

Calibration of Smooth Camera Models

Pedro Miraldo and Helder Araujo

Abstract—Generic imaging models can be used to represent any camera. Current generic models are discrete and define a mapping between each pixel in the image and a straight line in 3D space. This paper presents a modification of the generic camera model that allows the simplification of the calibration procedure. The only requirement is that the coordinates of the 3D projecting lines are related by functions that vary smoothly across space. Such model is obtained by modifying the general imaging model using *radial basis functions* to interpolate image coordinates and 3D lines, thereby allowing both an increase in resolution (due to their continuous nature) and a more compact representation. Using this variation of the general imaging model we also develop a calibration procedure. This procedure only requires that a 3D point be matched to each pixel. In addition not all the pixels need to be calibrated. As a result the complexity of the procedure is significantly decreased. Normalization is applied to the coordinates of both image and 3D points which increases the accuracy of the calibration. Results with both synthetic and real data sets show that the model and calibration procedure are easily applicable and provide accurate calibration results.

Index Terms—General Camera Models, Camera Calibration, Smooth Vector-Valued Functions.



1 INTRODUCTION

MOST cameras can be modeled by a perspective projection, which implies that all projecting rays intersect at a single point. These cameras are usually called central cameras. The calibration process for perspective cameras model (pinhole model) is well known and easy to implement [1], [2], [3]. Other image models that verify the single view point constraint but do not satisfy the pinhole model have been defined, for example special cases of catadioptric image models [4], [5]. Moreover, central image models with non-parametric association from image pixels to rays, (*General Central Image Model*) and their respective calibration have also been studied [6], [7], [8].

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 a generic combination of a perspective camera and a curved mirror—the non-central catadioptric cameras [9], [10], [11], [12] (Figure 3(a)). They are used in several applications ranging from robotics to visualization.

Many other non-central image models have also been developed such as linear cameras [13], [14], [15], [16] (Figure 3(d)), fisheye cameras [17], [18] or image models that include refractive elements [19], [20] (Figure 3(g)). All of these systems have, in general, models that are characterized by a small number of parameters. Since the

number of parameters is small the calibration procedures are in general, simple.

All of these models are specific since they are derived using geometric models of the cameras [21] and therefore they lack generality. To deal with these drawbacks, some techniques were developed to calibrate special classes of cameras such as [22], [23].

Grossberg and Nayar [24], [25] define a non-parametric discrete imaging model that can represent any type of camera, central or non-central (*General Camera Model*). Differently from the usual parametric image models, this camera model consists in the individual mapping between pixels and rays in 3D space, Figure 1. To each pixel, a set of parameters called *raxel* is associated. The set of all *raxels* (representing all pixels) constitutes the complete *General Camera 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 Camera 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. It requires two world points for the same image pixel.

Sturm and Ramalingam at [26] and Ramalingam *et al.* at [27] 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 methods require 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

- P. Miraldo and H. Araujo are with the Institute for System and Robotics, Department of Electrical and Computer Engineering, University of Coimbra, 3030-290 Coimbra, Portugal.
E-mail: {miraldo, helder}@isr.uc.pt
- This work was supported by the Portuguese Foundation for Science and Technology (FCT) with grants PTDC/EIA-EIA/122454/2010 and PTDC/EIA-CCO/109120/2008. P. Miraldo also wants to think to the FCT for his doctoral degree grant with reference SFRH/BD/49054/2008

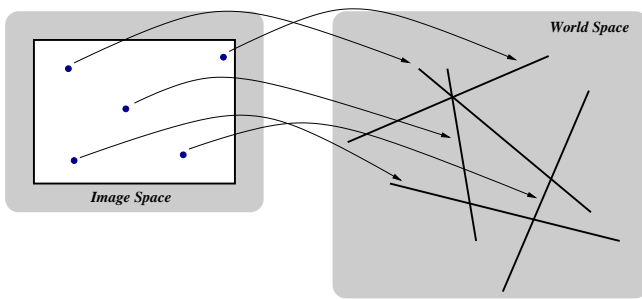


Fig. 1. A unified image model only assumes that an association between image pixels and rays in the world space exists.

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

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.

Most of the useful image models have, in general, a pixel-ray relationships that vary smoothly along the image. That is the case of the central perspective camera, non-central catadioptric systems with quadric mirrors, linear cameras, fisheye cameras and of image models that include refracting elements.

In this article we change the discrete model defined by Grossberg and Nayar by assuming that the $3D$ rays associated to the pixels vary smoothly throughout the image. The assumption of smoothness is used as a constraint in the definition of the image model and therefore in the calibration method. This paper extends the approach presented in [28] where the *Smooth Camera Model* was presented for the first time. This paper provides further details concerning the application of the *Smooth Camera Model*, and improves it by using normalized coordinates. An example and discussion of calibration with minimal data is presented. Additional experimental results with synthetic data are presented. The additional experimental results include a crossed-slits camera model, a "refraction camera" made up of a perspective camera looking through a volume of water and a non-smooth configuration made up of four perspective cameras looking at the same scene. Additional experimental results with real images are also included namely the case of a perspective camera looking at a scene through a water-filled glass tank.

1.1 Model and Approach

Since the estimation of the *raxels* for all the image pixels require a lengthy calibration procedure, other authors have also assumed that the relationship between neighboring pixels and neighboring $3D$ rays in the world vary smoothly.

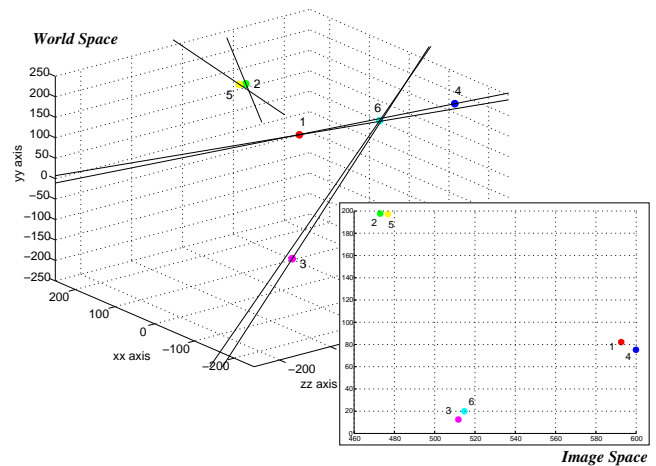


Fig. 2. In this figure we display the result of the application of the calibration method to the minimal case according to the theoretical constraints (see Section 4.2.2). In the minimal case there are six points in the the image and their correspondent points in the world.

In [25] and [26], [27], the authors have also used this constraint. The constraint is applied after the estimation of at least a subset of all possible image pixels. Sturm and Ramalingam used this constraint to interpolate between a subset of calibrated *raxels* corresponding to neighboring pixels. Grossberg and Nayar at [25] defined a *Caustic Roxel Model* that aimed at estimating a caustic surface from a set of *raxels*.

In that model the estimation of the caustic (and therefore the calibration of the imaging model) requires the estimation of the *raxels*. Such estimation needs that more than one $3D$ point be used for each image pixel.

In both previous cases, the use of constraints enforcing smoothness between neighboring pixels is used post calibration to interpolate between pixels that were not calibrated. In the proposed approach, the assumption that the relationship between pixels and rays varies smoothly is used directly in the model. For that purpose a vector-valued function that can represent any *Smooth Camera Model* is defined. A parametric representation for the *General Camera Model* is defined which results in:

- a decrease in the number of unknown parameters;
- the number of parameters and image resolution becoming independent.

Using this image model, a new calibration procedure can be developed, where:

- only one world point is needed for each image point, unlike previous methods, [24] and [26] that require at least two points in the world, for each image point;
- the calibration procedure has lower complexity since all the parameters are estimated in a single step.

As in [26], only geometric entities of *General Camera Model* are considered.

In Figure 2, an example of the advantages of the use of the proposed method in the calibration of smooth image models is shown. The minimal case (defined according to the theoretical conditions) is considered, where six matchings between world and image points are known (see the proof in Section 4.2.2). In this example all the coordinates of the image points are different. The estimated 3D lines pass through the corresponding 3D points using one-to-one mappings, which occurs as a result of applying the constraint that neighboring pixels must correspond to neighboring rays in the world (smoothness constraint).

The methods proposed can be considered a black box general *Smooth Camera Model* which is linear and leads to a simpler calibration method.

1.2 Outline of the Paper

In the next section, the mathematical notation that will be used in the article is presented. In Section 3 the proposed change to the *General Camera Models*, *Smooth Camera Model* is defined. In Section 4 the proposed method is described. Results are presented in Section 5 and discussed in Section 6.

2 NOTATION AND BACKGROUND

2.1 Notation

Matrices are represented as bold capital letters (eg. $\mathbf{A} \in \mathbb{R}^{n \times m}$, n rows and m columns). Vectors are represented as bold small letters (eg. $\mathbf{a} \in \mathbb{R}^n$, n elements). By default, a vector is considered a column. Small letters (eg. a) represent one dimensional elements. By default, the j th column vector of \mathbf{A} is specified as \mathbf{a}_j . The j th 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 (eg. A) indicate one dimensional constants.

