As our results show non-central catadioptric camera models can be estimated using calibration data consisting of only 200 pairs of calibrating vectors (with the Plücker coordinates). As a result the number of parameters of a generic camera model can be significantly reduced. For example, in the case of a non-central system with an image with a resolution of 3456×2304, the discrete generic camera model (Grossberg and Nayar and Sturm and Ramalingam) would need a total of $7 \times 3456 \times 2304$ parameters. The use of Plücker coordinates modeled with thin-plate splines (with $p = 200$) would require a total of 6×203 parameters, which is significantly less.

References

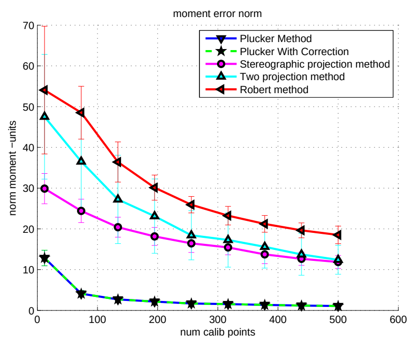
[1] Joao Barreto, J.M. Santos, P. Menezes, and F. Fonseca. Ray-based Calibration of Rigid Medical Endo-



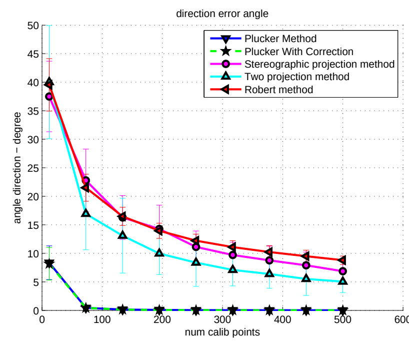
(a) Average of the angular error of the direction vector, for the spherical catadioptric system



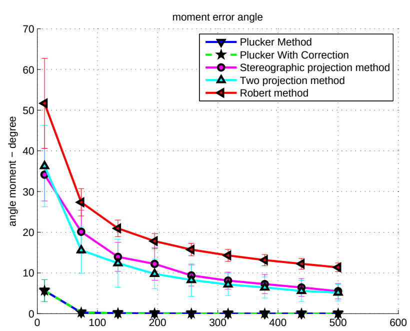
(b) Average of the angular error of the moment vector for the spherical catadioptric system



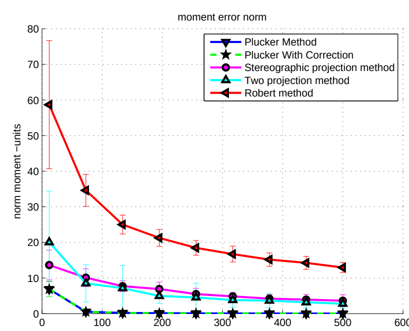
(c) Average of the error on the length of the moment vector, for the spherical catadioptric system



(d) Average of the angular error of the direction vector, for the elliptical catadioptric system



(e) Average of the angular error of the moment vector, for the elliptical catadioptric system



(f) Average of the error on the length of the moment vector, for the elliptical catadioptric system

Figure 2: Error analysis for a spherical and an elliptical catadioptric system for varying values of calibration data. 2(a), 2(b) and 2(c) are the average direction angular errors in degrees, the average moment angular errors in degrees and the average of the error on the length of the moment vector respectively for the spherical catadioptric system. 2(d), 2(e) and 2(f) are the same for the elliptical catadioptric system. Error bars represent the standard deviation of the errors at each point.

scopes. 2008. 2

[2] Adrien Bartoli and Peter Sturm. Structure-from-motion using lines: representation, triangulation, and bundle adjustment. *Comput. Vis. Image Underst.*, 100 (3):416–441, 2005. ISSN 1077-3142. 5

[3] M.D. Grossberg and S.K. Nayar. A general imaging model and a method for finding its parameters. *Computer Vision, IEEE International Conference on*, 2: 108–115, 2001. 1, 2, 3, 6

[4] M.D. Grossberg and S.K. Nayar. The raxel imaging model and ray-based calibration. 61:119 –137, 2005. 1

[5] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2000. 4, 5

[6] H Pottmann and J Wallner. *Computational Line Geometry*. Springer, 2001. 4, 5

[7] K.S. Roberts. A new representation for a line. pages 635 –640, jun 1988. 4, 5

[8] Peter Sturm and Srikumar Ramalingam. A generic concept for camera calibration. *European Conference on Computer Vision*, 2:1–13, May 2004. 1, 2, 6