HAL
open science

Calibration of Central Catadioptric Cameras Using a DLT-Like Approach

Luis Puig, Yalin Bastanlar, Peter Sturm, Josechu Guerrero, Joao Barreto

To cite this version:

Luis Puig, Yalin Bastanlar, Peter Sturm, Josechu Guerrero, Joao Barreto. Calibration of Central Catadioptric Cameras Using a DLT-Like Approach. International Journal of Computer Vision, 2011, 93 (1), pp.101-114. 10.1007/s11263-010-0411-1 . inria-00590268

HAL Id: inria-00590268

https://inria.hal.science/inria-00590268

Submitted on 3 May 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Noname manuscript No.
(will be inserted by the editor)

Calibration of Central Catadioptric Cameras Using a DLT-Like Approach

Luis Puig · Yalin Bastanlar · Peter Sturm · J.J. Guerrero · João Barreto

the date of receipt and acceptance should be inserted later

Abstract In this study, we present a calibration technique that is valid for all single-viewpoint catadioptric cameras. We are able to represent the projection of 3D points on a catadioptric image linearly with a 6×10 projection matrix, which uses *lifted coordinates* for image and 3D points. This projection matrix can be computed from 3D-2D correspondences (minimum 20 points distributed in three different planes). We show how to decompose it to obtain intrinsic and extrinsic parameters. Moreover, we use this parameter estimation followed by a non-linear optimization to calibrate various types of cameras. Our results are based on the sphere camera model which considers that every central catadioptric system can be modeled using two projections, one from 3D points to a unitary sphere and then a perspective projection from the sphere to the image plane. We test our method both with simulations and real images, and we analyze the results performing a 3D reconstruction from two omnidirectional images.

Luis Puig
Departamento de Informática e Ingeniería de Sistemas (DIIS) and Instituto de Investigación en Ingeniería de Aragón (I3A) Universidad de Zaragoza, Zaragoza, Spain
E-mail: lpuig@unizar.es

Yalin Bastanlar
Informatics Institute, Middle East Technical University, Ankara, Turkey
E-mail: yalinb@ii.metu.edu.tr

J.J. Guerrero
Departamento de Informática e Ingeniería de Sistemas (DIIS) and Instituto de Investigación en Ingeniería de Aragón (I3A) Universidad de Zaragoza, Zaragoza, Spain
E-mail: jguerrer@unizar.es

Peter Sturm
INRIA Rhône-Alpes and Laboratoire Jean Kuntzmann, Grenoble, France
E-mail: Peter.Sturm@inrialpes.fr

João Barreto
ISR/DEEC, University of Coimbra, Coimbra, Portugal
E-mail: jpbar@deec.uc.pt

1 Introduction

Since their introduction to the computer vision community, catadioptric omnidirectional cameras have been utilized in many application areas such as surveillance [1], tele-presence [2], robot navigation [3] and 3D reconstruction [4]. Omnidirectional cameras being single-viewpoint are searched, since it is an important property. If single-viewpoint cameras are used, directions of the light rays coming into the camera can easily be calculated and combined in a multiview geometric framework [5]. Catadioptric systems, combinations of camera lenses and mirrors were extensively studied by Baker and Nayar [6]. They showed which of these systems are able to provide the single-viewpoint property, i.e., if the mirror has a focal point which can behave like an effective pinhole. Among those systems the most useful ones are the para-catadioptric and the hyper-catadioptric models, using a mirror of parabolic/hyperbolic shape, coupled with an orthographic/perspective camera. Swaminathan et al. [7] conducted a detailed study on the geometry of non-single-viewpoint systems. There also exist studies for approximating a viewpoint in non-single-viewpoint systems as Derrien and Konolige proposed for spherical mirrors [8].

Camera calibration is essential when we want to extract metric information from images. It establishes a relationship between the 3D rays and their corresponding pixels in the image. This relationship makes possible to measure distances in a real world from their projections on the images [9]. Camera calibration is ba-

sically composed of two steps. The first step consists of modeling the physical and optical behavior of the sensor through a geometric-mathematical model. There exist several approaches that propose different models to deal with central catadioptric systems [10–14]. The second step consists of estimating the parameters that compose this model using direct or iterative methods. These parameters are of two types, intrinsic and extrinsic. The intrinsic parameters basically consider how the light is projected through the mirror and the lens onto the image plane of the sensor. The extrinsic parameters describe the position and orientation of the catadioptric system with respect to a world coordinate system.

Several methods have been proposed for calibration of catadioptric systems. Some of them consider estimating the parameters of the parabolic [15,10], hyperbolic [16] and conical [17] mirrors together with the camera parameters. Calibration of outgoing rays based on a radial distortion model is another approach. Kannala and Brandt [18] used this approach to calibrate fisheye cameras. Scaramuzza et al. [12] and Tardif et al. [19] extended the approach to include central catadioptric cameras as well. Mei and Rives [20], on the other hand, developed another Matlab calibration toolbox that estimates the parameters of the sphere camera model. Parameter initialization is done by user input, namely, the location of the principal point and depiction of a real world straight line in the omnidirectional image (for focal length estimation).

Svoboda and Pajdla [21] derived epipolar geometry constraints for all types of central catadioptric cameras. Geyer and Daniilidis have shown the existence of a fundamental matrix for para-catadioptric cameras [22,23]. This has been extended by Sturm towards fundamental matrices and trifocal tensors for mixtures of para-catadioptric and perspective images [24]. Barreto showed that the framework can also be extended to cameras with lens distortion due to the similarities between the para-catadioptric and division models [25, 26]. Recently, Sturm and Barreto [27] extended these relations to the general catadioptric camera model, which is valid for all central catadioptric cameras. They showed that the projection of a 3D point can be modeled using a projection matrix of size 6×10. They also show the existence of a general fundamental matrix of size 15×15 and plane homographies, again of size 15×15. They used the sphere camera model [28] and so-called *lifted coordinates*.

This paper is an extended version of our previous work [29] where the calibration theory of central cameras proposed by Sturm and Barreto [27] is put into practice. We compute the generic projection matrix, P_{cata}, with 3D-2D correspondences, using a straightforward DLT-like (Direct Linear Transform [30]) approach, i.e. by solving a linear equation system. Then, we decompose P_{cata} to estimate intrinsic and extrinsic parameters. Having these estimates as initial values of system parameters, we optimize the parameters based on minimizing the reprojection error. A software version of our method is available at the author's Web page[1]. When compared to the technique of Mei and Rives [20] our approach has the advantages of not requiring input for parameter initialization and being able to calibrate perspective cameras as well. On the other hand, our algorithm currently needs a 3D calibration object.

In the next section, we introduce the sphere camera model and the proposed linear representation using lifted coordinates. In Section 3, we show how to compute and decompose the generic projection matrix into the intrinsic and extrinsic camera parameters. In Section 4 we present an analysis on the spatial distribution of 3D points required for an unambiguous estimation of the generic projection matrix. We also show the relation between the parameters of the real catadioptric systems and the parameters of the sphere camera model. In Sections 5 and 6, we present the results of experiments for the mentioned calibration approach with simulated and real images, respectively. Conclusions are stated in Section 7.

