Point–based Calibration Using a Parametric Representation of the General Imaging Model

Pedro Miraldo and Helder Araujo
Institute for Systems and Robotics
Department of Electrical and Computer Engineering–Polo 2
University of Coimbra
3030-290 Coimbra, Portugal
{miraldo, helder}@isr.uc.pt

João Queiró
Department of Mathematics
University of Coimbra
3001-454 Coimbra, Portugal
jfqueiro@mat.uc.pt

Abstract

Generic imaging 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. Current models are discrete and define a mapping between each pixel in the image and a straight line in $3D$ space. Due to difficulties in the calibration procedure and model complexity these methods have not been used in practice. The focus of our work was to relax these drawbacks. In this paper we modify the general imaging model using radial basis functions to interpolate image coordinates and $3D$ lines allowing both an increase in resolution (due to their continuous nature) and a more compact representation. Using this new variation of the general imaging model we also develop a new linear calibration procedure. In this process it is only required to match one $3D$ point to each image pixel. Also it is not required the calibration of every image pixel. As a result the complexity of the procedure is significantly decreased.

1. Introduction

Most cameras can be modeled by a perspective projection, which implies that all projecting rays intersect at a single point [9]. 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 catadioptric cameras [5, 1, 20, 10]. 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. A recent result, however, derived an analytical model for the forward projection of a non-central system made up of a perspective camera and a rotationally symmetric conic mirror [1]. For the special case of a spherical mirror they also derived the back-projection equations [1].

In this article, we study the calibration of generic camera models. Calibration of a generic camera model was first discussed by Grossberg and Nayar [6, 7]. 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 estimating 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 [19] and Ramalingam *et al*. [17] proposed a calibration method based on the non-parametric imaging model, suggested by Grossberg and Nayar. However, they excluded from their model the radiometric entities of the *raxel*. Their method assumes that the camera is fully described by the coordinates of rays and the mapping between rays and pixels.

Instead of using two images, Sturm and Ramalingam developed a method that requires three images of the calibration object, acquired from arbitrary and unknown viewing positions. If three points of the calibration object are seen for the same pixel, the collinearity constraint allows the computation of the motion between the images of the

978-1-4577-1102-2/11/$26.00 ©2011 IEEE

calibration object and, as a result, it allows the estimation of the direction of the ray in $3D$ space.

In [16], a minimal solution for the generic imaging model was proposed based on their previous work [19]. They showed that the minimal solution is more robust to noise when compared to the non-minimal solution.

All the methods mentioned above are discrete and non-parametric, using mapping arrays (*raxels*) 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 useful non–central cameras have, in general, a pixel–ray relations that vary smoothly along the image. That is the case for non-central catadioptric systems with quadric mirrors [5, 20, 10], or linear cameras [13, 8, 22, 23].

Our goal in this work is to relax the complexity of the general imaging model and calibration process using the assumption that the parameters of the pixel–ray relationship vary smoothly. This assumption is used to define a parametric representation for the general imaging model. As a result the number of unknown parameters can be decreased. In addition the number of parameters is independent from the image resolution. Using this representation of the general imaging model, a new method for its calibration is described. This method reduces the complexity of the calibration process. The calibration of the imaging model used in [6] and [19] requires at least two points in the world, for the same image point. We develop a calibration procedure that only needs one world point for an image point. As in [19], only geometric entities of general imaging model are considered.

2. Notation and Background

2.1. Notation

Matrices are represented as bold capital letters (*e.g.* $\mathbf{A} \in \mathbb{R}^{n\times m}$, n rows and m columns). Vectors are represented as bold small letters (*e.g.* $\mathbf{a} \in \mathbb{R}^n$, n 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 $a_{i,j}$. Regular capital letters (*e.g.* A) indicate one dimensional constants.

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 non-homogeneous coordinates with $\mathbf{x}' = (x_1/x_0, x_2/x_0, \ldots, x_n/x_0)$.

We use $\mathbb{R}$ after a vector or matrix to denote that it is represented up to a scale factor.

Let $\mathbf{U} \in \mathbb{R}^{m\times n}$ and $\mathbf{V} \in \mathbb{R}^{k\times l}$ and the equation

$$\mathbf{U}\mathbf{X}\mathbf{V}^T = \mathbf{C} \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{n\times l}$ is the equation unknown. It is possible to rewrite the previous equation as

$$\left(\mathbf{V} \otimes \mathbf{U}\right) vec\left(\mathbf{X}\right) = vec\left(\mathbf{C}\right) \tag{2}$$

where $\otimes$ is the *Kronecker* product of $\mathbf{U}$ and $\mathbf{V}$, with $(\mathbf{V} \otimes \mathbf{U}) \in \mathbb{R}^{mk\times nl}$, and $vec\left(\mathbf{X}\right)$ is a nl–vector formed by stacking the columns of $\mathbf{X}$.

2.2. *Plücker* coordinates

Plücker coordinates are a special case of *Grassmann* coordinates [14]. A *Grassmann* manifold is the set of k dimensional subspaces, in a n dimensional vector space, and it is denoted as $\Lambda^k\mathbb{R}^n$. *Plücker* coordinates can be obtained as a result of the application of an representation 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 in the world ($\mathbf{x}$ and $\mathbf{w}$ in $\mathcal{P}^3$) represented in homogeneous coordinates $\mathbb{R}^4$. Using *Plücker* coordinates, we can represent a $3D$ line, incident with both points, as

$$\mathbf{l}\mathbb{R} = \mathbf{x} \wedge \mathbf{w} = (\underbrace{l_{01}, l_{02}, l_{03}}_{\mathbf{d}}, \underbrace{l_{23}, l_{31}, l_{12}}_{\mathbf{m}}) \in \Lambda^2\mathbb{R}^4 \subset \mathbb{R}^6 \tag{3}$$