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

Let $\mathbf{U} \in \mathbb{R}^{m \times n}$, $\mathbf{V} \in \mathbb{R}^{k \times l}$ and $\mathbf{C} \in \mathbb{R}^{m \times k}$ be known and $\mathbf{X} \in \mathbb{R}^{n \times l}$ unknown. Using *Kronecker product*, we can write

$$\mathbf{UXV}^T = \mathbf{C} \implies (\mathbf{V} \otimes \mathbf{U}) \text{vec}(\mathbf{X}) = \text{vec}(\mathbf{C}) \quad (1)$$

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 $\text{vec}(\mathbf{X})$ 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* [29]. 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 representation of the exterior product to four dimensional vectors $\mathbf{x} \wedge \mathbf{w}$. The result of this operation lies in a six dimensional vector space \mathbb{R}^6 , that can represent lines in 3D space.

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 on both points, as

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

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 (2) 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* which is the same as write $\langle \mathbf{d}, \mathbf{m} \rangle = 0$.

Plücker coordinates enable the computation of the incidence condition of lines and points in the world. Using the direction and moment vectors, a point $\mathbf{p} \in \mathcal{P}^3$ is incident on a line $\mathbf{l} \in \Lambda^2 \mathbb{R}^4$ if

$$\underbrace{\begin{pmatrix} [\mathbf{p}]_{\mathbf{x}} & -\mathbf{I} \\ \mathbf{0} & \mathbf{p}^T \end{pmatrix}}_{\mathbf{Q}(\mathbf{p})} \mathbb{R} = \mathbf{0} \quad (3)$$

where $\mathbf{p} \in \mathbb{R}^3$ (non-homogeneous representation), $[\mathbf{a}]_{\mathbf{x}}$ is a matrix that linearizes the exterior product as $[\mathbf{a}]_{\mathbf{x}} \mathbf{b} = \mathbf{a} \times \mathbf{b}$ 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 $\mathcal{X} \doteq \{\mathbf{x}_i\} \subset \mathbb{R}^D$ (with D a natural number) and $\mathcal{Y} \doteq \{y_i\}$, 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

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

Radial basis functions (RBF) [30], [31], [32], [33] can be used to solve this problem. For a set P of training points $\{\mathbf{x}_1, \dots, \mathbf{x}_P\}$, the RBF interpolant function has the form

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

where $\|\cdot\|$ is the 2-norm vector [34], $\phi : \mathbb{R}_+ \mapsto \mathbb{R}$ is the *radial basis function* and $\mathbf{a}_{\mathbf{x}} \in \mathbb{R}^D$. a_0 , $\mathbf{a}_{\mathbf{x}}$ 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* $\phi(r) = \exp(-\gamma^2 r^2)$, and *multi-quadrics* with $\phi(r) = (\gamma^2 + r^2)^{1/2}$ where γ is the shape parameter.

The interpolation is obtained by means of the estimation of the unknown parameters of the interpolant ($\mathbf{a} = (a_0, \mathbf{a}_x)$) and $\mathbf{w} = (w_1, \dots, w_P)$ of Equation (5)). The interpolating function $s(\mathbf{x})$ has $P + D + 1$ degrees of freedom and the data sets \mathcal{X} and \mathcal{Y} only yield P equations. For the estimation of the unknowns in Equation (5), additional constraints have to be used. Considering the function ϕ conditionally positive definite [31], the following equations are verified,

$$\sum_{i=1}^P w_i = 0 \quad \& \quad \sum_{i=1}^P w_i x_1^{(i)} = \dots = \sum_{i=1}^P w_i x_D^{(i)} = 0 \quad (6)$$

[28], [31] where $x_i^{(j)}$ is the i th element of the j th observation. The use of these constraints allows the estimation of all unknowns.

The estimation of the unknown parameters can be obtained using the following relation

$$\underbrace{\begin{pmatrix} \Phi & \mathbf{P}^T \\ \mathbf{P} & \mathbf{0} \end{pmatrix}}_{\mathbf{\Gamma}} \begin{pmatrix} \mathbf{w} \\ \mathbf{a} \end{pmatrix} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \quad (7)$$

where $\phi_{i,j} = \phi(\|\mathbf{x}_i - \mathbf{x}_j\|)$. Furthermore, considering $D = 2$, we can see that $\mathcal{C}(\mathbf{\Gamma}) = P + 3$ implies $P \geq 3$.

3 SMOOTH CAMERA MODEL

From the definition of *General Camera Model* Figure 1, introduced by Grossberg and Nayar [24], 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* and it can be used to represent central or non-central image model as well as smooth and non-smooth image models.

In that model, a complete *General Camera 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 (corresponding 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 computed. For an image with size $N \times M$, there are $7NM$ unknown parameters to be computed. We aim at deriving a model that requires a smaller number of parameters.

Since most of the imaging devices are characterized by having a smooth relationship between image pixels and lines in the world we use the assumption that the pixel-line mappings 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. We use this approach to represent the *Smooth Camera Model*. This assumption significantly decreases the number of model unknowns and also allows to filter out some error due to noise.

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 Camera Model* can only be defined when considering the mapping from image coordinates to $3D$ lines. 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 [10].

As previously described, a single mapping between points in the image to points in the world is insufficient to define a *raxel* and as a consequence it can not define a *General Camera Model*. The information of the direction of the line can not be recovered. However, if we assume that the relationship between image points and world lines varies smoothly, we are imposing new constraints to lines. Those constraints will implicitly allow the complete calibration of the *Smooth Camera Model* using only a single point in the world for a point in the image, as proved in the following sections. An example can be seen in Figure 2.

4 THE PROPOSED METHOD

In this section we describe the formulation of the proposed approach to represent the *Smooth Camera Model*, Section 4.1, and the calibration procedure, Section 4.2. In Section 4.3, we describe the data-set normalization.

4.1 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 the 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 to represent $3D$ lines, we use a vector made up by stacking the vectors of the direction and moment $\widehat{\mathbb{R}} = (\widehat{\mathbf{d}}, \widehat{\mathbf{m}})$. In other words, two independent vectors are estimated, up to the same scale factor.

Good estimates of $\widehat{\mathbf{d}}$ and $\widehat{\mathbf{m}}$ 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 [34], by finding the closest rotation matrix to $\begin{pmatrix} \widehat{\mathbf{d}}' & \widehat{\mathbf{m}}' & \widehat{\mathbf{d}}' \times \widehat{\mathbf{m}}' \end{pmatrix}$, where $\mathbf{x}' = \mathbf{x}/\|\mathbf{x}\|$, or by using the algorithm proposed by Bartoli and Sturm [35].

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

$$s(\mathbf{x}) = a_0 + \mathbf{a}_x^T \mathbf{x} + \sum_{i=1}^P w_i \phi(\|\mathbf{x} - \mathbf{c}_i\|) \quad (8)$$

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

$$\mathbf{s}(\mathbf{x}) = \begin{pmatrix} \phi(\mathbf{x}) & \mathbf{p}(\mathbf{x}) \end{pmatrix} \underbrace{\begin{pmatrix} \mathbf{w} \\ \mathbf{a} \end{pmatrix}}_{\mathbf{h}_{\mathbf{wa}}} \quad (9)$$

where $\phi(\mathbf{x})$ and $\mathbf{p}(\mathbf{x})$ are row vectors, with $\phi_i(\mathbf{x}) = \phi(\|\mathbf{x} - \mathbf{c}_i\|)$ and $\mathbf{p}(\mathbf{x}) = (1 \ x_1 \ x_2)$.

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

$$\mathbf{s}(\mathbf{x}) = \begin{pmatrix} \phi(\mathbf{x}) & \mathbf{p}(\mathbf{x}) \end{pmatrix} \underbrace{\begin{pmatrix} \mathbf{h}_{\mathbf{wa}}^{(1)} & \mathbf{h}_{\mathbf{wa}}^{(2)} & \dots & \mathbf{h}_{\mathbf{wa}}^{(6)} \end{pmatrix}}_{\mathbf{H}_{\mathbf{wa}}}. \quad (10)$$

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

For a set $\{\mathbf{c}_i\}$, a matrix $\mathbf{H}_{\mathbf{wa}}$ and a certain RBF, the estimates of the direction and moment vectors, as a function of the coordinates of the image points, are given by

$$\hat{\mathbf{I}}(\mathbf{x})^T \mathbb{R} = \left(\hat{\mathbf{d}}(\mathbf{x})^T \hat{\mathbf{m}}(\mathbf{x})^T \right) \mathbb{R} = \begin{pmatrix} \phi(\mathbf{x}) & \mathbf{p}(\mathbf{x}) \end{pmatrix} \mathbf{H}_{\mathbf{wa}}. \quad (11)$$

From the previous equation, it can be observed that for two different imaging systems using the same set of points $\{\mathbf{c}_i\}$, the estimation of $\hat{\mathbf{I}}$ for an image point \mathbf{x} only depends on matrix $\mathbf{H}_{\mathbf{wa}}$. Thus, we call $\mathbf{H}_{\mathbf{wa}}$ 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 $\{\mathbf{c}_i\}$, and that is why we call them *control points*.

Usually, in statistics, the set $\{\mathbf{c}_i\}$ 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 $\{\mathbf{c}_i\}$ as *control points*.

For a set P of *control points* defined *a priori* $\{\mathbf{c}_i\}$ and a *camera matrix* $\mathbf{H}_{\mathbf{wa}} \in \mathbb{R}^{(P+3) \times 6}$, we define the general *Smooth Camera Model* model by the vector-valued function $\mathbf{s} : \mathbb{R}^2 \mapsto \mathbb{R}^6$.

The *Smooth Camera Model* only depends on the unknown matrix $\mathbf{H}_{\mathbf{wa}}$, for a set of previously defined *control points*. Therefore, the complete calibration of the *Smooth Camera Model* can be obtained by estimating $6P + 18$ unknown parameters, that sets up the *camera matrix*.