2 Background

Notations. We do not distinguish between a projective transformation and the matrix representing it. Matrices are represented by symbols in sans serif font, e.g. M and vectors by bold symbols, e.g. $\mathbf{Q}$. Equality of matrices or vectors up to a scalar factor is written as $\sim$. $[\mathbf{a}]_\times$ denotes the skew-symmetric matrix associated with the cross product of 3-vectors.

Camera model. We use the sphere model for catadioptric projection introduced by Geyer and Daniilidis [28]. All central catadioptric cameras can be modeled by a unit sphere and a perspective projection, such that the projection of 3D points can be performed in two steps (Fig. 1). First, one projects the point onto the unit sphere, obtaining the intersection of the sphere and the line joining its center and the 3D point. There are two such intersection points, which are represented as $\mathbf{s}_\pm = \left(Q_1, Q_2, Q_3, \pm\sqrt{Q_1^2+Q_2^2+Q_3^2}\right)^\mathsf{T}$. These points are then projected in the second step, using a perspective projection P resulting in two image points, $\mathbf{q}_\pm = \mathsf{P}\mathbf{s}_\pm$, one of which is physically true. This model covers all central catadioptric cameras, encoded by ξ,

[1] http://webdiis.unizar.es/~lpuig/DLTOmniCalibration/Toolbox.tar.gz

which is the distance between the perspective camera and the center of the sphere, and ψ which is the distance between the center of the sphere and the image plane. We have $\xi = 0$ for perspective, $\xi = 1$ for para-catadioptric and $0 < \xi < 1$ for hyper-catadioptric cameras.

Let the unit sphere be located at the origin and the optical center of the perspective camera, at the point $\mathbf{C}_p = (0, 0, \xi)^\mathsf{T}$. The perspective camera is modeled by the projection matrix $\mathsf{P} \sim \mathsf{A}_p \mathsf{R}_p \left(\mathtt{I} \; -\mathbf{C}_p\right)$, where A_p is its calibration matrix. We assume it is of the form

$$\mathsf{A}_p = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} \tag{1}$$

with f the focal length and (c_x, c_y) the principal point. The rotation R_p denotes a rotation of the perspective camera looking at the mirror (this rotation is usually very small, thus often neglected). Rotation about the $z-$axis can always be neglected since it is coupled with the rotation of the whole system about the $z-$axis. Since both intrinsic and extrinsic parameters of the *perspective* camera are intrinsic parameters for the *catadioptric* camera, we replace $\mathsf{A}_p\mathsf{R}_p$ by a generic projective transformation K. Note that the focal length of the perspective camera in the sphere model is different from the focal length of the physical camera looking at the mirror; its value is actually determined by the physical camera's focal length, the mirror parameters and the rotation between the camera and the mirror (R_p). In Section 4 we study this relationship. The **intrinsic parameters of the catadioptric camera** are thus ξ and K.

To simplify, it is usual to work with the intermediate image points $\mathbf{r}_\pm \sim \mathsf{K}^{-1}\mathbf{q}_\pm$. Explicitly defined as $\mathbf{r}_\pm = (Q_1, Q_2, Q_3 \pm \xi\sqrt{Q_1^2 + Q_2^2 + Q_3^2})^\mathsf{T}$, before giving final results for the actual image points $\mathbf{q}_\pm$.

Lifted coordinates from symmetric matrix equations. The derivation of multi-linear relations for catadioptric imagery requires the use of lifted coordinates. The Veronese map [26] $V_{n,d}$ of degree d maps points of $\mathcal{P}^n$ into points of an m dimensional projective space $\mathcal{P}^m$, with $m = \begin{pmatrix} n+d \\ d \end{pmatrix} - 1$.

Consider the second order Veronese map $V_{2,2}$, that embeds the projective plane into the 5D projective space, by lifting the coordinates of point $\mathbf{q}$ to

$$\hat{\mathbf{q}} = \left(q_1^2 \; q_1q_2 \; q_2^2 \; q_1q_3 \; q_2q_3 \; q_3^2\right)^\mathsf{T} \tag{2}$$

Vector $\hat{\mathbf{q}}$ and matrix $\mathbf{q}\mathbf{q}^\mathsf{T}$ are composed by the same elements. The former can be derived from the latter through a suitable re-arrangement of parameters. Define $\mathbf{v}(\mathsf{U})$ as the vector obtained by stacking the columns of a generic matrix U [31]. For the case of $\mathbf{q}\mathbf{q}^\mathsf{T}$, $\mathbf{v}(\mathbf{q}\mathbf{q}^\mathsf{T})$ has several repeated elements because of the matrix symmetry. By left multiplication with a suitable permutation matrix S that adds the repeated elements, it follows that

$$\hat{\mathbf{q}} = \mathsf{D}^{-1} \underbrace{\begin{pmatrix} 1&0&0&0&0&0&0&0&0 \\ 0&1&0&1&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0&0 \\ 0&0&1&0&0&0&1&0&0 \\ 0&0&0&0&0&1&0&1&0 \\ 0&0&0&0&0&0&0&0&1 \end{pmatrix}}_{\mathsf{S}} \mathbf{v}(\mathbf{q}\mathbf{q}^\mathsf{T}), \tag{3}$$

with D a diagonal matrix, $D_{ii} = \sum_{j=1}^{9} S_{ij}$.

If U is symmetric, then it is uniquely represented by $\mathbf{v}_{sym}(\mathsf{U})$, the row-wise vectorization of its lower left triangular part:

$$\mathbf{v}_{sym}(\mathsf{U}) = \mathsf{D}^{-1}\mathsf{S}\mathbf{v}(\mathsf{U}) = (U_{11}, U_{21}, U_{22}, U_{31}, \cdots, U_{nn})^\mathsf{T} \tag{4}$$

Since S gives us the position of the repeated elements of $\mathbf{v}(\mathsf{U})$, it is easy to recover $\mathbf{v}(\mathsf{U})$ from $\mathbf{v}_{sym}(\mathsf{U})$

$$\mathbf{v}(\mathsf{U}) = \mathsf{S}^\mathsf{T}\mathbf{v}_{sym}(\mathsf{U}) \tag{5}$$

In this paper, we use the Veronese map $V_{3,2}$ to lift the homogeneous coordinates of 3D points $\mathbf{Q}$ to 10-vectors $\hat{\mathbf{Q}}$.

Lifted matrices. Let us now discuss the lifting of linear transformations, induced by liftings of points. Consider A such that $\mathbf{r} = \mathsf{A}\mathbf{q}$. The relation $\mathbf{r}\mathbf{r}^\mathsf{T} = \mathsf{A}(\mathbf{q}\mathbf{q}^\mathsf{T})\mathsf{A}^\mathsf{T}$ can be written as a vector mapping

$$\mathbf{v}(\mathbf{r}\mathbf{r}^\mathsf{T}) = (\mathsf{A} \otimes \mathsf{A})\mathbf{v}(\mathbf{q}\mathbf{q}^\mathsf{T}), \tag{6}$$

with $\otimes$ denoting the Kronecker product [31]. Using the symmetric vectorization, we have $\hat{\mathbf{q}} = \mathbf{v}_{sym}(\mathbf{q}\mathbf{q}^\mathsf{T})$ and $\hat{\mathbf{r}} = \mathbf{v}_{sym}(\mathbf{r}\mathbf{r}^\mathsf{T})$, thus, from (5) and (6):

$$\hat{\mathbf{r}} = \underbrace{\mathsf{D}^{-1}\mathsf{S}(\mathsf{A} \otimes \mathsf{A})\mathsf{S}^\mathsf{T}}_{\widehat{\mathsf{A}}} \hat{\mathbf{q}} \tag{7}$$

where $\widehat{\mathsf{A}}$ represents the lifted linear transformation, which is a 6×6 matrix.