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, Equation (3) is the result of a four dimensional space exterior product (and therefore it is a line in $3D$ space), if and only if it belongs to the *Klein* quadric

$$\Omega\left(\mathbf{l}, \mathbf{l}\right) = l_{01}l_{23} + l_{02}l_{31} + l_{03}l_{12} = \langle \mathbf{d}, \mathbf{m} \rangle = 0 \tag{4}$$

One of the most important properties of *Plücker* coordinates is its ability to compute incidence relation of lines and points, in the world. Using the direction and moment vectors, a point $\mathbf{p} \in \mathcal{P}^3$ is incident with a line $\mathbf{l} \in \Lambda^2\mathbb{R}^4$ if

$$\underbrace{\begin{pmatrix} [\mathbf{p}]_\times & -\mathbf{I} \\ \mathbf{0}^T & \mathbf{p}^T \end{pmatrix}}_{\mathbf{Q}(\mathbf{p})} \mathbf{l}\mathbb{R} = \mathbf{0} \tag{5}$$

where $\mathbf{p} \in \mathbb{R}^3$ (non–homogeneous representation), $[\mathbf{a}]_\times$ is a matrix that linearizes the exterior product as $[\mathbf{a}]_\times \mathbf{b} = \mathbf{a} \times \mathbf{b}$ and $\mathbf{I}$ is a 3×3 identity matrix.

2.3. 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 a natural number) and $\mathbf{y} = \{y_i\}$, where the set $\{\mathbf{x}_i, y_i\}$ forms a training data set $\{y_i = f(\mathbf{x}_i)\}$.

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

$$s(\mathbf{x}_i) = f(\mathbf{x}_i), \quad \forall i \tag{6}$$

Radial basis functions (RBF) [3, 21, 15, 18] can be used to solve this problem. For a set P of training points $\{\mathbf{x}_1, \ldots, \mathbf{x}_P\}$, the RBF interpolant function has the form

$$s(\mathbf{x}) = a_0 + \mathbf{a}_{\mathrm{x}}^T \mathbf{x} + \sum_{i=1}^{P} w_i \phi(||\mathbf{x} - \mathbf{x}_i||) \tag{7}$$

where $||.||$ is the *2–norm* vector [4], $\phi : \mathbb{R}_+ \mapsto \mathbb{R}$ is the *radial basis function* and $\mathbf{a}_{\mathrm{x}} \in \mathbb{R}^D$. a_0, $\mathbf{a}_{\mathrm{x}}$ and w_i are the interpolant unknowns.

There are two types of kernel functions that can be used as the RBF interpolant. One type of kernel functions does not have shape parameters, like *thin–plate splines*, $\phi(r) = r^2 log(r)$, or $\phi(r) = r^2$. The other type of kernel functions does have shape parameters, such as *Gaussian* functions $\phi(r) = exp\left(-\beta^2 r^2\right)$, and *multi–quadrics* with $\phi(r) = \left(\beta^2 + r^2\right)^{1/2}$ where β is the shape parameter.

The interpolation is obtained by means of the estimation of the unknown parameters of the interpolant $\mathbf{a} = (a_0, \mathbf{a}_{\mathrm{x}})$ and $\mathbf{w} = (w_1, \ldots, w_P)$ of Equation (7). The interpolating function $s(\mathbf{x})$ has $P{+}D{+}1$ degrees of freedom and the data sets $\mathbf{X}$ and $\mathbf{y}$ only yield P equations. For the estimation of the unknowns in Equation (7), additional constraints have to be used. Since function ϕ is conditionally positive definite [21] the following equations are verified,

$$\sum_{i=1}^{P} w_i = 0 \quad \& \quad \sum_{i=1}^{P} w_i x_1^{(i)} = \cdots = \sum_{i=1}^{P} w_i x_D^{(i)} = 0 \tag{8}$$

[21, 11] where $x_i^{(j)}$ is the ith element of the jth observation. The use of these constraints allows the estimation of all unknowns.

3. Parametric Representation of the General Imaging Model

From the definition of general imaging model, introduced by Grossberg and Nayar [6], each pixel in the image $\mathbf{x} \in \mathcal{P}^2$ is mapped to a ray in $3D$ space $\mathbf{l} \in \mathcal{L}^3$. As mentioned before, the model is based on an array of parameters called *raxel*.

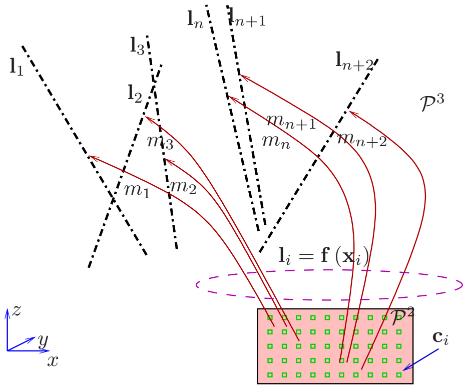


Figure 1: General imaging model with parametric modeling. m_i represents the individual mapping between pixels in $\mathcal{P}^2$ and the line space in $\mathcal{L}^3$ (that represents geometric parameters of the raxels). Our assumption is that this mapping is smooth and that it can be represented by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

In their model, a complete general imaging model is represented by a non-parametric discrete array of *raxels*, that contains all possible pixels in an image. This means that the pixel–line mappings are required for all pixels. Independently of the image resolution or of the smoothness on the variation of the parameters, that correspond to the $3D$ lines associated to neighboring pixels. If we consider only the geometric entities in Grossberg and Nayar's model, each *raxel* contains at least seven parameters. Thus, for each pixel, there are seven unknown parameters to be estimated. For an image with size $N \times M$, there are $7NM$ parameters to be estimated. Our target in this section is to reduce the number of parameters to be estimated.