4.2 Linear Point-based Calibration

One disadvantage of the use of the *General Camera Model* is the difficulty of its calibration. Grossberg and Nayar [24] and Sturm and Ramalingam [26] define two different methods for the calibration. However in both cases, the complete calibration is achieved when we have a *raxel* for any pixel of the image. Note that the estimation of a

single *raxel* requires at least two points in the world for the each image pixel. Since the estimation of the *raxels* for every pixel is a difficult task, interpolation methods were used to estimate all the *raxels* from a subset of calibrated *raxels* (already described in Section 1.1). The goal in the calibration of a *Smooth Camera Model* is to allow the calibration of the respective model without estimating each and all the *raxels* while, at same time, allowing the same resolution.

In the method described in this paper, the calibration procedure only requires a set of point correspondences $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, (where $\{\mathbf{p}_i\}$ is the set of world points and $\{\mathbf{x}_i\}$ are their corresponding points in the image plane). Point correspondences that satisfy

$$\mathbf{x}_i \neq \mathbf{x}_j, \quad \forall_{i \neq j}, \quad (12)$$

can be used. For each image pixel only the coordinates of one 3D point are necessary.

There are two sets of unknowns in the calibration procedure: the set of *control points* $\{\mathbf{c}_i\}$ and the elements of the *camera matrix* $\mathbf{H}_{\mathbf{wa}}$. *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 $\{\mathbf{c}_i\} \subset \{\mathbf{x}_i\}$.

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

4.2.1 Calibration Matrix M

World points incident on lines must verify Equation (3). From $\hat{\mathbf{I}}(\mathbf{x}) \mathbb{R} = \mathbf{s}(\mathbf{x})^T$ and Equation (3)

$$\hat{\mathbf{I}}(\mathbf{x})^T \mathbf{Q}(\mathbf{p})^T = \mathbf{s}(\mathbf{x}) \mathbf{Q}(\mathbf{p})^T = \mathbf{0}. \quad (13)$$

Replacing $\mathbf{s}(\mathbf{x})$ from the previous equation, using Equation (10)

$$\underbrace{\begin{pmatrix} \phi(\mathbf{x})^T & \mathbf{p}(\mathbf{x})^T \end{pmatrix}}_{\mathbf{r}(\mathbf{x})} \mathbf{H}_{\mathbf{wa}} \mathbf{Q}(\mathbf{p})^T = \mathbf{0} \quad (14)$$

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

The unknowns are the elements of matrix $\mathbf{H}_{\mathbf{wa}}$. Thus, using the *Kronecker product*, Section 2.1, which allows to rewrite (14) in order to isolate the unknown *camera matrix* as

$$(\mathbf{Q}(\mathbf{p}) \otimes \mathbf{r}(\mathbf{x})) \text{vec}(\mathbf{H}_{\mathbf{wa}}) = \mathbf{0} \quad (15)$$

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

For a set N of point correspondences $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ of the same imaging system, $(\mathbf{Q}(\mathbf{p}_i) \otimes \mathbf{r}(\mathbf{x}_i)) \text{vec}(\mathbf{H}_{\mathbf{wa}}) = \mathbf{0}$ and the calibration matrix can be built as

$$\mathbf{M} = \begin{pmatrix} \mathbf{Q}(\mathbf{p}_1) \otimes \mathbf{r}(\mathbf{x}_1) \\ \vdots \\ \mathbf{Q}(\mathbf{p}_N) \otimes \mathbf{r}(\mathbf{x}_N) \\ \mathbf{D} \end{pmatrix} \quad (16)$$

where $\mathbf{M} \in \mathbb{R}^{(4N+18) \times (6P+18)}$ and matrix $\mathbf{D} \in \mathbb{R}^{18 \times (6P+18)}$ is as [36].

The calibration procedure is reduced to the estimation of the unknown *camera matrix* \mathbf{H}_{wa} such that

$$\mathbf{M}\text{vec}(\mathbf{H}_{\text{wa}}) = \mathbf{0}, \quad \text{which means } \text{vec}(\mathbf{H}_{\text{wa}}) \in \text{null}(\mathbf{M}) \quad (17)$$

where $\text{null}(\cdot)$ indicates matrix *null-space*.

4.2.2 Computation of the Camera Matrix

From Equation (17), the estimate of the *camera matrix* must belong to the *null-space* of the calibration matrix \mathbf{M} . To ensure an unique solution, and since $\text{vec}(\mathbf{H}_{\text{wa}}) \in \mathbb{R}^{6P+18}$, the dimension of the *column space* of \mathbf{M} must be $\mathcal{C}(\mathbf{M}) = 6P+17$, which means that the dimension of the *null-space* of \mathbf{M} will be $\mathcal{N}(\mathbf{M}) = 1$.

From Equation (11), any solution for the *camera matrix* is defined up to a scale factor. Thus, assuming that $\mathcal{N}(\mathbf{M}) = 1$, any element of the one dimensional *null-space* of \mathbf{M} is a solution, except for the *trivial solution* $\text{vec}(\mathbf{H}_{\text{wa}}) = \mathbf{0}$.

To prove that the dimension of the *column space* of \mathbf{M} is $\mathcal{C}(\mathbf{M}) = 6P+17$, we decompose Equation (16) into rows, by means of the *Kronecker product*. Since the permutation of the rows in any matrix does not change the dimension of the *column space*, defining $N = 2P$, it is possible to rearrange matrix \mathbf{M} as a block of matrices similar to $\mathbf{\Gamma}$ (Equation (7)). It can be proved that if a specific set of constraints is met, then matrices $\mathbf{\Gamma}$ are full rank, $\mathcal{C}(\mathbf{D}) = 17$ and $\mathcal{C}(\mathbf{M}) = 6P+17$ —see next section and [36].

From this result, and since $\mathcal{C}(\mathbf{\Gamma}) = P+3$ implies $P \geq 3$, we can conclude that the minimal configuration to ensure a unique solution for the computation of the camera matrix corresponds to $N = 6$. An example of a calibration corresponding to the minimal case is shown in Figure 2.

4.2.3 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 $\mathcal{C}(\mathbf{M}) = 6P+17$. Two types of *radial basis functions*, that can be applied to get that *rank* are also discussed.

Let us assume that a set of point correspondences, $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ for $i = 1, \dots, 2P$ and $\mathbf{x}_i \neq \mathbf{x}_j$, $\forall i \neq j$ is available, and that it can be split into two sub-sets such that

$$\begin{aligned} \mathcal{K}^{(1)} &\doteq \{\mathbf{x}_i\} \quad \text{for } i = 1, \dots, P \\ \mathcal{K}^{(2)} &\doteq \{\mathbf{x}_i\} \quad \text{for } i = P+1, \dots, 2P. \end{aligned} \quad (18)$$

In [32] and [36] it is shown that only *control points* ($\{\mathbf{c}_i\}$, for $i = 1, \dots, P$) and *data points* ($\{\mathbf{x}_i\}$, for $i = 1, \dots, 2P$) that meet the condition

$$d < \epsilon q \quad (19)$$

can be considered. In Equation (19), $d = \min\{d_1, d_2\}$, $0 < \epsilon \leq 1$, $d_1 = \max\{\|\mathbf{x}_i - \mathbf{c}_i\|\}$, where the set $\{\mathbf{x}_i\}$

belongs to the set $\mathcal{K}^{(1)}$, $d_2 = \max\{\|\mathbf{x}_i - \mathbf{c}_i\|\}$, where the set $\{\mathbf{x}_i\}$ belongs to $\mathcal{K}^{(2)}$ and $2q = \min_{j \neq i} \{\|\mathbf{c}_i - \mathbf{c}_j\|\}$.

Note that (from [36]) if we consider $\{\mathbf{c}_i\} \doteq \mathcal{K}^{(1)}$ and if $d_2 < \epsilon q$ then we can also obtain $\mathcal{C}(\mathbf{M}) = 6P+17$. This solution also meets the constraint $\mathbf{x}_i \neq \mathbf{x}_j$, $\forall i \neq j$.

Quak *et al.* [32] proved that $\phi_1(r) = (\gamma_1^2 + r^2)^{1/2}$ and $\phi_2(r) = \exp(-\gamma_2 r^2)$ are good choices for *radial basis functions*, since that, by choosing an appropriate γ_1 and γ_2 , the negative effects of small values of q and ϵ respectively in Equation (19) can be reduced.

4.3 Data Normalization

One of the issues in the method proposed in [28] is related to the shape parameters of the RBF. The choice of this values depends on the distribution of set of the image points $\{\mathbf{x}_i\}$. To avoid this problem, a data normalization procedure for both image and world coordinates was developed. We consider non-isotropic normalization [37] for both image and world points, such that

- the centroid is at the origin;
- the principal moments are equal to unity.

If on one hand, normalization of the image space coordinates is easy to implement, on the other hand, normalization of the coordinates of the set of world points $\{\mathbf{p}_i\}$ will have additional implications. The normalization of the coordinates of the world points, $\{\mathbf{p}_i\}$, in the calibration procedure will imply a change on the parameters of the *Smooth Camera Model*. To account for this change, the inverse of this transformation has to be applied to the parameters of the *Smooth Camera Model* that were estimated.

4.3.1 Scaling the Coordinates of the Set of Image Points

For the normalization of the image space, an affine transformation

$$\mathbf{u}_i = a^{-1}(\mathbf{A}\mathbf{x}_i + \mathbf{a}) \quad (20)$$

is considered, where $\mathbf{A} \in \mathbb{R}^{2 \times 2}$ is upper triangular, and $\mathbf{a} \in \mathbb{R}^2$.