A few useful properties of the lifting of transformations are [31,32]:

$$\widehat{\mathsf{AB}} = \widehat{\mathsf{A}}\widehat{\mathsf{B}} \qquad \widehat{\mathsf{A}^{-1}} = \widehat{\mathsf{A}}^{-1} \qquad \widehat{\mathsf{A}^\mathsf{T}} = \mathsf{D}^{-1}\widehat{\mathsf{A}}^\mathsf{T}\mathsf{D} \tag{8}$$

3 Generic Projection Matrix

As explained in the previous section, a 3D point is mathematically projected to two image points. Sturm and Barreto [27] represented these two 2D points via the degenerate dual conic generated by them, i.e. the dual conic containing exactly the lines going through

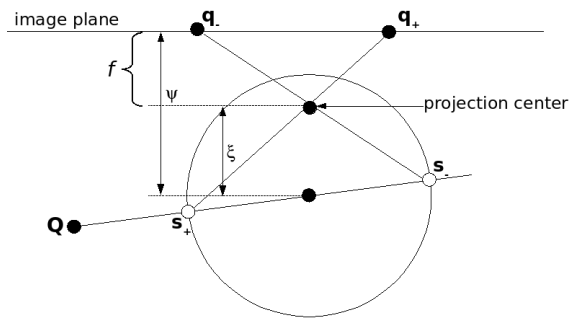


Fig. 1 Projection of a 3D point to two image points in the sphere camera model. The z-axis of the camera coordinate system is positive upwards. The camera is looking up.

at least one of the two points. Let the two image points be $\mathbf{q}_+$ and $\mathbf{q}_-$; the dual conic is then given by

$$\Omega \sim \mathbf{q}_+\mathbf{q}_-^\mathsf{T} + \mathbf{q}_-\mathbf{q}_+^\mathsf{T} \tag{9}$$

The vectorized matrix of the conic can be computed as shown below using the lifted 3D point coordinates, intrinsic and extrinsic parameters.

$$\mathbf{v}_{sym}(\Omega) \sim \widehat{\mathsf{K}}_{6\times6}\mathsf{X}_\xi\widehat{\mathsf{R}}_{6\times6}\left(\mathtt{I}_6\ \mathsf{T}_{6\times4}\right)\hat{\mathbf{Q}}_{10} \tag{10}$$

Here, R represents the rotation of the catadioptric camera. X_ξ and $\mathsf{T}_{6\times4}$ depend only on the sphere model parameter ξ and the position of the catadioptric camera $\mathbf{C} = (t_x, t_y, t_z)$ respectively, as shown here:

$$\mathsf{X}_\xi = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ -\xi^2 & 0 & -\xi^2 & 0 & 0 & 1-\xi^2 \end{pmatrix} \tag{11}$$

$$\mathsf{T}_{6\times4} = \begin{pmatrix} -2t_x & 0 & 0 & t_x^2 \\ -t_y & -t_x & 0 & t_xt_y \\ 0 & -2t_y & 0 & t_y^2 \\ -t_z & 0 & -t_x & t_xt_z \\ 0 & -t_z & -t_y & t_yt_z \\ 0 & 0 & -2t_z & t_z^2 \end{pmatrix} \tag{12}$$

Thus, a 6×10 **catadioptric projection matrix**, P_{cata}, can be expressed by its intrinsic and extrinsic parameters, like the projection matrix of a perspective camera.

$$\mathsf{P}_{cata} = \underbrace{\widehat{\mathsf{K}}\mathsf{X}_\xi}_{\mathsf{A}_{cata}}\underbrace{\widehat{\mathsf{R}}_{6\times6}\left(\mathtt{I}_6\ \mathsf{T}_{6\times4}\right)}_{\mathsf{T}_{cata}} \tag{13}$$

3.1 Computation of the Generic Projection Matrix

Here we show the way used to compose the equations using 3D-2D correspondences to compute P_{cata}. Analogous to the perspective case ($[\mathbf{q}]_\times\mathsf{P}\mathbf{Q} = \mathbf{0}$), we write the constraint based on the lifted coordinates [27]:

$$\widehat{[\mathbf{q}]_\times}\ \mathsf{P}_{cata}\ \hat{\mathbf{Q}} = \mathbf{0} \tag{14}$$

This is a set of 6 linear homogeneous equations in the coefficients of P_{cata}. Using the Kronecker product, this can be written in terms of the 60-vector $\mathbf{p}_{cata}$ containing the 60 coefficients of P_{cata}:

$$\left(\hat{\mathbf{Q}}^\mathsf{T} \otimes \widehat{[\mathbf{q}]_\times}\right)\mathbf{p}_{cata} = \mathbf{0}_6 \tag{15}$$

Stacking these equations for n 3D-2D correspondences gives a system of equations of size $6n \times 60$, which can be solved by linear least squares, e.g. using the SVD (Singular Value Decomposition). Note that the minimum number of required correspondences is 20: a 3×3 skew symmetric matrix has rank 2, its lifted counterpart rank 3. Therefore, each correspondence provides only 3 independent linear constraints.

3.2 Generic Projection Matrix and Calibration

The calibration process consists of getting the intrinsic and extrinsic parameters of a camera. Once P_{cata} has been computed from point correspondences, our purpose is to decompose P_{cata} as in (13). Consider first the leftmost 6×6 submatrix of P_{cata}:

$$\mathsf{P}_s \sim \widehat{\mathsf{K}}\mathsf{X}_\xi\widehat{\mathsf{R}} \tag{16}$$

Let us define $\mathsf{M} = \mathsf{P}_s\mathsf{D}^{-1}\mathsf{P}_s^\mathsf{T}$. Using the properties given in (8) and knowing that for a rotation matrix $\mathsf{R}^{-1} = \mathsf{R}^\mathsf{T}$, we can write $\widehat{\mathsf{R}}^{-1} = \mathsf{D}^{-1}\widehat{\mathsf{R}}^\mathsf{T}\mathsf{D}$. And from that we obtain $\mathsf{D}^{-1} = \widehat{\mathsf{R}}\mathsf{D}^{-1}\widehat{\mathsf{R}}^\mathsf{T}$ which we use to eliminate the rotation parameters:

$$\mathsf{M} \sim \widehat{\mathsf{K}}\mathsf{X}_\xi\widehat{\mathsf{R}}\,\mathsf{D}^{-1}\widehat{\mathsf{R}}^\mathsf{T}\mathsf{X}_\xi^\mathsf{T}\widehat{\mathsf{K}}^\mathsf{T} = \widehat{\mathsf{K}}\mathsf{X}_\xi\,\mathsf{D}^{-1}\mathsf{X}_\xi^\mathsf{T}\widehat{\mathsf{K}}^\mathsf{T} \tag{17}$$

Equation (17) holds up to scale, i.e. there is a λ with $\mathsf{M} = \lambda\widehat{\mathsf{K}}\mathsf{X}_\xi\,\mathsf{D}^{-1}\mathsf{X}_\xi^\mathsf{T}\widehat{\mathsf{K}}^\mathsf{T}$. For initialization we assume that the camera is well aligned with the mirror axis, i.e. assume that $\mathsf{R}_p = \mathtt{I}$, thus $\mathsf{K} = \mathsf{A}_p = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix}$.