Our assumption is that the pixel–line mapping can be represented by a smoothly varying vector–valued 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 assumption can significantly decrease the number of model unknowns and also filter out some error due to noise.

Different points in the image can correspond to the same ray in the world. On the other hand, in a general imaging system, different rays in the world can not be mapped to the same image point. Thus, the general vector–valued function $\mathbf{l} = \mathbf{f}(\mathbf{x})$ takes a non–injective form. In most cases, a general direct projection model does not exist, since one $3D$ line can be mapped into more than one point in the image plane. As a result, a general imaging model can only be defined when considering the mapping from image coordinates to $3D$ lines.

A schematic representation of this model is shown in the Figure 1.

3.1. Non–Injective Vector–Valued Function

A $3D$ line representation has to be chosen for the output of the vector–valued function. Lines in $3D$ have four degrees of freedom. However, none of compact four variable representations for $3D$ lines is complete.

Plücker coordinates (Section 2.2) are a complete, elegant and easy to understand line representation. On the other hand, it has six elements to represent four degrees of freedom. It is defined up to a scale factor and has an orthogonal constraint associated to its elements.

Instead of using *Plücker* coordinates as estimator, we use a vector as the stacking of direction and moment $\hat{\mathbf{l}}\mathbb{R} = \left(\hat{\mathbf{d}}, \hat{\mathbf{m}}\right)$. In other words, we are estimating two independent vectors, up to the same scale factor.

Good estimates of $\hat{\mathbf{d}}$ and $\hat{\mathbf{m}}$ yield small deviations from the orthogonal constraint. However, it is possible to find orthogonal vectors $\mathbf{d}$ and $\mathbf{m}$ from their estimates using *Schmidt* orthogonalization [4], by finding the closest rotation matrix to $\left(\begin{array}{ccc} \bar{\hat{\mathbf{d}}} & \bar{\hat{\mathbf{m}}} & \bar{\hat{\mathbf{d}}} \times \bar{\hat{\mathbf{m}}} \end{array} \right)$, where $\bar{\mathbf{x}} = \mathbf{x} / \left|\left|\mathbf{x}\right|\right|$, or by using the algorithm proposed by Bartoli and Sturm [2].

There are several ways to estimate a non–injective function from a set of scattered data. We use the RBF interpolant described in Section 2.3

$$s\left(\mathbf{x}\right) = a_0 + \mathbf{a}_{\mathrm{x}}^T \mathbf{x} + \sum_{i=1}^{P} w_i \phi\left(\left|\left|\mathbf{x} - \mathbf{c}_i\right|\right|\right) \tag{9}$$

where $\left\{\mathbf{c}_i\right\}, \mathbf{x} \in \mathbb{R}^2$. We can rewrite the previous equation, in matrix form as

$$s\left(\mathbf{x}\right) = \left(\begin{array}{cc} \boldsymbol{\phi}\left(\mathbf{x}\right) & \mathbf{p}\left(\mathbf{x}\right) \end{array} \right) \underbrace{\left(\begin{array}{c} \mathbf{w} \\ \mathbf{a} \end{array} \right)}_{\mathbf{h_{wa}}} \tag{10}$$

where $\mathbf{a} = \left(a_0, \mathbf{a}_{\mathrm{x}}\right)$ and $\mathbf{w} = \left(w_1, w_2, \ldots, w_P\right)$. $\phi_i\left(\mathbf{x}\right) = \phi\left(\left|\left|\mathbf{x} - \mathbf{c}_i\right|\right|\right)$. $\mathbf{p}\left(\mathbf{x}\right) = \left(\begin{array}{ccc} 1 & x_1 & x_2 \end{array} \right)$. $\boldsymbol{\phi}\left(\mathbf{x}\right)$ and $\mathbf{p}\left(\mathbf{x}\right)$ are row vectors.

The vector–valued function output is $\hat{\mathbf{l}} \in \mathbb{R}^6$ where each $\hat{l}_i$ is independent. Thus, we can use six independent RBF interpolants to form the vector–valued function $\mathbf{s}\left(\mathbf{x}\right) = \left(\begin{array}{cccc} s_1\left(\mathbf{x}\right) & s_2\left(\mathbf{x}\right) & \ldots & s_6\left(\mathbf{x}\right) \end{array} \right)$, which can be rewrote as

$$\mathbf{s}\left(\mathbf{x}\right) = \left(\begin{array}{cc} \boldsymbol{\phi}\left(\mathbf{x}\right) & \mathbf{p}\left(\mathbf{x}\right) \end{array} \right) \underbrace{\left(\begin{array}{cccc} \mathbf{h}_{\mathbf{wa}}^{(1)} & \mathbf{h}_{\mathbf{wa}}^{(2)} & \ldots & \mathbf{h}_{\mathbf{wa}}^{(6)} \end{array} \right)}_{\mathbf{H_{wa}}} \tag{11}$$

The vector–valued function is a row vector, $\mathbf{s}\left(\mathbf{x}\right) \in \mathbb{R}^{1\times 6}$, which implies $\hat{\mathbf{l}}\mathbb{R} = \mathbf{s}^T\left(\mathbf{x}\right)$.