To obtain the normalization parameters $\{\mathbf{A}, \mathbf{a}, a\}$, an affine transformation based on *Choleski factorization* is used [34], [37]. Let us consider the *homogeneous* representation of the image coordinates \mathbf{x} as $\bar{\mathbf{x}}$. Since matrix $\sum_{i=1}^N \bar{\mathbf{x}}_i \bar{\mathbf{x}}_i^T$ is symmetric and positive definite, using the

using the *Choleski factorization* one can define $\sum_{i=1}^N \bar{\mathbf{x}}_i \bar{\mathbf{x}}_i^T = N\mathbf{K}_1\mathbf{K}_1^T$, where $\mathbf{K}_1 \in \mathbb{R}^{3 \times 3}$. Developing this equation, we get $\sum_{i=1}^N \mathbf{K}_1^{-1} \bar{\mathbf{x}}_i \bar{\mathbf{x}}_i^T \mathbf{K}_1^{-T} = N\mathbf{I}$. As a result, and using

$\bar{\mathbf{u}}_i = \mathbf{K}_1^{-1} \bar{\mathbf{x}}_i$, one obtains $\sum_{i=1}^N \bar{\mathbf{u}}_i \bar{\mathbf{u}}_i^T = N\mathbf{I}$ which means that the set $\{\mathbf{u}_i\}$ has its centroid at the origin of the coordinate systems and the two principal moments of the set of points are equal to one.

Since \mathbf{K}_1 is upper triangular, \mathbf{K}_1^{-1} is also upper triangular and we can define

$$\mathbf{K}_1^{-1} = \begin{pmatrix} \mathbf{A} & \mathbf{a} \\ \mathbf{0} & a \end{pmatrix}, \quad (21)$$

which includes the parameters of the affine transformation of Equation (20).

4.3.2 Scaling the Coordinates of the Set of World Points

To apply the normalization to the coordinates of the points in world space, a method similar to the method described in the previous section is applied. The coordinates of the set of world points $\{\mathbf{p}_i\}$ are changed into $\{\mathbf{q}_i\}$ such that

$$\mathbf{q}_i = b^{-1}(\mathbf{B}\mathbf{p}_i + \mathbf{b}) \quad \text{and} \quad \mathbf{K}_2^{-1} = \begin{pmatrix} \mathbf{B} & \mathbf{b} \\ \mathbf{0} & b \end{pmatrix} \quad (22)$$

where $\mathbf{K}_2^{-1} \in \mathbb{R}^{4 \times 4}$ is an upper triangular matrix (for points in 3D space). $\mathbf{B} \in \mathbb{R}^{3 \times 3}$ is upper triangular and $\mathbf{b} \in \mathbb{R}^3$.

Note that this normalization has consequences: it will change the coordinates of the line space. The output of function \mathbf{s} are the coordinates of the 3D lines. Therefore one has to account for this change of coordinates. The line coordinates which are the output of function \mathbf{s} have to be changed into the original coordinate frame.

Using the affine transformation defined at the Equation (22) and from the definition of *Plücker* coordinates, we define an affine transformation in line space as

$$\mathbf{l}^{(2)} \mathbb{R} = \underbrace{\begin{pmatrix} \mathbf{B} & \mathbf{0} \\ b^{-1} [\mathbf{b}]_{\times} \mathbf{B} & b^{-1} \det(\mathbf{B}) \mathbf{B}^{-T} \end{pmatrix}}_{\mathbf{E}} \mathbf{l}^{(1)} \quad (23)$$

(for the derivation of the matrix \mathbf{E} see the author web page). Note that \mathbf{B} is *non-singular*, which means that matrix \mathbf{E} is invertible and $\mathbf{l}^{(1)}$ can be estimated such that $\mathbf{l}^{(1)} \mathbb{R} = \mathbf{E}^{-1} \mathbf{l}^{(2)}$.

4.3.3 Camera Model and Calibration with Normalization

In this section we describe the algorithm for the calibration of the *Smooth Camera Model*, using normalized data.

Step 1: Compute matrices \mathbf{K}_1 and \mathbf{K}_2 using the method described in Sections 4.3.1 and 4.3.2, for the set of points $\{\mathbf{x}_i\}$ and $\{\mathbf{p}_i\}$ respectively. Compute the affine transformations parameters $\{\mathbf{A}, \mathbf{a}, a\}$ and $\{\mathbf{B}, \mathbf{b}, b\}$ from Equations (21) and (22) respectively. Compute the normalized image coordinates $\{\mathbf{u}_i\}$ and $\{\mathbf{q}_i\}$, using Equations (20) and (22) respectively. Using the affine transformation, get the matrix \mathbf{E} , using Equation (23).

Step 2: Compute the calibration matrix \mathbf{M} using Equation (16). Note that instead of $\{\mathbf{x}_i\}$ and $\{\mathbf{p}_i\}$, $\{\mathbf{u}_i\}$ and $\{\mathbf{q}_i\}$ should be used, respectively.

Step 3: Get $\text{vec}(\mathbf{H}_{\text{wa}})$ which minimizes $\text{Mvec}(\mathbf{H}_{\text{wa}})$, using the *Singular Value Decomposition* [3], and un-stack vector $\text{vec}(\mathbf{H}_{\text{wa}})$ to matrix \mathbf{H}_{wa} .

Step 4: The image model with normalization is defined by a function $\mathbf{s} : \mathbb{R}^2 \mapsto \mathbb{R}^6$ where $\widehat{\mathbb{R}} = \mathbf{s}(\mathbf{x})$, such that

$$\mathbf{s}(\mathbf{x}) = \mathbf{r}(a^{-1}(\mathbf{A}\mathbf{x} + \mathbf{a})) \mathbf{H}_{\text{wa}} \mathbf{E}^{-T} \quad (24)$$

where $\mathbf{r} : \mathbb{R}^2 \mapsto \mathbb{R}^{(p+3)}$ is defined in Equation (14). Note that the coordinate system where line coordinates $\widehat{\mathbb{R}}(\mathbf{x})$ are represented is the same coordinate system where the original set of point in the world $\{\mathbf{p}_i\}$ is represented.

All the results with real experiments were obtained by applying normalization to the data.

5 EXPERIMENTS

5.1 Results with Synthetic Data Sets

To evaluate the method described, synthetic data sets were generated. Using known camera models, 3D lines are generated, which are mapped into image points. The data set made up by pairings between 3D points and image points is obtained from the first data set (containing coordinates of 3D lines) by computing 3D points incident on the lines. The camera models used in the evaluation were: catadioptric system with quadric mirror, Figure 3(a); crossed-slits camera model, Figure 3(d); refraction camera model, made up of a camera looking through a volume of water contained between two parallel planes, Figure 3(g).

Since the mappings between points in the image and 3D lines are known, the calibration results can be evaluated by the estimates of the distance errors (distances between lines) in line space. Note that the camera model is represented by a mapping between points in the image and 3D lines.

5.1.1 Evaluation Results Using Smooth Camera Models

Synthetic data sets were used to evaluate the effect of the number of *control points* on the error defined as a distance between ground truth 3D lines and estimated 3D lines. As *radial basis functions multi-quadrics* and *exponential* were used.

The synthetic data sets were obtained for the following smooth camera models $\mathbf{f} : \mathcal{P}^2 \mapsto \mathcal{L}^3$: non-central catadioptric systems using quadric mirrors [9], crossed-slits camera model [16] and refraction-based camera model [19], [20].

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 analytic representation of the corresponding imaging model.

\mathbf{f} is used to generate a data set of $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, for $i = 1, \dots, N$ and P *control points* are selected from the set $\{\mathbf{x}_i\}$, with the conditions described in Section 4.2.3 being met.

Using \mathbf{f} , a ground truth data set is generated $\{\mathbf{y}_i \mapsto \mathbf{l}_i\}$, where \mathbf{y}_i are the ground truth image coordinates and \mathbf{l}_i are the ground truth 3D line coordinates.

The sets $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ and $\{\mathbf{c}_i\}$ are used to calibrate the camera model. Using the interpolant functions \mathbf{s} , the set $\{\mathbf{y}_i \mapsto \widehat{\mathbf{l}}_i\}$ is estimated, where $\widehat{\mathbf{l}}_i = \mathbf{s}(\mathbf{y}_i)$.