We then use some elements of M to extract the intrinsic parameters:

$$\mathsf{M}_{16} = \lambda\left(-(f^2\xi^2) + c_x^2(\xi^4 + c_x(1-\xi^2)^2\right)$$
$$\mathsf{M}_{44} = \lambda\left(\frac{f^2}{2} + c_x^2(2\xi^4 + (1-\xi^2)^2)\right)$$
$$\mathsf{M}_{46} = \lambda c_x(2\xi^4 + (1-\xi^2)^2)$$
$$\mathsf{M}_{56} = \lambda c_y(2\xi^4 + (1-\xi^2)^2)$$
$$\mathsf{M}_{66} = \lambda\left(2\xi^4 + (1-\xi^2)^2\right)$$

The intrinsic parameters are computed as follows:

$$c_x = \frac{\mathsf{M}_{46}}{\mathsf{M}_{66}} \qquad c_y = \frac{\mathsf{M}_{56}}{\mathsf{M}_{66}} \qquad \xi = \sqrt{\frac{\frac{M_{16}}{M_{66}} - c_x^2}{-2(\frac{M_{44}}{M_{66}} - c_x^2)}}$$
$$f = \sqrt{2(2\xi^4 + (1-\xi^2)^2)\left(\frac{M_{44}}{M_{66}} - c_x^2\right)}$$

After extracting the intrinsic part A_{cata} of the projection matrix, we are able to obtain the 6×10 extrinsic part T_{cata} by multiplying P_{cata} with the inverse of A_{cata}:

$$\mathsf{T}_{cata} = \widehat{\mathsf{R}}_{6\times 6}(\mathtt{I}_6\mathsf{T}_{6\times 4}) \sim (\widehat{\mathsf{K}}\mathsf{X}_\xi)^{-1}\mathsf{P}_{cata} \tag{18}$$

Hence, the leftmost 6×6 part of T_{cata} will be the estimate of the lifted rotation matrix $\widehat{\mathsf{R}}_{est}$. If we multiply the inverse of this matrix with the rightmost 6×4 part of T_{cata}, we obtain an estimate for the translation $(\mathsf{T}_{6\times 4})$. This translation should have an ideal form as given in (12) and we are able to identify translation vector elements (t_x, t_y, t_z) from it straightforwardly.

We finally have to handle the fact that the estimated $\widehat{\mathsf{R}}_{est}$ will not, in general, be an exact lifted rotation matrix. This lifted rotation matrix in particular is oversized since it considers the lifting of a full rotation matrix $\widehat{\mathsf{R}} = \widehat{\mathsf{R}}_z(\gamma)\widehat{\mathsf{R}}_y(\beta)\widehat{\mathsf{R}}_x(\alpha)$. For illustration in (19) we show the lifting of a rotation matrix around the $x-$axis.

$$\widehat{\mathsf{R}}_x(\alpha) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \cos\alpha & 0 & -\sin\alpha & 0 & 0 \\ 0 & 0 & \cos^2\alpha & 0 & -2\cos\alpha\sin\alpha & \sin^2\alpha \\ 0 & \sin\alpha & 0 & \cos\alpha & 0 & 0 \\ 0 & 0 & \cos\alpha\sin\alpha & 0 & \cos^2\alpha - \sin^2\alpha & -\cos\alpha\sin\alpha \\ 0 & 0 & \sin^2\alpha & 0 & 2\cos\alpha\sin\alpha & \cos^2\alpha \end{pmatrix} \tag{19}$$

Since P_{cata} has been estimated up to scale it is impossible to extract the rotation components from single elements of $\widehat{\mathsf{R}}_{est}$. To deal with this problem we algebraically manipulate the ratios between the elements of this lifted matrix and we extract the angles one by one. First, we recover the rotation angle around the z axis, $\gamma = tan^{-1}\left(\frac{\widehat{\mathsf{R}}_{est,51}}{\widehat{\mathsf{R}}_{est,41}}\right)$. Then, $\widehat{\mathsf{R}}_{est}$ is modified by being multiplied by the inverse of the rotation around the z axis, $\widehat{\mathsf{R}}_{est} = \widehat{\mathsf{R}}_z^{-1}(\gamma)\widehat{\mathsf{R}}_{est}$. Then, the rotation angle around the y axis, β, is estimated and $\widehat{\mathsf{R}}_{est}$ is modified $\beta = tan^{-1}\left(\frac{-\widehat{\mathsf{R}}_{est,52}}{\widehat{\mathsf{R}}_{est,22}}\right)$, $\widehat{\mathsf{R}}_{est} = \widehat{\mathsf{R}}_y^{-1}(\beta)\,\widehat{\mathsf{R}}_{est}$. Finally, the rotation angle around the x axis, α, is estimated as $\alpha = tan^{-1}\left(\frac{\widehat{\mathsf{R}}_{est,42}}{\widehat{\mathsf{R}}_{est,22}}\right)$.

3.3 Other Parameters of Non-linear Calibration

The intrinsic and extrinsic parameters extracted in closed-form in Section 3.2 are not always adequate to model a real camera. Extra parameters are needed to correctly model the catadioptric system, namely, tilting and lens distortions.

As mentioned before $\widehat{\mathsf{K}} = \widehat{\mathsf{A}_p\mathsf{R}_p} = \widehat{\mathsf{A}}_p\widehat{\mathsf{R}}_p$ where R_p is the rotation between camera and mirror coordinate systems, i.e. tilting. Tilting has only R_x and R_y components, because rotation around the optical axis, R_z, is coupled with the external rotation around the z axis of the entire catadioptric system. Note that tilting angles of the sphere camera model are not equivalent to the tilting angles of the actual perspective camera looking at the mirror.

As is well known, imperfections due to lenses are modeled as distortions for camera calibration. Radial distortion models contraction or expansion with respect to the image center and tangential distortion models lateral effects. To add these distortion effects to our calibration algorithm, we employed the approach of Heikkila and Silven [33].

Radial distortion:

$$\begin{aligned} \Delta x &= x(k_1r^2 + k_2r^4 + k_3r^6 + ..) \\ \Delta y &= y(k_1r^2 + k_2r^4 + k_3r^6 + ..) \end{aligned} \tag{20}$$

where $r = \sqrt{x^2 + y^2}$ and $k_1, k_2..$ are the radial distortion parameters. We observe that estimating two parameters is enough for an adequate estimation.

Tangential distortion:

$$\begin{aligned} \Delta x &= 2p_1xy + p_2(r^2 + 2x^2) \\ \Delta y &= p_1(r^2 + 2y^2) + 2p_2xy \end{aligned} \tag{21}$$

where $r = \sqrt{x^2 + y^2}$ and p_1, p_2 are the tangential distortion parameters.