For a set $\left\{\mathbf{c}_i\right\}$, a matrix $\mathbf{H_{wa}}$ and a certain RBF, the estimates of the direction and moment vectors are given by

$$\hat{\mathbf{l}}^T\mathbb{R} = \left(\begin{array}{cc} \hat{\mathbf{d}}^T & \hat{\mathbf{m}}^T \end{array} \right)\mathbb{R} = \left(\begin{array}{cc} \boldsymbol{\phi}\left(\mathbf{x}\right) & \mathbf{p}\left(\mathbf{x}\right) \end{array} \right)\mathbf{H_{wa}} \tag{12}$$

For two different imaging systems, using the same set of points $\left\{\mathbf{c}_i\right\}$, the estimation of $\hat{\mathbf{l}}$ for an image point $\mathbf{x}$ only depends on the matrix $\mathbf{H_{wa}}$. Thus, we call $\mathbf{H_{wa}}$ the *camera matrix*. On the other hand, for the same imaging system, the values of the parameters of the *camera matrix* depend on the set $\left\{\mathbf{c}_i\right\}$, and that is why we call them *control points*.

Usually, in statistics, the set $\left\{\mathbf{c}_i\right\}$ is called *centers*. In computer vision, the word center in an imaging system typically designates the center of the projection. As a result, we chose to name to the set $\left\{\mathbf{c}_i\right\}$ as *control points*.

For a set P of *control points* defined *a priori* $\left\{\mathbf{c}_i\right\}$ and a *camera matrix* $\mathbf{H_{wa}} \in \mathbb{R}^{(P+3)\times 6}$, it is possible to define a generic smooth general imaging model by a vector–valued function $\mathbf{s} : \mathbb{R}^2 \mapsto \mathbb{R}^6$.

The general imaging model only depends on the unknown matrix $\mathbf{H_{wa}}$, for a set of previously defined *control points*. Therefore, the complete calibration of general imaging models is obtained by estimating $6P + 18$ unknown parameters, that sets up the *camera matrix*.

4. Point–based Calibration

One significant disadvantage in the use of the general imaging model is the difficulty in its calibration. Grossberg and Nayar [6] and Sturm and Ramalingam [19] define two methods for the calibration. The complete calibration is achieved when each image pixel has a *raxel* associated to it. To estimate one *raxel*, at least two $3D$ points are required for each same image point.

The method presented in this paper aims at relaxing the calibration procedure, by using the parametric representation of the general imaging model described in Section 3. The goal is to calibrate a general imaging model without estimating each and all the *raxels* while, at same time, allowing the same resolution. In addition, the method should require only a set of point correspondences $\left\{\mathbf{x}_i \mapsto \mathbf{p}_i\right\}$, ($\left\{\mathbf{p}_i\right\}$ is the set of world points and $\left\{\mathbf{x}_i\right\}$ its correspondents points in the image plane), where we can use points correspondences that satisfy

$$\mathbf{x}_i \neq \mathbf{x}_j, \quad \forall_{i \neq j} \tag{13}$$

We never need to know more than one world points, for a point in the image plane.

There are two sets of unknowns in the calibration procedure: the set of *control points* $\left\{\mathbf{c}_i\right\}$ and the *camera matrix* $\mathbf{H_{wa}}$ elements. *Control points* can be defined *a priori*, by selecting a set P of scattered image points, that can or cannot be a subset of data points $\left\{\mathbf{c}_i\right\} \subset \left\{\mathbf{x}_i\right\}$.

In the rest of this section, we describe a linear method to estimate $6P + 18$ parameters of the *camera matrix*, for a set of *control points* defined *a priori*.

4.1. Calibration Matrix M

World points incident with lines must satisfy Equation (5). From function $\hat{\mathbf{l}}\mathbb{R} = \mathbf{s}\left(\mathbf{x}\right)^T$ and Equation (5)

$$\left(\mathbf{Q}\left(\mathbf{p}\right)\hat{\mathbf{l}}\mathbb{R}\right)^T = \left(\hat{\mathbf{l}}\mathbb{R}\right)^T \mathbf{Q}\left(\mathbf{p}\right)^T = \mathbf{0} \quad (14)$$

$$\mathbf{s}\left(\mathbf{x}\right)\mathbf{Q}\left(\mathbf{p}\right)^T = \mathbf{0} \quad (15)$$

Replacing $\mathbf{s}\left(\mathbf{x}\right)$, in Equation (15) by Equation (11)

$$\underbrace{\begin{pmatrix} \phi\left(\mathbf{x}\right)^T & \mathbf{p}\left(\mathbf{x}\right)^T \end{pmatrix}}_{\mathbf{r}(\mathbf{x})} \mathbf{H_{wa}}\mathbf{Q}\left(\mathbf{p}\right)^T = \mathbf{0} \quad (16)$$

where $\mathbf{r}\left(\mathbf{x}\right) \in \mathbb{R}^{1\times(P+3)}$.

The unknowns are the elements of the matrix $\mathbf{H_{wa}}$. Thus, we use *Kronecker* product, Section 2.1, that allow us to rewrite (16) in order to isolate the unknown *camera matrix* as

$$\left[\mathbf{Q}\left(\mathbf{p}\right) \otimes \mathbf{r}\left(\mathbf{x}\right)\right] vec\left(\mathbf{H_{wa}}\right) = \mathbf{0} \quad (17)$$

where the matrix $\mathbf{Q}\left(\mathbf{p}\right) \otimes \mathbf{r}\left(\mathbf{x}\right) \in \mathbb{R}^{4\times(6P+18)}$ and $vec\left(\mathbf{H_{wa}}\right) \in \mathbb{R}^{(6P+18)}$ is the stacking of $\mathbf{h}_{\mathbf{wa}}^{(i)}$ columns, for $i = 1,\ldots,6$.