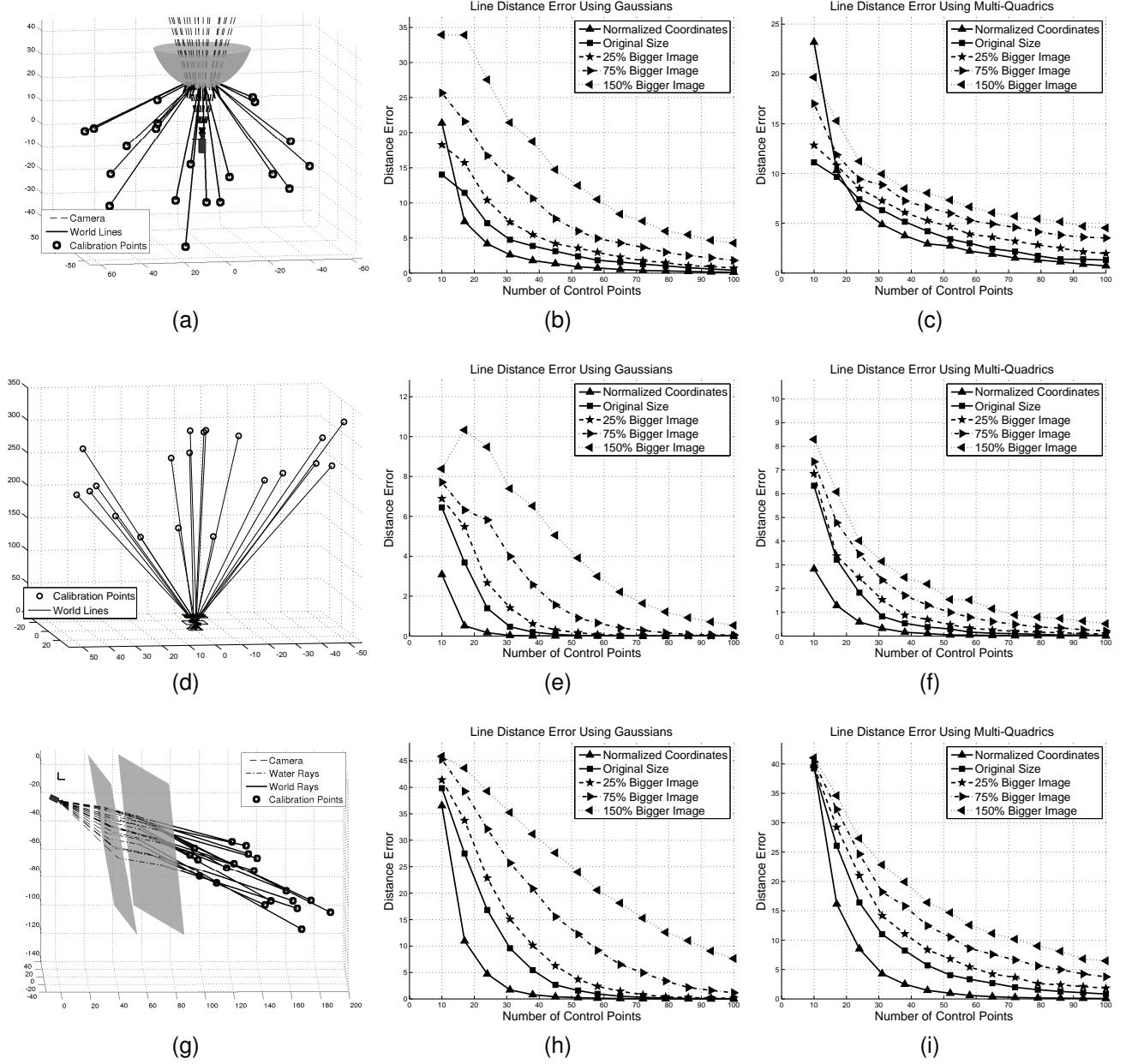


Fig. 3. Evaluation of the average line distance error, Equation (27), as a function of the number of *control points*, for non-central catadioptric systems with a quadric mirror, for a crossed-slits camera model and for a model including refraction (a camera looking through a volume of water contained between two parallel planes) (a), (d) and (g) respectively. For each camera model we evaluate the distance error defined in Equation (27), for *gaussian* and *multi-quadratics radial basis functions* and the errors are shown in Figures (b)-(c), (e)-(f) and (h)-(i) respectively.

To evaluate the calibration method the distances between $\hat{\mathbf{l}}_i$ and \mathbf{l}_i are used. To characterize the error the average of the distances $\epsilon_i = d(\mathbf{l}_i, \hat{\mathbf{l}}_i)$ is computed.

Let us consider two lines represented by *Plücker* coordinates \mathbf{g} and \mathbf{h} , which can be approximated via a local mapping into an Euclidean 4-space $\tilde{\mathbf{g}} = (g_1, g_2, g_3, g_4)$ and $\tilde{\mathbf{h}} = (h_1, h_2, h_3, h_4)$ [29] such that

$$\mathbf{g} = (g_3 - g_1, g_4 - g_2, 1, g_2, -g_1, g_1g_4 - g_2g_3) \quad (25)$$

$$\mathbf{h} = (h_3 - h_1, h_4 - h_2, 1, h_2, -h_1, h_1h_4 - h_2h_3). \quad (26)$$

The distances between the lines are estimated using

$$d(\tilde{\mathbf{g}}, \tilde{\mathbf{h}})^2 = \sum_{i=1}^4 (g_i - h_i)^2 + (g_1 - h_1)(g_3 - h_3) + (g_2 - h_2)(g_4 - h_4). \quad (27)$$

The results are evaluated by varying the number of *control points*, P , from 10 to 100. For each number of *control points*, the calibration is evaluated using the ground truth data set $\{\mathbf{y}_i \mapsto \mathbf{l}_i\}$ for $i = 1, \dots, 120$. For each set of *control points*, the calibration is repeated 150 times for

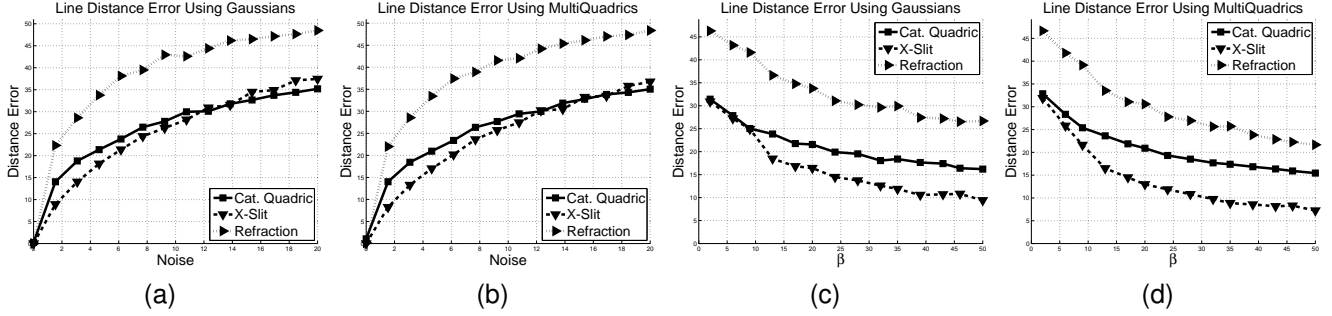


Fig. 4. In this figure we show results of tests performed with the three previously mentioned camera models, Figure 3, with noise added. We set the number of control points to 60 and vary the variance of the noise added to the coordinates of the 3D points used in the calibration procedure. The standard deviation of the noise is proportional to the smallest distance among all the world points of the calibration data set. (a) is the distance error in the line space for *gaussian* RBF and (b) for *multi-quadratics*. To eliminate some errors, we can use more points than the ones needed for the minimal solution. We consider the noise as 10% rate of the smallest distance between the world points and we vary the β , where $N = \beta P$. The results are show in Figures (c) and (d) for *gaussian* and *multi-quadratics* respectively.

different values of $\{x_i\}$, $\{y_i\}$ and $\{p_i\}$, $\{l_i\}$, respectively. The average of the distance errors is computed for each number of control points.

The calibration results are dependent on the value of the shape parameter in the case of either *multi-quadratic* or *gaussian radial basis functions*. By normalizing the data coordinates as described in Section 4.3.3, the value of the shape parameter is no longer dependent on the image size and also on the spatial distribution of the image points.

For a fixed value of the shape parameter, the error as a function of the spatial distribution of the set $\{x_i\}$ was evaluated. For that purpose and for each set $\{x_i\}$ and $\{y_i\}$, $\{x_i^{(j)}\}$ and $\{y_i^{(j)}\}$ were computed with

$$\mathbf{x}_i^{(j)} = \mathbf{x}_i \sigma_j \quad \forall i \quad (28)$$

for $j = 1, 2, 3$ with $\sigma_1 = 1.25$, $\sigma_2 = 1.75$ and $\sigma_3 = 2.5$. The results are shown in Figure 3. Note that by using normalized coordinates the error variation does not depend on the image size. Therefore the distance error can be represented by a single curve.

5.1.2 Results with Noise Added

The same evaluation procedure was repeated by adding Gaussian noise $\{\mu_i\}$ to the 3D coordinates of the points used in the calibration procedure $\{p_i\}$. For that purpose the standard deviation of the norm of the noise vector was defined as

$$\text{std}(\|\mu_i\|) = \alpha e \quad (29)$$

where $e = \min(\|p_i - p_j\|)$ for all $i \neq j$. Instead of using the set $\{x_i \mapsto p_i\}$ to calibrate, the set $\{x_i \mapsto \tilde{p}_i\}$, where $\tilde{p}_i = p_i + \mu_i$ for all i , was used.

The errors are evaluated in line space (Equation (27)) for the three previously mentioned camera models, with values for α from zero to 0.20. Calibration was performed using normalized coordinates (Section 4.3.3) with $P = 60$.

Results are shown in Figure 4(a) and 4(b) for *gaussian* and *multi-quadratics* RBF, respectively.

Note that in the previous experiments, $N = 2P$ was used, for P equal to the number of *control points*. On the other hand if $N = \beta P$ for $\beta > 2$, then matrix \mathbf{M} can become *full-column rank* and, as a consequence, there will be no exact solution for \mathbf{H}_{wa} . However, an estimate for $\text{vec}(\mathbf{H}_{wa})$ can still be obtained, in the least-squares sense [3].

The errors for β from 2 to 50 were also evaluated (measured as the distances between the 3D lines corresponding to the ground truth and the estimated 3D lines). For that purpose, $P = 60$ was used, and Gaussian noise with $\alpha = 0.10$, Equation (29) was added.