Once we have identified all the parameters to be estimated we perform a non-linear optimization to compute the whole model. We use the Levenberg-Marquardt method (LM)[2]. The minimization criterion is the root

[2] Method provided by the function **lsqnonlin** in Matlab

mean square (RMS) of distance between a measured image point and its reprojected correspondence. Since the projection equations we use map 3D points to dual image conics, we have to extract the two potential image points from it. The one closer to the measured point is selected and then the reprojection error measured. We take as initial values the parameters obtained from P_{cata} and initialize the additional 4 distortion parameters and the tilt angles in R_p, by zero.

3.4 Algorithm to Compute P_{cata}

Here we summarize the algorithm used to compute the generic projection matrix P_{cata}.

1. **Linear Solution**. Using 3D-2D correspondences we compute P_{cata} by a DLT-like approach.
2. **Intrinsic/Extrinsic Parameter Extraction**. Assuming that the perspective camera is perfectly aligned with the mirror axis, i.e. there is no tilting and that the images are not distorted. We extract from the linear solution, the intrinsic (ξ, f, c_x, c_y) and extrinsic $(\alpha, \beta, \gamma, t_x, t_y, t_z)$ parameters in closed-form.
3. **Initialization Vector**. An initialization vector is constructed with the extracted parameters. Two parameters are added to consider the tilting angles $(\mathsf{r}_x, \mathsf{r}_y)$ and four more corresponding to the radial (k_1, k_2) and tangential (p_1, p_2) distortion.
4. **Non-linear Optimization Process**. Using this vector as an initialization vector, we perform a non-linear optimization process using the LM algorithm. The minimization criterion is the reprojection error.

4 Other Theoretical and Practical Issues

In the last section we explained that 20 3D-2D correspondences are enough to compute the calibration of the central catadioptric systems. In principle these 20 correspondences can be located anywhere inside the FOV of the catadioptric system. Since we want to construct a feasible calibration system based on planar patterns we restrict the 3D points to be located in planes. From simulations we observed that the minimum number of planes where the 3D points should be located is three in the general case. In particular, two planes can be used to compute P_{cata} if several constraints are imposed, but the simplicity of using linear equations is lost.

Since we restrict the calibration points to lie on planes (planar grid-based calibration) some degeneracies can appear if the calibration points are located in a particular configuration. Something similar to the pinhole camera case with the twisted cubic [34], for which calibration fails even if the points lie on more than two planes. However, a complete analysis of such degeneracies is out of the scope of this paper.

In this section we present a proof that points lying in three different planes are required to linearly and uniquely compute the generic projection matrix P_{cata}. We also show that under several assumptions we can compute P_{cata} from points lying in just two planes. We also explain how the parameters in the sphere camera model are related with those of the real catadioptric system.

4.1 Three Planes Are Needed to Compute P_{cata} Using Linear Equations.

Here we show that in order to compute P_{cata}, the 3D calibration points must lie in at least 3 different planes. We first prove that two planes are not sufficient. Let $\mathbf{\Pi}_1$ and $\mathbf{\Pi}_2$ be the two planes. Hence, each calibration point $\mathbf{Q}$ satisfies $\left(\mathbf{\Pi}_1^\mathsf{T}\mathbf{Q}\right)\left(\mathbf{\Pi}_2^\mathsf{T}\mathbf{Q}\right) = 0$. This can be written as a linear constraint on the lifted calibration points: $\mathbf{p}^\mathsf{T}\hat{\mathbf{Q}} = 0$, where the 10-vector $\mathbf{p}$ depends exactly on the two planes. Thus, if P_{cata} is the true 6×10 projection matrix, then adding some multiple of $\mathbf{p}^\mathsf{T}$ to any row of P_{cata} gives another 6×10 projection matrix, $\bar{\mathsf{P}}_{cata}$, which maps the calibration points to the same image entities as the true projection matrix. We may write the ambiguity as

$$\bar{\mathsf{P}}_{cata} = \mathsf{P}_{cata} + \mathbf{v}\mathbf{p}^\mathsf{T} \tag{22}$$

where $\mathbf{v}$ is a 6-vector and represents the 6-dof on P_{cata} that can not be recovered using only linear projection equations and calibration points located in only two planes. This is not the case for perspective cameras, where two planes are enough to compute the 3×4 perspective projection matrix.

For three planes, there is no linear equation as above that holds for all calibration points. Hence, also supported by our experiments, it seems plausible that three planes are sufficient for uniquely computing the projection matrix. Note that by planes we do not mean that calibration grids have to be composed of three or more planar grids. The planes can be virtual: whenever it is possible to fit the two planes to the whole set of 3D points, P_{cata} can not be computed.

4.2 Adding Constraints to Estimate the Projection Matrix from Points on Two Planes Only

In the last section we observe that to compute P_{cata} linearly and uniquely, 3D points must be sufficiently

well distributed, such that no two planes contain all of them. In this section we analyze what prior information allows nevertheless to compute the calibration parameters using two planes. We know by (22) that the true projection matrix is related to any other solution by

$$\mathsf{P}_{cata} = \bar{\mathsf{P}}_{cata} - \mathbf{v}\mathbf{p}^{\mathsf{T}} \tag{23}$$

Consider the equation to eliminate the extrinsic parameters:

$$\mathsf{M} \sim \mathsf{P}_s \mathsf{D}^{-1} \mathsf{P}_s^{\mathsf{T}} \tag{24}$$

where P_s is the leftmost 6×6 submatrix of P_{cata}.

Now we redefine it as follows:

$$\mathsf{M} \sim (\bar{\mathsf{P}}_s - \mathbf{v}\mathbf{p}_s^{\mathsf{T}})\mathsf{D}^{-1}(\bar{\mathsf{P}}_s - \mathbf{v}\mathbf{p}_s^{\mathsf{T}})^{\mathsf{T}} \tag{25}$$

where $\bar{\mathsf{P}}_s$ is the leftmost 6×6 submatrix of $\bar{\mathsf{P}}_{cata}$ and $\mathbf{p}_s$ is the first 6 elements of the 10-vector $\mathbf{p}$. Assuming that the two planes are perpendicular to each other, we can write $\mathbf{\Pi}_1 = [1,0,0,0]^{\mathsf{T}}$ and $\mathbf{\Pi}_2 = [0,1,0,0]^{\mathsf{T}}$ which gives us $\mathbf{p}_s = [0,1,0,0,0,0]^{\mathsf{T}}$ (we obtain $\mathbf{p}$ by $\mathbf{v}_{sym}(\mathbf{\Pi}_1\mathbf{\Pi}_2^{\mathsf{T}} + \mathbf{\Pi}_2\mathbf{\Pi}_1^{\mathsf{T}})$ since $\mathbf{\Pi}_1\mathbf{\Pi}_2^{\mathsf{T}}$ represents a degenerate dual conic on which all $\mathbf{Q}$ lie).