For a set N of point correspondences $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ of the same imaging system, $\left[\mathbf{Q}\left(\mathbf{p}_i\right) \otimes \mathbf{r}\left(\mathbf{x}_i\right)\right] vec\left(\mathbf{H_{wa}}\right) = \mathbf{0}$ and we can build the calibration matrix

$$\mathbf{M} = \begin{pmatrix} \mathbf{Q}\left(\mathbf{p}_1\right) \otimes \mathbf{r}\left(\mathbf{x}_1\right) \\ \mathbf{Q}\left(\mathbf{p}_2\right) \otimes \mathbf{r}\left(\mathbf{x}_2\right) \\ \vdots \\ \mathbf{Q}\left(\mathbf{p}_N\right) \otimes \mathbf{r}\left(\mathbf{x}_N\right) \\ \mathbf{D} \end{pmatrix} \quad (18)$$

where $\mathbf{M} \in \mathbb{R}^{(4N+18)\times(6P+18)}$ and matrix $\mathbf{D} \in \mathbb{R}^{18\times(6P+18)}$ is as [11].

Calibration procedure is reduced to the estimation of the unknown *camera matrix* $\mathbf{H_{wa}}$ such that

$$\mathbf{M} vec\left(\mathbf{H_{wa}}\right) = \mathbf{0} \quad (19)$$

that means

$$vec\left(\mathbf{H_{wa}}\right) \in null\left(\mathbf{M}\right) \quad (20)$$

where $null\left(.\right)$ indicates matrix *null–space*.

4.2. Computation of the *Camera Matrix*

From Equation (20), it results that the estimate of the *camera matrix* must belong the the *null–space* of the calibration matrix $\mathbf{M}$.

To ensure an unique solution, and since $vec\left(\mathbf{H_{wa}}\right) \in \mathbb{R}^{6P+18}$, we have to make sure that the dimension of the *column space* of $\mathbf{M}$ is $\mathcal{C}\left(\mathbf{M}\right) = 6P + 17$, which means that the dimension of the *null–space* of $\mathbf{M}$ must be $\mathcal{N}\left(\mathbf{M}\right) = 1$.

From Equation (12), any solution for the *camera matrix* is defined up to a scale factor. Thus, assuming that $\mathcal{N}\left(\mathbf{M}\right) = 1$, any element of the one dimensional *null–space* of $\mathbf{M}$ is solution, except for the *trivial solution* $vec\left(\mathbf{H_{wa}}\right) = \mathbf{0}$.

To prove that the dimension of the *column space* of $\mathbf{M}$ is $\mathcal{C}\left(\mathbf{M}\right) = 6P + 17$, we decompose Equation (18) into rows, by means of the *Kronecker* product. Permutation of the rows in any matrix does not change the dimension of the *column space*. Making $N = 2P$, and for a matrix $\mathbf{D} \in \mathbb{R}^{18\times 6P+18}$ it can be proved that $\mathcal{C}\left(\mathbf{M}\right) = 6P + 17$–see next section and [11].

The computation of the dimension of the *column space* and coresponding constraints is described in [11].

4.3. Relationship between *Control Points*, point correspondences and *radial basis function* used in the calibration

In this section, we describe the constraints that must be met by $\{\mathbf{x}_i\}$ and $\{\mathbf{c}_i\}$ to obtain $\mathcal{C}\left(\mathbf{M}\right) = 6P + 17$. We also suggest two types of *radial basis functions*, that can be used to get $\mathcal{C}\left(\mathbf{M}\right) = 6P + 17$.

Consider that we have a set of point correspondences, $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ for $i = 1,\ldots,2P$, where $\mathbf{x}_i \neq \mathbf{x}_j, \quad \forall i \neq j$. Next split up the set $\{\mathbf{x}_i\}$ into two sub-sets $X = \{\mathbf{x}_i\}$, for $i = 1,\ldots,P$ and $Y = \{\mathbf{x}_j\}$, for $j = P+1,\ldots,2P$.

In [15] and [11] it is shown that only *control points* ($\{\mathbf{c}_i\}$, for $i = 1,\ldots,P$) and *data points* ($\{\mathbf{x}_j\}$, for $j = 1,\ldots,2P$) that meet the condition

$$d < \epsilon q \quad (21)$$

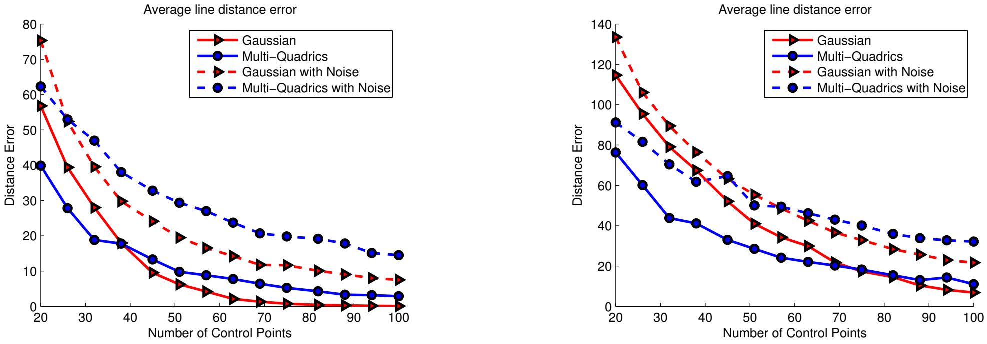
can be considered. In Equation (21), $d = \min\{d_1, d_2\}$, $0 < \epsilon \leq 1$, $d_1 = \max\{||\mathbf{x}_i - \mathbf{c}_i||\}$, where the set $\{\mathbf{x}_i\}$ belongs to the set X, $d_2 = \max\{||\mathbf{x}_i - \mathbf{c}_i||\}$, where the set $\{\mathbf{x}_i\}$ belongs to Y and $2q = \min_{j\neq i}\{||\mathbf{c}_i - \mathbf{c}_j||\}$.

Note that (from [11]) if we consider $\{\mathbf{c}_i\} = X$ and if $d_2 < \epsilon q$ then we can also obtain $\mathcal{C}\left(\mathbf{M}\right)) = 6P + 17$. This solution also meets the constraint $\mathbf{x}_i \neq \mathbf{x}_j, \quad \forall i \neq j$.