The results are shown in Figure 4(c) and 4(d) for *gaussian* and *multi-quadratics* RBF, respectively.

5.1.3 Results Using Non-Smooth Camera Models

The approach described in this paper is based on the assumption that the camera model is smooth. Therefore this method can not be applied to general camera models which line positions and orientations do not change smoothly. However, and even in some of those camera models, the approach can still be applied, provided that errors at the discontinuities can be accepted.

As an example the calibration procedure was applied to a non-central camera model obtained by combining images acquired by four different perspective cameras, with different poses (Figure 5(a)). The errors in 3D line space (Equation (27)), displayed as a function of the image coordinates are shown in Figure 5(b). These results were obtained for $P = 250$ and *multi-quadratics* RBF. The errors are small over most of the global image with the exception of the boundaries between the individual images.

To obtain results with smaller errors, the image points can be clustered based on the error distribution. To each

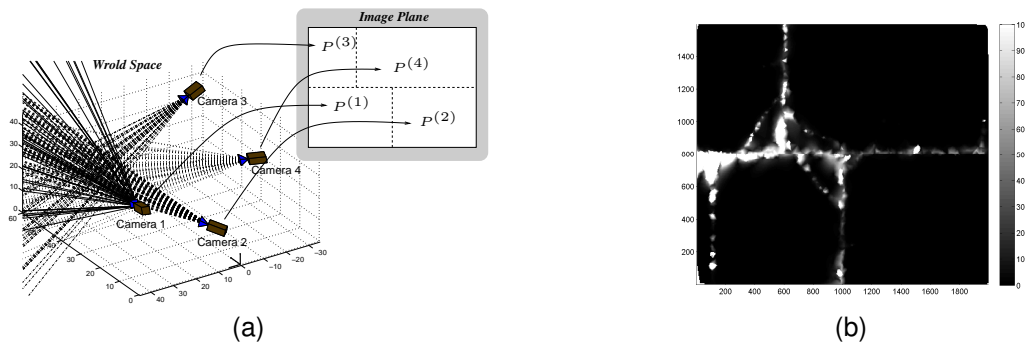


Fig. 5. This figure shows results for a non-smooth image model synthetically generated, and made up by combining four images obtained from four different perspective cameras as shown in (a). In figure (b) errors corresponding to the distances between the 3D lines are shown.

cluster, a specific *radial basis functions* can be assigned (based on the the distribution of control points). Each cluster can then be recalibrated individually.

5.1.4 Experimental Results Using 3D Data from a Single Surface

In this section we present the results of a new calibration experiment where the 3D points data used for the calibration belong to a single surface. Synthetic data was obtained from a non-central catadioptric system made up by a quadric mirror (Figure 3(a)). The catadioptric system was used to generate the data for the calibration, where it was assumed that the correspondences between the world points and their images were known. We consider world points on a spherical surface. Note that the 3D points can not belong to a single plane or to a single 3D line since in those cases the theoretical conditions described in [36] are not satisfied and the null space of matrix M has dimension greater than 1.

The results are shown in Figure 6. The image points used to evaluate the calibration (ray estimation) are not the image points used in the calibration process. We use the a non-minimal solution, with 40 control points and 360 data points, and the camera matrix is given in the least-squares sense [3]. From the experiments, lines are estimated with small error.

5.2 Results with Data sets of Real Images

For experiments with real data sets, three different types of imaging systems were calibrated, using the model described in this article and point-based calibration. A perspective camera and two different catadioptric systems were used, shown in Figures 7(a), 7(d) and 7(g) respectively. Note that the catadioptric camera model with two planar mirrors is a non-smooth camera model.

To acquire the point correspondences $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, which is the data necessary for the calibration method, a chess board was used. Infrared (IR) LEDs were attached to the chess board and their positions in the world were measured using an IR tracker [38]. This tracker has an accuracy of 0.1[mm] and a resolution of 0.01[mm].

Each corner of the chess board image \mathbf{x}_i is associated to a position in the world \mathbf{p}_i , which is given by its corresponding position in the 3D chess board. The set of associations $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, for $i = 1, \dots, 3840$, was used in the calibration procedure.

The set of image points $\{\mathbf{x}_i\}$ are the yellow points in Figures 7(b), 7(e) and 7(h). The set of world points $\{\mathbf{p}_i\}$ are the yellow points in Figures 7(c), 7(f) and 7(i).

Control points $\{\mathbf{c}_i\}$ are chosen as a subset of $\{\mathbf{x}_i\}$. They are shown as green points, in Figures 7(b), 7(e) and 7(h).

Using $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$, the interpolant function s is estimated. Since in the real experiments we have $N > 2P$, the solution will be obtained solving an over-determined system of equations. The *camera matrix* is estimated as least-squares solution for the homogeneous equations [3].

To evaluate the calibration, a different set of 3D coordinates, corresponding to 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 $\{\mathbf{y}_i \mapsto \mathbf{w}_i\}$, for $i = 1, \dots, 700$, 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 geometric distance ϵ_i from the world point \mathbf{w}_i to the line estimated $\hat{\mathbf{l}}_i$, where $\hat{\mathbf{l}}_i = s(\mathbf{y}_i)$. A subset of $\{\mathbf{y}_i\}$ and the corresponding lines $\{\hat{\mathbf{l}}_i\}$ are shown as red squares, in Figures 7(b), 7(e) and 7(h), and as red lines, in Figure 7(c), 7(f) and 7(i), respectively. We derived the geometric distance between a line (in *Plücker* coordinates) and a point in the world as $\epsilon_i = \|\epsilon_i\|$, where

$$\epsilon_i = ([\mathbf{w}_i]_x \quad -\mathbf{I}) \bar{\mathbf{l}}_i \quad (30)$$

and $\bar{\mathbf{l}}_i = \hat{\mathbf{l}}_i / \|\hat{\mathbf{d}}_i\|$. For more informatin, see the author web page.

The number of *control points* used in the calibration were 30, 35 and 40, for the imaging systems of Figures 7(a), 7(d) and 7(g) respectively. The results are shown in Table 1.

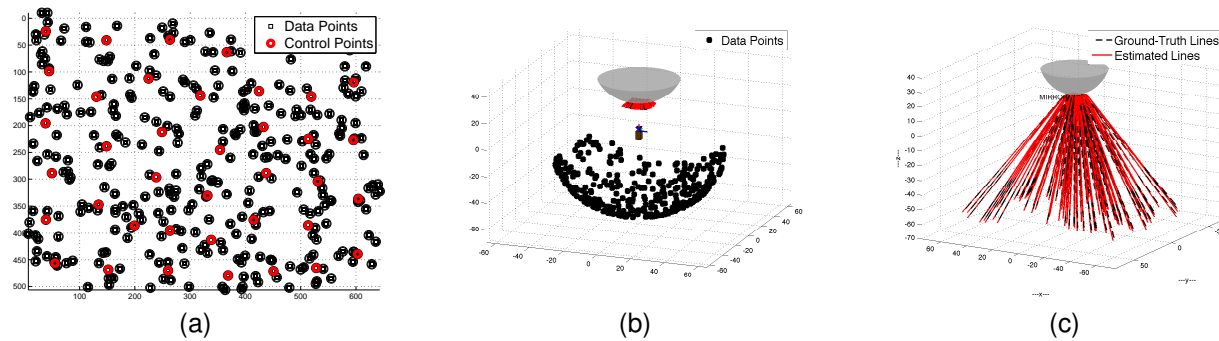


Fig. 6. Figure (a) represents the image points (black squares) and control points (red circles) used in the calibration process. Their correspondent world points are shown in Figure (b) (black points). We consider 3D points on a spherical surface. Figure (c) shows the results, where the ground–truth lines are shown as black lines and the estimated lines in red. Points in the image that were used to generate the estimated lines Figure (c) are different from the image points used in the calibration, (Figure (a)).

5.3 Using a Calibrated Perspective Camera to Acquire a Data-set with Real Data

To further evaluate the calibration method, an additional data-set of real data was acquired.

Consider an object or scene whose images are acquired by two perspective cameras. One of the cameras is looking directly at the object/scene, whereas the second camera acquires images of the object by viewing it through a glass tank filled with water (Figure 8(a)). The goal is to calibrate the second imaging system, i.e., including the effect of viewing the object through the glass tank filled with water. An example of the image acquired by this system is shown in Figure 8(b).

A chess board is used to calibrate the system. Several images of the chess board were acquired, with the chess board in different positions and orientations. The direct images of the chess board, i.e., acquired by the camera that views the scene directly (not through the glass tank) were used to estimate the coordinates of the planes in the world coordinate system. For that purpose the Bouguet calibration toolbox was used [39].

Eighteen different images of the chess board in different positions and orientations were used. In each chess board image there were of 160 points whose coordinates could be used (these points are located at the intersection of the edges). The points are displayed in Figure 8(c) as yellow and blue dots. The coordinates of the points on the chess board are used as world points forming the set $\{\mathbf{p}_i^{(j)}\}$ where $\mathbf{p}_i^{(j)}$ are the coordinates of the i 'th point of the j 'th chess position in the world.

Since these points are also visible in the image ac-

quired through the water-filled glass tank (i.e. including the effects of refraction), the images of the points on of each chess-board images $\{\mathbf{x}_i^{(j)}\}$ (where $\mathbf{x}_i^{(j)}$ is the i 'th point coordinate of the j 'th image of the j 'th chess board position) are associated with the corresponding world coordinates $\{\mathbf{p}_i^{(j)}\}$, forming the set of mappings $\{\mathbf{x}_i^{(j)} \mapsto \mathbf{p}_i^{(j)}\}$.