Let us develop (25):

$$\mathsf{M} \sim \underbrace{\bar{\mathsf{P}}_s\mathsf{D}^{-1}\bar{\mathsf{P}}_s^{\mathsf{T}}}_{\bar{\mathsf{M}}} - \underbrace{\bar{\mathsf{P}}_s\mathsf{D}^{-1}\mathbf{p}_s}_{\mathbf{b}}\mathbf{v}^{\mathsf{T}} - \mathbf{v}\underbrace{\mathbf{p}_s^{\mathsf{T}}\mathsf{D}^{-1}\bar{\mathsf{P}}_s}_{\mathbf{b}^{\mathsf{T}}} + \mathbf{v}\underbrace{\mathbf{p}_s^{\mathsf{T}}\mathsf{D}^{-1}\mathbf{p}_s}_{\rho}\mathbf{v}^{\mathsf{T}} \tag{26}$$

$$\mathsf{M} \sim \bar{\mathsf{M}} - \mathbf{b}\mathbf{v}^{\mathsf{T}} - \mathbf{v}\mathbf{b}^{\mathsf{T}} + \rho\mathbf{v}\mathbf{v}^{\mathsf{T}} \tag{27}$$

We can compute ρ, it is $\frac{1}{2}(D_{22} = 2)$. So we just need to obtain elements of $\mathbf{v}$ to recover P_{cata}. The principal point can be computed using different approaches, one of these is shown in [20], which requires the user interaction. Let us suppose we know the principal point (c_x, c_y), and we put the origin of the image reference system on it $(c_x = 0, c_y = 0)$. Then we have:

$$\mathsf{M} = \begin{pmatrix} f^4 & 0 & 0 & 0 & 0 & -f^2\xi^2 \\ 0 & \frac{f^4}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & f^4 & 0 & 0 & -f^2\xi^2 \\ 0 & 0 & 0 & \frac{f^2}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{f^2}{2} & 0 \\ -f^2\xi^2 & 0 & -f^2\xi^2 & 0 & 0 & 2\xi^4 + (1-\xi^2)^2 \end{pmatrix} \tag{28}$$

From this matrix we can extract 6 equations to solve for the elements of $\mathbf{v}$. For example: $\mathsf{M}_{11} - \mathsf{M}_{33} = 0$, $\mathsf{M}_{11} - 2\mathsf{M}_{22} = 0$, $\mathsf{M}_{44} - \mathsf{M}_{55} = 0$, $\mathsf{M}_{13} = 0$, $\mathsf{M}_{35} = 0$, $\mathsf{M}_{56} = 0$.

We test the case where $f_x = f_y$ using simulated data with perfect 3D-2D correspondences. We observe that as explained in theory, the only modified column is the second one, described by the vector $\mathbf{p}_s = [0,1,0,0,0,0]^{\mathsf{T}}$. In this case we are able to obtain the correct P_{cata}. However, when we added Gaussian noise to the 3D-2D correspondences, more than one column is modified making very difficult to recover the real projection matrix. Therefore, we conclude that the approach using points lying in just two planes is not suitable to compute the generic projection matrix in real situations. We continue our experiments with calibration grids having three planes.

4.3 Relation Between the Real Catadioptric System and the Sphere Camera Model

As mentioned before, the focal length in the sphere model is not the same as the focal length of the real camera, looking at the mirror. This is best seen for the para-catadioptric case, where the real camera is orthographic (infinite focal length) whereas the perspective camera in the sphere model has a finite focal length. The analogous is true also for the tilting parameters.

In this section we analyze the relation between the parameters present in a real catadioptric system and their representation in the sphere camera model. The objective of this analysis is to observe if it is possible to recover the intrinsic parameters of the real catadioptric system from their counterparts in the sphere camera model. We analyze the tilting and focal length f of the conventional camera.

4.3.1 Tilting

Tilting in a camera can be defined as a rotation of the image plane w.r.t. the pinhole. This is also equivalent to tilting the incoming rays since both have the same pivoting point: the pinhole. In the Fig. 2a the tilt in a catadioptric camera is represented. Similarly, the tilt in the sphere model (R_p in $\mathsf{K} = \mathsf{A}_p\mathsf{R}_p$) corresponds to tilting the rays coming to the perspective camera of the sphere model (Fig. 2b). Although the same image is generated by both models, the tilting angles are not identical, even they are not proportional to each other. So, it is also not possible to obtain the real system tilt amount by multiplying the sphere model tilt by a coefficient.

4.3.2 Focal Length f.

The composition of para-catadioptric and hyper-catadioptric systems is different. The first one uses a parabolic

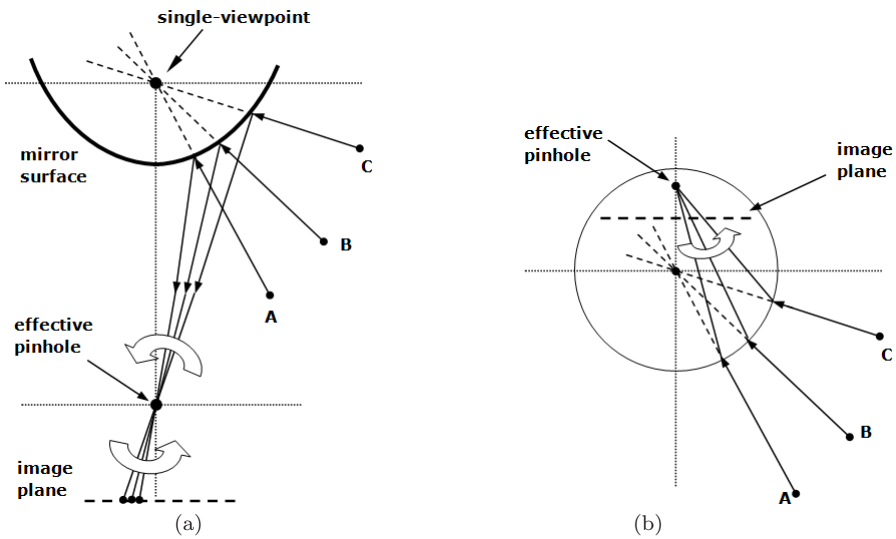


Fig. 2 Tilt in a real system (a) and in sphere model (b).

mirror and an orthographic camera. In this case the focal length of the real system, f_c, is infinite.

For the hyper-catadioptric system, we are able to relate f with the focal length of the perspective camera in the real system, f_c. We start with defining explicitly the projection matrix K. Assuming image skew is zero, $\mathsf{R}_p = \mathtt{I}$ and principal point is $(0,0)$, K is given in [35] as

$$\mathsf{K} = \begin{pmatrix} (\psi-\xi)f_c & 0 & 0 \\ 0 & (\psi-\xi)f_c & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{29}$$

where ψ corresponds to the distance between the effective viewpoint and the re-projection plane (cf. Fig. 1). The relation between the focal lengths is $f = (\psi-\xi)f_c$. From the same reference [35] we get

$$\xi = \frac{d}{\sqrt{d^2+4p^2}} \qquad \psi = \frac{d+2p}{\sqrt{d^2+4p^2}} \tag{30}$$

where d is the distance between the foci of the hyperboloid and $4p$ equals to the latus rectum. Developing the equations we obtain p in terms of d and ξ, $2p = d\sqrt{1-\xi^2}/\xi$, which is used to obtain $\psi = \xi + \sqrt{1-\xi^2}$. With this final relation we can write

$$f = f_c\sqrt{1-\xi^2} \tag{31}$$

from which we extract the focal length of the perspective camera in the real system

$$f_c = \frac{f}{\sqrt{1-\xi^2}}. \tag{32}$$

5 Calibration Experiments with a Simulated Environment

We use a simulated calibration object having 3 planar faces which are perpendicular to each other. The size of a face is 50x50cm. There are a total of 363 points, since each face has 11x11 points and the distance between points is 5cm. The omnidirectional image fits in a 1 Megapixel square image. To represent the real world points we expressed the coordinates in meters, so they are normalized in a sense. This is important because we observed that using large numerical values causes bad estimations with noisy data in the DLT algorithm. Normalization of image coordinates is also performed since we observed a positive effect both on estimation accuracy and the convergence time. Therefore, in the presented experiments, 3D point coordinates are in meters and image coordinates are normalized to be in the same order of magnitude, this is performed by dividing the image coordinates by a constant.