Quak *et al*. [15] proved that $\phi_1\left(r\right) = \left(\beta_1^2 + r^2\right)^{1/2}$ and $\phi_2\left(r\right) = e^{-\beta_2 r^2}$ are good choices for *radial basis functions*, because, choosing an appropriate β_1 and β_2, they reduce the negative effects of small values of q and ϵ respectively, in Equation (21).

5. Experiments

The calibration method and general imaging model described in this paper were evaluated using both synthetic data sets and real data sets.



(a) Sphere mirror

(b) Hyperbolic mirror

Figure 2: Evaluation of the average line distance error, Equation (24), as a function of the number of *control points*, for non–central catadioptric systems with sphere (a) and hyperbolic mirror (b). Dashed lines correspond to the average errors in the cases where Gaussian random noise was added to the coordinates.

5.1. Results with Synthetic Data Sets

Synthetic data sets were used to evaluate the effect of a varying number of *control points* on a error measure defined as a distance between $3D$ lines. We used *multi–quadrics* $\phi_1(r) = \left(\beta_1^2 + r^2\right)^{1/2}$ and *exponential* $\phi_2(r) = e^{-\beta_2 r^2}$ as *radial basis functions*.

The synthetic data sets were obtained for non–central catadioptric systems [5]. Two types of quadric mirrors were considered: hyperbolic and spherical mirrors.

The calibration method estimates an interpolant function $\mathbf{s} : \mathbb{R}^2 \mapsto \mathbb{R}^6$ that should fit an imaging model, defined by a function $\mathbf{f} : \mathcal{P}^2 \mapsto \mathcal{L}^3$, where $\mathbf{f}$ is an analytical representation of the corresponding imaging model. In the case of a non–central catadioptric system made up of a perspective camera and hyperbolic or spherical mirror, an analytical expression for $\mathbf{f}$ can be obtained [5, Section 4.3].

We use $\mathbf{f}$ to generate a data set of $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, for $i = 1, \ldots, N$ and select P *control points* from the set $\{\mathbf{x}_i\}$, where the conditions described in Section 4.3 are met.

Using $\mathbf{f}$, we generate a ground truth data set $\{\mathbf{y}_i \mapsto \mathbf{l}_i\}$, where $\mathbf{y}_i$ are the ground truth image coordinates and $\mathbf{l}_i$ are the ground truth line coordinates.

We use the set $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ and $\{\mathbf{c}_i\}$ to calibrate the imaging system. With the interpolant functions $\mathbf{s}$, we generate the estimated set $\left\{\mathbf{y}_i \mapsto \hat{\mathbf{l}}_i\right\}$, where $\hat{\mathbf{l}}_i = \mathbf{s}(\mathbf{y}_i)$ and $\mathbf{s}$ are defined in Section 3.1.

We want to measure the deviation of $\hat{\mathbf{l}}_i$ from $\mathbf{l}_i$. To characterize the error we use the average of the distances $\epsilon_i = d\left(\mathbf{l}_i, \hat{\mathbf{l}}_i\right)$.

Consider two lines represented as *Plücker* coordinates $\mathbf{g}\mathbb{R}$ and $\mathbf{h}\mathbb{R}$ that can be approximated via local mappings into euclidean 4-space $\tilde{\mathbf{g}} = (g_1, g_2, g_3, g_4)$ and $\tilde{\mathbf{h}} = (h_1, h_2, h_3, h_4)$ [14] such that

$$\mathbf{g} = (g_3 - g_1, g_4 - g_2, 1, g_2, -g_1, g_1 g_4 - g_2 g_3) \quad (22)$$

$$\mathbf{h} = (h_3 - h_1, h_4 - h_2, 1, h_2, -h_1, h_1 h_4 - h_2 h_3) \quad (23)$$

The distances between the lines are estimated using

$$d\left(\tilde{\mathbf{g}}, \tilde{\mathbf{h}}\right)^2 = \sum_{i=1}^{4} (g_i - h_i)^2 + (g_1 - h_1)(g_3 - h_3) + (g_2 - h_2)(g_4 - h_4) \quad (24)$$

We evaluate the results by varying the number of *control points*, P, from 20 to 100, and with a set of point correspondences $N = 2P$. For each number of *control points* (P), we evaluate the calibration using the ground truth data set $\{\mathbf{y}_i \mapsto \mathbf{l}_i\}$ for $i = 1, \ldots, 120$. For each set of *control points*, we repeat the calibration 200 times, where the sets $\{\mathbf{x}_i\}$ and $\{\mathbf{y}_i\}$ are chosen randomly.