To test the calibration method the following procedure was followed: for each specific chess board position and orientation j , all the data from all other chess boards at different positions and orientations was used as calibration data set. The coordinates of the points of the j 'th chess board position were used as ground truth. Therefore the coordinates of the points of the j 'th chess board position were used as ground truth data set $\{\mathbf{y}_i \mapsto \mathbf{w}_i\}$ (160 mappings between image and world points) and the coordinates of the other points of the other chess board positions were used as calibration data-set $\{\mathbf{x}_i \mapsto \mathbf{p}_i\}$ (2720 mappings between image and world points). For each \mathbf{y}_i , the distances between the estimated lines $\hat{\mathbf{l}}_i$ and the corresponding points in the world \mathbf{w}_i are estimated, using Equation (30).

This procedure is repeated for all the j 'th chess board positions and the means and standard deviation of the distance errors are computed, for both *gaussian* and *multi-quadrics* RBFs. The number of *control points* used was 10. The corresponding results were 0.111 ± 0.075 [cm] for *multi-quadrics* and 0.313 ± 0.545 [cm] for *gaussian* RBF, respectively.

An example of the estimated lines, estimated using the method described (for a specific plane position) is shown in Figure 8(c), as red lines.

5.4 Removing Distortion: Results for a Spherical Catadioptric System

In general, undistorted views cannot be computed from images acquired by a general non–central image models (without any additional constraint). As mentioned by

TABLE 1

Results for the experiments, using the IR tracker [38].

RBF	<i>Gaussian</i> [cm]	<i>Multi-quadrics</i> [cm]
Projective	0.93058 ± 1.9783	0.54711 ± 0.43567
Sphere mirror	2.8743 ± 2.1074	1.8863 ± 1.2379
Plane mirrors	0.91795 ± 1.0608	0.99492 ± 0.78778

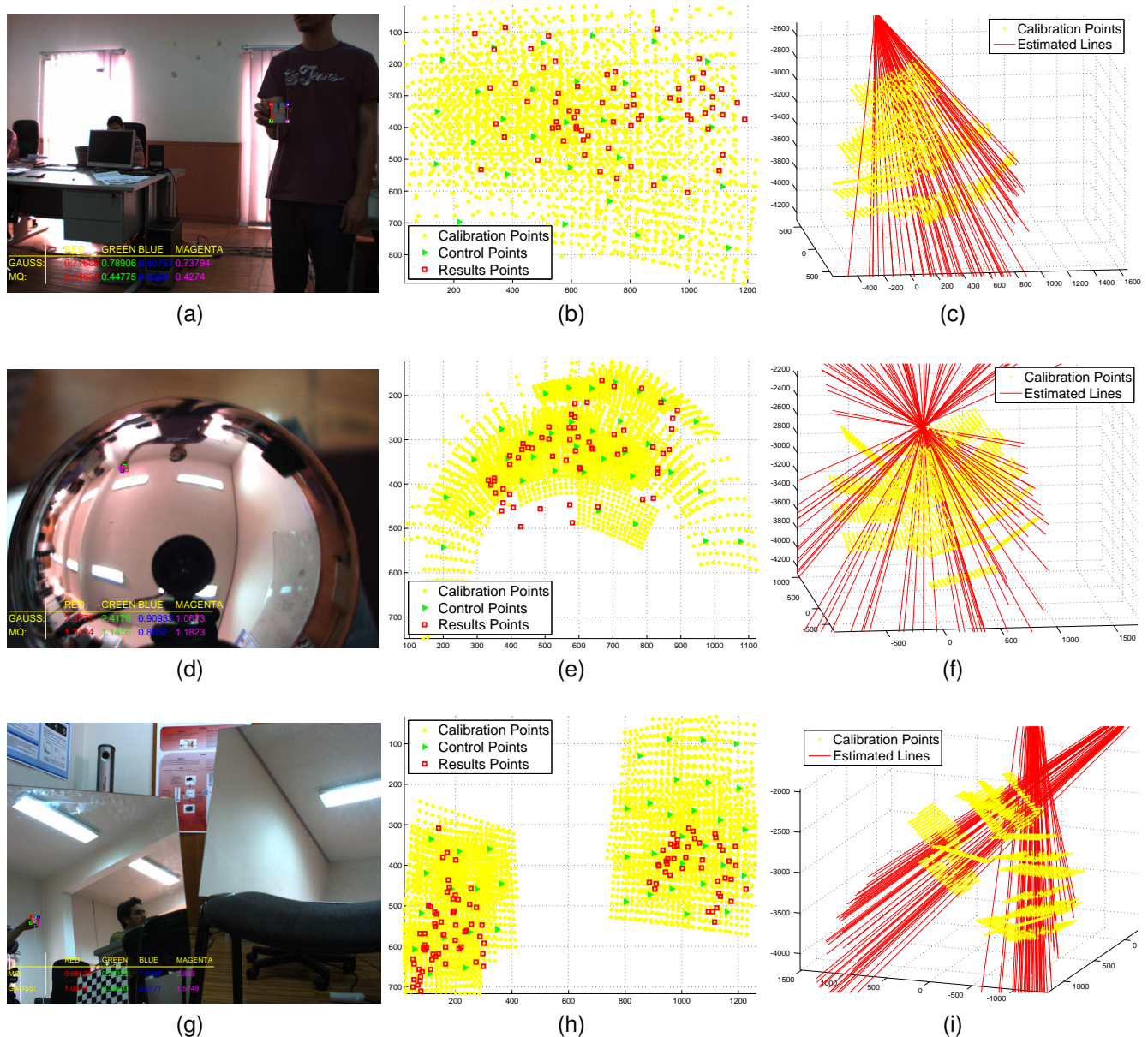


Fig. 7. Results with real images, obtained using the method described for three types of camera models: perspective camera (a), catadioptric system with spherical mirror (d) and catadioptric system with two planar mirrors (g). Yellow points in (b), (e) and (h) are the set of image coordinates $\{x_i\}$ and yellow points in (c), (f) and (i) are the corresponding set of world points $\{p_i\}$, used in the calibration method. Red squares in the 2D plot and the red rays in the 3D plot are a subset of $\{y_i \mapsto \hat{l}_i\}$ obtained with *multi-quadrics* RBF. Green points in (b), (e) and (h), are the corresponding set of *control points* used in the calibration.

Swaminathan *et al.* at [40], obtaining undistorted images from general non-central systems requires the knowledge of the scene structure.

Using the approach proposed by Swaminathan *et al.* (and the knowledge of the scene structure) we performed a new experiment to estimate undistorted images from the non-central catadioptric camera using a spherical mirror.

We used the same calibration process described in Section 5.3. We know position of the seventeen chess boards

and use sixteen of these chessboard positions to calibrate the camera model. We then obtained the undistorted image of the 17th plane. Using the knowledge of the position of the plane not used in the calibration, and its image, the undistorted image was computed.

Using a virtual perspective camera (located at $(0, 0, 0)$), we generated the undistorted image. The results are presented in Figure 9. As it can be seen from Figure 9(a) and Figure 9(b) the curved lines in the distorted image are mapped into straight line in the undistorted image.

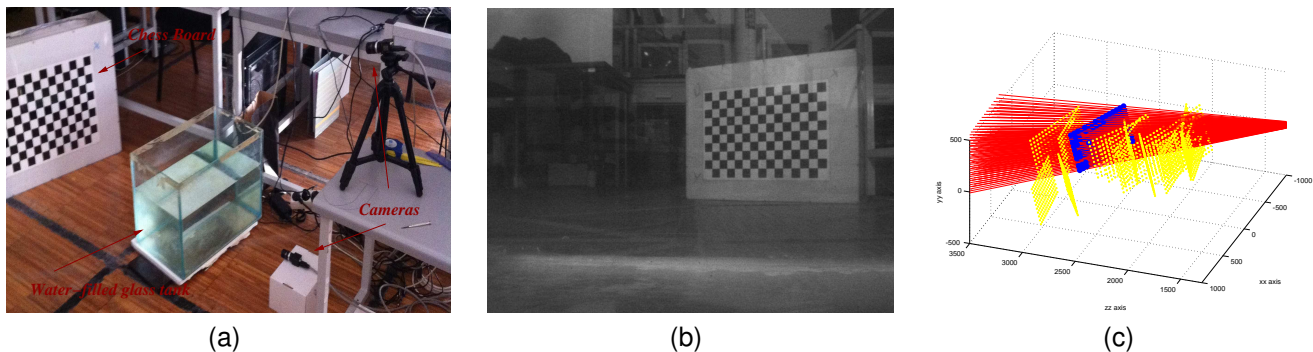


Fig. 8. In Figure (a), we display the set-up used to acquire the images of the data set. The goal is the calibration of an imaging system that views a scene through a water-filled glass tank. An example of an image acquired by this imaging system is shown in Figure (b). A planar chess board was used. The data set is made up by the coordinates of the images of the points corresponding to the intersections of the chess board edges (seen through the tank). An example of the results is shown in Figure (c). In yellow we show all the $3D$ points used in the calibration procedure. In blue we show the $3D$ points on the which the estimated lines must be incident. The estimated lines are shown in red. The units in Figure (c) are millimeters.

6 CONCLUSIONS

6.1 Discussion

The experiments and results with synthetic data were specially relevant for the goals of this paper. They were particularly important to evaluate the effect of the number of control points as well as to evaluate the sensitivity of the method to the noise, by means of a fully specified data set.