We performed experiments for different settings of intrinsic parameters and varying position of the 3D calibration grid. We especially tested the accuracy of calibration to variations in the intrinsic parameters (ξ and f), the distance between the camera and the grid and the orientation of the grid w.r.t. the camera. In all these cases, we measure the errors in final estimates of ξ and f, the main parameters of the sphere camera model. Errors are depicted in Fig. 3, where an individual graph is plotted for each case for clarity. In all experiments, Gaussian noise with $\sigma = 1$ pixel is added to the actual coordinates of grid corners. The plotted errors are $err_\xi = 100\cdot|\xi_{nonlin} - \xi_{real}|/\xi_{real}$ and $err_f = 100\cdot|f_{nonlin} - f_{real}|/f_{real}$. For all the nodes in

the graphs, the experiment was repeated 100 times and the mean value of estimates is plotted.

Fig. 3a shows the effect of changing distance between the camera and the grid. From left to right in the graph distance-to-grid increases and distance values are selected randomly within the given ranges. When the distance is small, we reach an "optimal" position, such that the grid fills the image well. As the grid moves away from the omnidirectional camera, its image gets smaller and smaller. Examples of the omnidirectional images generated are shown in Fig. 4. In Fig. 4a, distance-to-grid is 45 cm., whereas in Fig. 4b it is 60 cm. The quality of parameter estimation decreases with increasing distance. Since the grid covers a smaller area, the same amount of noise (in pixels) affects the non-linear optimization more and errors in non-linear results increase as can be expected. We observe the importance of a good placement of the calibration grid, i.e. such that it fills the image as much as possible.

Fig. 3b shows the effect of real ξ and f values on the estimation error (for two different distance-to-grid value ranges). From left to right in the graph, ξ and f values decrease. They decrease in parallel, otherwise decreasing ξ with fixed f would cause grid to get smaller in the image. We truncated (ξ,f) pairs at ξ=0.6 since even smaller ξ values are unlikely for omnidirectional cameras. We observe that larger (ξ,f) values produce slightly better results especially for increased distances. This observation can also be made in Fig. 3a since the errors are depicted with two different ranges of intrinsic parameter values. The reason is that for fixed distance-to-grid values, higher (ξ,f) spreads the grid points to a larger area in the image, which decreases the effect of noise. Observe Fig. 4b with Fig. 4c, where distance-to-grid values are equal but Fig. 4b has higher (ξ,f).

Fig. 3c shows the effect of changing orientation of the grid w.r.t. the camera. This is expressed in terms of the angle between the optical axis of the omnidirectional camera and the grid center. The grid is not rotated independently from the camera axis because camera (mirror) has to see the inside of the 3D grid always. Fig. 4d shows the case when the grid is rotated so that the angle between its center and camera optical axis is 40°. Compare with Fig. 4b, where the intersection of the three planes of the grid is at the image center. We observe improvement with rotation specially for increased distance-to-grid since grid points are more spread and effect of noise decreases.

In Table 1, we list the results of the algorithm after linear (DLT) and non-linear steps for a few cases. Our main observation is that the errors in linear estimates, ξ_{DLT} and f_{DLT}, are biased (values are smaller than they should be). For all the cases, however, the true intrinsic parameters are reached after non-linear optimization, modulo errors due to noise.

	Distance-to-grid			
	45 cm.		60 cm.	
ξ_{real}	0.96	0.8	0.96	0.80
f_{real}	360	270	360	270
ξ_{DLT}	0.54	0.40	0.04	0.03
f_{DLT}	361	268	243	190
ξ_{nonlin}	0.96	0.80	0.98	0.78
f_{nonlin}	360	270	365	266
err_ξ	0.0	0.0	2.1	2.5
err_f	0.0	0.1	1.4	1.5

Table 1 Initial and optimized estimates with different intrinsics and distance-to-grid values. Amount of noise: $\sigma = 1$ pixel. ξ_{DLT},f_{DLT} and ξ_{nonlin},f_{nonlin} are the results of the DLT algorithm and non-linear optimization respectively, err_ξ and err_f are the relative errors, in percent after non-linear optimization.

	Real values	$\sigma = 0.5$		$\sigma = 1$	
		Initial	Estimated	Initial	Estimated
f	360	361	360	354	360
c_x	500	503	500	505	500
c_y	500	498	500	509	500
ξ	0.96	0.84	0.96	0.53	0.96
$R_x(\alpha)$	-0.62	-0.60	-0.62	-0.40	-0.62
$R_y(\beta)$	0.62	0.62	0.62	0.65	0.62
$R_z(\gamma)$	0.17	0.15	0.17	0.18	0.17
t_x	0.30	0.38	0.30	0.45	0.30
t_y	0.30	0.40	0.30	0.44	0.30
t_z	0.20	0.05	0.20	0.01	0.20
RMSE			0.70		1.42

Table 2 Non-linear optimization results for a hyper-catadioptric system, 10 parameters (rotation, translation and intrinsic) are optimized. Distance-to-grid is 45 cm. and grid center coincides with camera optical axis (no rotation).

5.1 Estimation Errors for Different Camera Types

Here we discuss the intrinsic and extrinsic parameter estimation for the two most common catadioptric systems: hyper-catadioptric and para-catadioptric, with hyperbolic and parabolic mirrors respectively. We also discuss calibration results for perspective cameras.

5.1.1 Hyper-catadioptric System.

Table 2 shows non-linear optimization results including the rotation and translation parameters for fixed intrinsic parameters which corresponds to a hyper-catadioptric system. 3D pattern is used at the "optimal" grid position, i.e. it fills the omnidirectional image like Fig. 4a. Results are in accordance with Table 1 and Fig. 3.