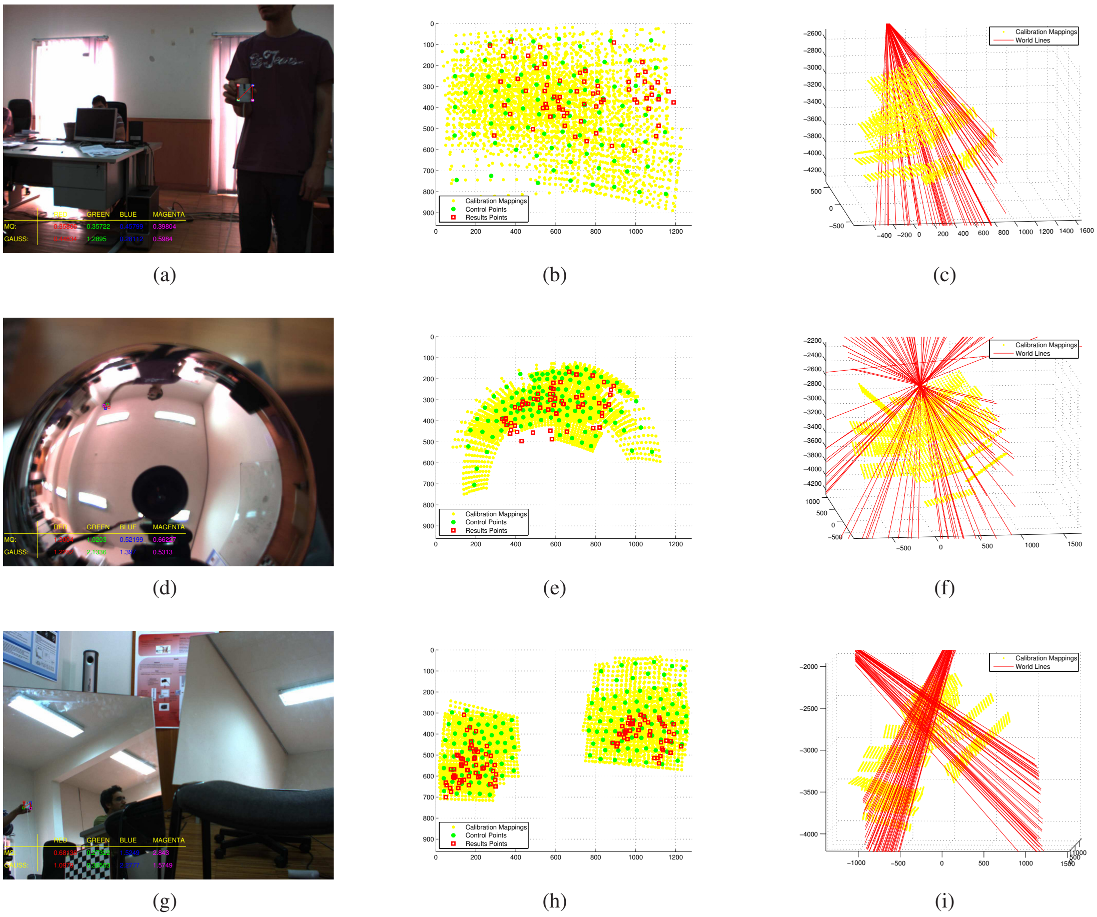
We repeat the same procedure adding Gaussian random noise $\{\boldsymbol{\mu}_i\}$ to the world coordinates $\{\mathbf{p}_i + \boldsymbol{\mu}_i\}$. The standard deviation of the Gaussian noise added to the coordinates $\{\boldsymbol{\mu}_i\}$ was $std\left(\mu_i^{(j)}\right) = 0.25e$ where $\boldsymbol{\mu}_i = \left(\mu_i^{(1)}, \mu_i^{(2)}, \mu_i^{(3)}\right)$ and $e = \min\{||\mathbf{p}_i - \mathbf{p}_j||\}$, for all $i \neq j$.

Results are shown in Figure 2.

5.2. Experiments Using Real Data Sets

For experiments with real data sets, we calibrated three different types of imaging systems, using our representation of the general imaging model and its point–based calibration. We used a projective camera and two different catadioptric systems, Figure 3a, 3d and 3g respectively.

To acquire the point correspondences $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ for the calibration process, we used a chess board. We attached to the chess board infrared (IR) LEDs and their positions in the



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 3: Results with real data generated by our approach of the general imaging model, for three types of camera models: perspective camera (a), catadioptric system with spherical mirror (d) and catadioptric system with two planar mirrors (g). Yellow points in (b), (e) and (h) are the set of image coordinates $\{\mathbf{x}_i\}$ and yellow points in (c), (f) and (i) are the corresponding set of world points $\{\mathbf{p}_i\}$, used in the calibration process. Red squares in the $2D$ plot and the red rays in the $3D$ plot are the a subset of $\left\{\mathbf{y}_i \mapsto \hat{\mathbf{l}}_i\right\}$ produced with *multi–quadrics* RBF. Green points in (b), (e) and (h), are the corresponding set of *control points* used in the calibration.

world are measured using an IR tracker [12]. This tracker has an accuracy of $0.1mm$ and a resolution of $0.01mm$.

Each corner of the chess board in the image $(\mathbf{x}_i)$ is associated to a position in the world $(\mathbf{p}_i)$, which is given by its corresponding position in the chess board. The association set $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ will be used in the calibration process.

The set of image points $\{\mathbf{x}_i\}$ are the yellow points in Figures 3b, 3e and 3h. The set of world points $\{\mathbf{p}_i\}$ are the yellow points in Figures 3c, 3f and 3i.

Control points $\{\mathbf{c}_i\}$ are chosen as a subset of $\{\mathbf{x}_i\}$. They are shown as green points, in Figures 3b, 3e and 3h.

Using $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, we estimate the interpolant function $\mathbf{s}$. Since, in the real experiments, we have $N > 2P$, we will have an over–determined solution. The solution for the *camera matrix* is obtained using a least–squares solution for the homogeneous equations [9].

To evaluate the calibration, 3D coordinates from a different object were used. The IR tracker has a ”test object” (also with LEDs) which is provided to enable the estimation of 3D coordinates of points. This different object was used to generate a new data set $\{\mathbf{y}_i \mapsto \mathbf{w}_i\}$, where $\mathbf{y}_i$ are image points and $\mathbf{w}_i$ are world points. This new data set was used to evaluate the calibration performed with the former data set.

The distance error is defined by the distance from the world point $\mathbf{p}_i$ to the generated line $\hat{\mathbf{l}}_i$, where $\hat{\mathbf{l}}_i = \mathbf{s}(\mathbf{y}_i)$. A subset of $\{\mathbf{y}_i\}$ and corresponding lines $\left\{\hat{\mathbf{l}}_i\right\}$ is shown as red squares, in Figures 3b, 3e and 3h, and as red lines, in Figure 3c, 3f and 3i, respectively.