Considering the experimental results obtained without noise Figure 3, those obtained using *gaussian* RBF are slightly better than those obtained using *multi-quadrics*. However, *multi-quadrics* RBF tend to be less sensitive to the image scale variation. The experimental results also show that the inclusion of the coordinate normalization procedure (Section 4.3.3) tends to consistently yield better results, as expected. When noise is added to the synthetic data, and as expected, the use the least-squares estimates yields better results than the ones obtained applying the theoretical condition $N = 2P$.

In some cases the calibration is affected with errors

that go up to 60 units. These are extreme cases and we discuss the reasons why they occur. In one case (Figure 3) these errors correspond to results obtained with a very small number of control points 10. Note that this number of control points is small for the specific camera model (for other systems it might be adequate). Therefore in that case the reason for the error value is the small number of control points. In the case of Figure 4 these values are obtained for levels of noise with standard deviations which go up to 20 units. These are presented to show the performance of the model in very special (and unrealistic) conditions.

Experiments with synthetic data also show that this method can be applied to the calibration of non-smooth camera models if the discontinuities on the data are limited.

To conclude the synthetic experiments, we perform an experiment where (with the appropriate choices for the interpolant parameters) we can calibrate a camera model using world points in a single surface, Figure 6. These experiments show that the method can be used without explicitly accounting for the collinearity constraints.

Experimental results with real data-sets were also performed to evaluate the behavior of the method under real conditions. The results shown in Figures 7 and 8 show the expected behavior of the $3D$ line estimates obtained under the assumption of a *Smooth Camera Model*.

In addition, we also show distortion correction results for a non-central catadioptric camera, using the proposed camera model.

6.2 Closure

The proposed *Smooth Camera Model* and calibration procedure can be used to calibrate complex smooth camera models, namely in the case of cameras for which no analytical model exists. Despite the fact that this model

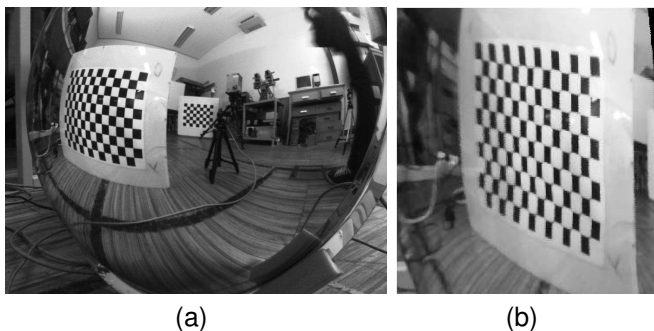


Fig. 9. In Figure (a) we show the original image obtained from a catadioptric camera system, made up by a spherical mirror. In Figure (b), we show the undistorted image.

is for smooth imaging systems, it can also be used also in special cases of non-smooth imaging systems.

This approach can model a camera with significantly less parameters than the discrete *General Camera 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 *General Camera Model*, this approach only requires $6(P + 3)$ for P control points.

The calibration procedure only requires the 3D coordinates of a single 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 are implicitly generalized for all image pixels, which constitutes an important advantage of this method.

REFERENCES

- [1] R. Y. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," *In CVPR*, 1986.
- [2] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2000.
- [3] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision (2nd ed.)*. The Edinburgh Building, Cambridge CB2 2RU, UK: Cambridge University Press, 2000.
- [4] S. Baker and S. K. Nayar, "A Theory of Single-Viewpoint Catadioptric Image Formation," *International Journal of Computer Vision*, 1999.
- [5] S. Gasparini, P. Sturm, and J. P. Barreto, "Plane-Based Calibration of Central Catadioptric Cameras," *In ICCV*, 2009.
- [6] R. Hartley and S. B. Kang, "Parameter-free Radial Distortion Correction with Centre of Distortion Estimation," *In ICCV*, 2005.
- [7] D. Nistér, H. Stewénius, and E. Grossmann, "Non-Parametric Self-Calibration," *In ICCV*, 2005.
- [8] S. Ramalingam, P. Sturm, and Lodha Suresh K., "Generic self-calibration of central cameras," *Computer Vision and Image Understanding*, 2010.
- [9] N. Goncalves, "Noncentral Catadioptric Systems with Quadric Mirrors, Geometry and Calibration," Ph.D. dissertation, University of Coimbra, October 2008.
- [10] A. Agrawal, Y. Taguchi, and S. Ramalingam, "Analytical Forward Projection for Axial Non-Central Dioptric & Catadioptric Cameras," *In ECCV*, 2010.
- [11] R. Swaminathan, M. D. Grossberg, and S. K. Nayar, "Non-Single Viewpoint Catadioptric Cameras: Geometry and Analysis," *International Journal of Computer Vision*, 2006.
- [12] B. Micusik and T. Pajdla, "Autocalibration & 3D Reconstruction with Non-Central Catadioptric Cameras," *In CVPR*, 2004.
- [13] J. Ponce, "What is a Camera?" *In CVPR*, 2009.
- [14] R. Gupta and R. I. Hartley, "Linear Pushbroom Cameras," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1997.
- [15] J. Yu and L. McMillan, "General Linear Cameras," *In ECCV*, 2004.
- [16] A. Zomet, D. Feldman, S. Peleg, and D. Weinshall, "Mosaicing New Views: The Crossed-Slits Projection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2003.
- [17] S. Hu and X. Feng, "Calibration of Panorama Based on Linear CCD and Fisheye Lens," *In CISP*, 2009.
- [18] J. Kannala and S. S. Brandt, "A Generic Camera Model and Calibration Method for Conventional, Wide-Angle, and Fish-Eye Lenses," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2006.
- [19] T. Treibitz, Y. Y. Schechner, and H. Singh, "Flat Refractive Geometry," *In CVPR*, 2008.
- [20] K. N. Kutulakos and E. Steger, "A Theory of Refractive and Specular 3D Shape by Light-Path Triangulation," *International Journal of Computer Vision*, 2008.
- [21] P. Sturm, S. Ramalingam, J.-P. Tardif, S. Gasparini, and J. Barreto, "Camera Models and Fundamental Concepts Used in Geometric Computer Vision," *Foundations and Trends in Computer Graphics and Vision*, 2011.
- [22] M. Menem and T. Pajdla, "Constraints on perspective images and circular panoramas," *In BMVC*, 2004.
- [23] S. Thirthala and M. Pollefeys, "Radial Multi-focal Tensors – Applications to Omnidirectional Camera Calibration," *International Journal of Computer Vision*, 2012.
- [24] M. D. Grossberg and S. K. Nayar, "A General Imaging Model and a Method for Finding its Parameters," *In ICCV*, 2001.
- [25] —, "The Raxel Imaging Model and Ray-Based Calibration," *International Journal of Computer Vision*, 2005.
- [26] P. Sturm and S. Ramalingam, "A Generic Concept for Camera Calibration," *In ECCV*, 2004.
- [27] S. Ramalingam, P. Sturm, and S. K. Lodha, "Towards Complete Generic Camera Calibration," *In CVPR*, 2005.
- [28] P. Miraldo, H. Araujo, and J. Queiró, "Point-based Calibration Using a Parametric Representation of the General Imaging Model," *In ICCV*, 2011.
- [29] H. Pottmann and J. Wallner, *Computational Line Geometry*. Berlin: Springer-Verlag, 2001.
- [30] M. Buhmann, *Radial Basis Functions: Theory and Implementations*. The Edinburgh Building, Cambridge CB2 8RU, UK: Cambridge University Press, 2003.
- [31] H. Wendland, *Scattered Data Approximation*. The Edinburgh Building, Cambridge CB2 8RU, UK: Cambridge University Press, 2005.
- [32] E. Quak, N. Sivakumar, and J. D. Ward, "Least Squares Approximation by Radial Functions," *SIAM Journal on Mathematical Analysis*, 1993.
- [33] N. Sivakumar and J. D. Ward, "On the least squares fit by radial functions to multidimensional scattered data," *Numerische Mathematik*, 1993.
- [34] G. H. Golub and C. F. Van Loan, *Matrix Computations (3rd ed.)*. Baltimore, MD, USA: Johns Hopkins University Press, 1996.
- [35] A. Bartoli and P. Sturm, "Structure-From-Motion Using Lines: Representation, Triangulation and Bundle Adjustment," *Computer Vision and Image Understanding*, 2005.
- [36] P. Miraldo, H. Araujo, and J. Queiró, "Unique Solution for the Estimation of the Plücker Coordinates Using Radial Basis Functions," University of Coimbra, Tech. Rep., (Supplement Material), 2011.
- [37] R. Hartley, "In Defense of the Eight-Point Algorithm," *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1997.
- [38] Northern Digital Incorporated, "Optotrak Certus Motion Capture System," Northern Digital Incorporated, 2009.
- [39] J.-Y. Bouguet. (2010, July (last update)) Camera Calibration Toolbox for Matlab. [Online].
- [40] R. Swaminathan, M. D. Grossberg, and S. K. Nayar, "A Perspective on Distortions," *In CVPR*, 2003.



Pedro Miraldo received the Master degree in Electrical and Computer Engineering from the University of Coimbra in 2008. Currently he is a researcher and PhD student at the Institute for Systems and Robotics-University of Coimbra, and his work focusses on general camera models.



Helder Araujo is a Professor in the Department of Electrical and Computer Engineering of the University of Coimbra, Portugal and also an Invited Professor at Blaise Pascal University, Clermont-Ferrand, France. He is a researcher at the Institute for Systems and Robotics-Coimbra. His research interests include computer and robot vision, robot navigation and localization, and sensor modeling for robot navigation and localization.