RBF	*Gaussian* [cm]	*Multi–quadrics* [cm]
Projective	1.1488 ± 3.1436	0.5651 ± 0.4373
Sphere mirror	3.0164 ± 6.2657	1.4592 ± 0.9974
Plane mirrors	0.9972 ± 2.1595	0.9738 ± 2.4991

Table 1: Average and standard deviation of the distances, in centimeters, between points in the world and lines generated by our representation of the general imaging model. We use the set $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ for $i = 1, \ldots, 700$ as calibration set and the set $\left\{\mathbf{y}_i \mapsto \hat{\mathbf{l}}_i\right\}$, where $i = 1, \ldots, 3840$, for each imaging system, as the test set.

The geometric distance between a line (in *Plücker* coordinates) and a point in the world is given by $\epsilon_i = ||\boldsymbol{\epsilon}_i||$, where

$$\boldsymbol{\epsilon}_i = \begin{pmatrix} [\mathbf{w}_i]_\text{x} & -\mathbf{I} \end{pmatrix} \bar{\mathbf{l}}_i \tag{25}$$

and $\bar{\mathbf{l}}_i = \hat{\mathbf{l}}_i / \, ||\mathbf{d}_i||$.

The number of *control points* used in the calibration were 90, 120 and 150, for the imaging systems of Figures 3a, 3d and 3g respectively. The results are shown in Table 1 as well as in the video, provided as supplementary material [11].

6. Conclusions

The calibration method described in this paper can be used in the calibration of complex camera models, namely for cameras which no analytical projection model exists.

The approach is based on a parametric version of the generic imaging model. The approach described in this paper can model an imaging model with significantly less parameters than the discrete generic imaging model. For the model described in this paper the number of parameters does not depend on the image size. Instead of the $7NM$ parameters, for an $M \times N$ image, required by the discrete generic imaging model, this approach only requires $6\,(P+3)$ for P *control points*.

The calibration procedure described in this paper only requires the 3D coordinates of a world point for each image point, whereas previous approaches require two or more 3D points for each image point. On the other hand, the calibration parameters which are estimated for a sub-set of image points can be generalized for all image pixels, which constitutes an important advantage of this method.

References

[1] A. Agrawal, Y. Taguchi, and S. Ramalingam. Analytical forward projection for axial non-central dioptric & catadioptric cameras. *In ECCV*, 2010. 1

[2] A. Bartoli and P. Sturm. Structure-from-motion using lines: representation, triangulation, and bundle adjustment. *Computer Vision and Image Understanding*, 100(3), 2005. 4

[3] M. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 8RU, UK, 2003. 3

[4] G. H. Golub and C. F. Van Loan. *Matrix computations (3rd ed.)*. Johns Hopkins University Press, Baltimore, MD, USA, 1996. 3, 4

[5] N. Goncalves. *Noncentral Catadioptric Systems with Quadric Mirrors, Geometry and Calibration*. PhD thesis, University of Coimbra, October 2008. 1, 2, 6

[6] M. Grossberg and S. Nayar. A general imaging model and a method for finding its parameters. *In ICCV*, 2001. 1, 2, 3, 4

[7] M. Grossberg and S. Nayar. The raxel imaging model and ray-based calibration. *International Journal of Computer Vision*, 61:119 –137, 2005. 1

[8] R. Gupta and R. I. Hartley. Linear pushbroom cameras. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 19:963–975, 1997. 2

[9] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision (2nd ed.)*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 2RU, UK, 2000. 1, 7

[10] B. Micusik and T. Pajdla. Autocalibration 3d reconstruction with non-central catadioptric cameras. *In CVPR*, 2004. 1, 2

[11] P. Miraldo, H. Araujo, and J. Queiró. Unique solution for the estimation of the plucker coordinates using radial basis functions. *In ICCV – Supplement Material*, 2011. www.deec.uc.pt/~miraldo/pub/iccv2011.zip. 3, 5, 8

[12] Northern Digital Incorporated. Optotrak certus motion capture system, 2009. www.ndigital.com. 7

[13] J. Ponce. What is a camera? *In CVPR*, 2009. 2

[14] H. Pottmann and J. Wallner. *Computational Line Geometry*. Springer–Verlag, Berlin, 2001. 2, 6

[15] E. Quak, N. Sivakumar, and D. J. Ward. Least squares approximation by radial functions. *SIAM Journal on Mathematical Analysis*, 24(4):1043–1066, 1993. 3, 5

[16] S. Ramalingam and P. Sturm. Minimal solutions for generic imaging models. *In CVPR*, 2008. 2

[17] S. Ramalingam, P. Sturm, and S. Lodha. Towards complete generic camera calibration. *In CVPR*, 2005. 1

[18] N. Sivakumar and J. D. Ward. On the least squares fit by radial functions to multidimensional scattered data. *Numerische Mathematik*, 65:219–243, 1993. 3

[19] P. Sturm and S. Ramalingam. A generic concept for camera calibration. *In ECCV*, 2004. 1, 2, 4

[20] R. Swaminathan, M. Grossberg, and S. Nayar. Non-single viewpoint catadioptric cameras: Geometry and analysis. *International Journal of Computer Vision*, 66:211–229, 2006. 1, 2

[21] H. Wendland. *Scattered Data Approximation*. Cambridge University Press, The Edinburgh Building, Cambridge CB2 8RU, UK, 2005. 3

[22] J. Yu and L. McMillan. General linear cameras. *In ECCV*, 2004. 2

[23] A. Zomet, D. Feldman, S. Peleg, D. Weinshall, and I. C. Society. Mosaicing new views: The crossed-slits projection. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 25:741–754, 2003. 2