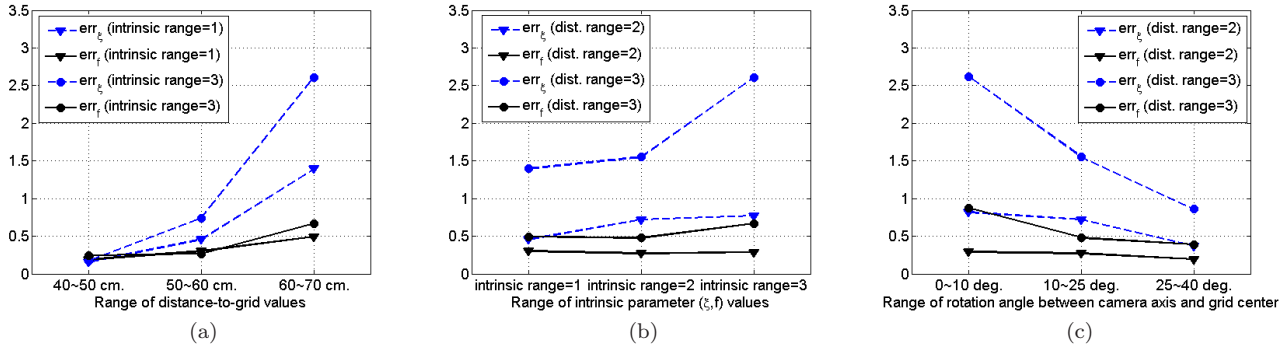


Fig. 3 Relative errors for ξ and f after non-linear optimization (in percent) for varying intrinsic parameters and varying position of the 3D calibration grid. For all the nodes in the graphs, the experiment was repeated 100 times and the mean value of estimates is plotted. Real intrinsics, distance and orientation values are selected randomly from the ranges given in $x-$axis. Intrinsic parameters range 1: (ξ, f)=[(0.96,360) (0.84,300)], range 2: (ξ, f)=[(0.84,300) (0.72,250)], range 3: (ξ, f)=[(0.72,250) (0.60,210)]. Distance-to-grid (in cm.) range 1: [40 50], range 2: [50 60], range 3: [60 70]. In (a), (b) and (c), errors depicted versus increasing distance-to-grid, decreasing (ξ, f) pairs and increasing rotation angle respectively.

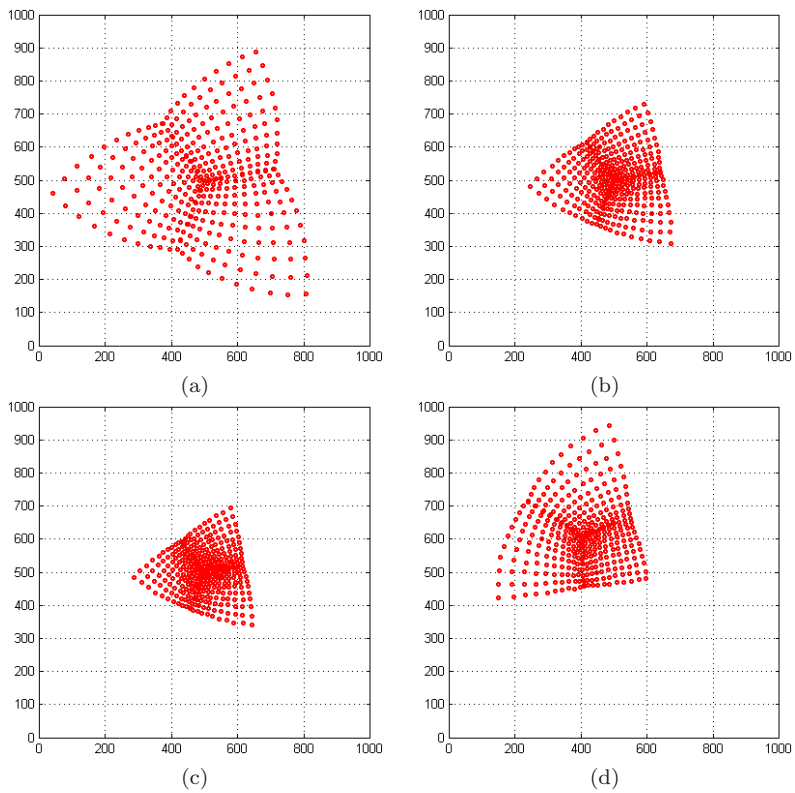


Fig. 4 Omnidirectional images generated with varying intrinsics, distance-to-grid and orientation. (a) (ξ, f)=(0.96,360), distance=45cm., no rotation. (b) (ξ, f)=(0.96,360), distance=60cm., no rotation. (c) (ξ, f)=(0.76,270), distance=60cm., no rotation. (d) (ξ, f)=(0.96,360), distance=60cm., rotated by 40°.

5.1.2 Para-catadioptric System.

Here $\xi = 1$, which has a potential to disturb the estimations because X_ξ becomes a singular matrix. We observe that the results of the DLT algorithm are not as close to the real values when compared to the hyper-catadioptric system (cf. initial values in Table 2). However, the non-linear optimization is able to estimate the parameters as successful as the hyper-catadioptric examples given in Table 2.

5.1.3 Perspective Camera.

In the sphere camera model, $\xi = 0$ corresponds to the perspective camera. Our estimations in linear and non-linear steps are as successful as with the hyper-catadioptric case and thus not shown in detail here.

5.2 Tilting and Distortion

It seems intuitive that small amounts of tangential distortion and tilting have a similar effect on the image. In our simulations we observed that trying to estimate both of them does not succeed. Therefore, we investigate if we can estimate tangential distortion of camera optics by tilt parameters, or estimate tilt in the system by tangential distortion parameters.

When there exists no tilt but tangential distortion and we try to estimate tilting parameters, we observed that the direction and amount of $tilt_x, tilt_y, c_x$ and c_y changes proportionally to the tangential distortion applied and the RMSE decreases. However, the RMSE does not reach as low values as when there is no distortion. In the noiseless case, for example, the RMSE is not zero. Hence, we concluded that tilt parameters compensate the tangential distortion effect up to some extent, but not perfectly. We also investigated if tilting can be compensated by tangential distortion parameters and we had very similar results. Thus, tangential distortion parameters have the same capability to estimate tilting.

6 Experiments with Real Images using a 3D Pattern

In this section we perform experiments of camera calibration using a 3D pattern, cf. Fig. 5(a). The 3D pattern has been measured accurately doing a photogrammetric reconstruction by bundle adjustment. We use 6 convergent views taken with a calibrated high-resolution camera (Canon EOS 5D with 12.8Mpix.) and software PhotoModeler. The estimated accuracy of the 3D model is better than 0.1mm. The omnidirectional images were acquired using a catadioptric system with a hyperbolic mirror[3]. We computed the projection matrix P_{cata} from a total of 144 3D-2D correspondences and extracted the intrinsic and extrinsic parameters as explained in Section 3. From simulations, we observed that we have better and faster estimations if the 3D-2D correspondences are in the same order of magnitude. So 3D points are given in meters and 2D points are normalized in all the experiments. A second evaluation of the calibration accuracy is performed by a Structure from Motion experiment from two omnidirectional images.

6.1 Intrinsic Parameters

The first experiment is focused on obtaining the intrinsic parameters from P_{cata} to get initial estimates of these values. As mentioned previously, we do not compute tilting and distortion parameters from P_{cata} but it is possible to include them in the non-linear optimization. From simulations we observed that we can compute either the tangential distortion or the tilting parameters which are coupled and can not be separated. We tested which one of these (tangential distortion and tilting) can deal better with the intrinsic parameter estimation. Table 3 shows a comparison of the estimations performed with these two options. The real values given in the table were computed using the calibration data of the perspective camera (previously calibrated) and the mirror parameters (provided by the manufacturer).

	Real	Using distortion	Using tilting
f	279.84	297.24	306.11
c_x	531.83	528.08	552.75
c_y	407.98	406.28	427.89
ξ	0.96	0.86	0.93
RMSE	0	0.34	0.27

Table 3 Parameters estimated using either tangential distortion or tilting angles.

	Theoretic	P_{cata} approach	[Mei and Rives,07]
f	279.84	297.24	298.65
ξ	0.96	0.86	0.72
c_x	531.83	528.02	528.15
c_y	407.98	406.28	403.39

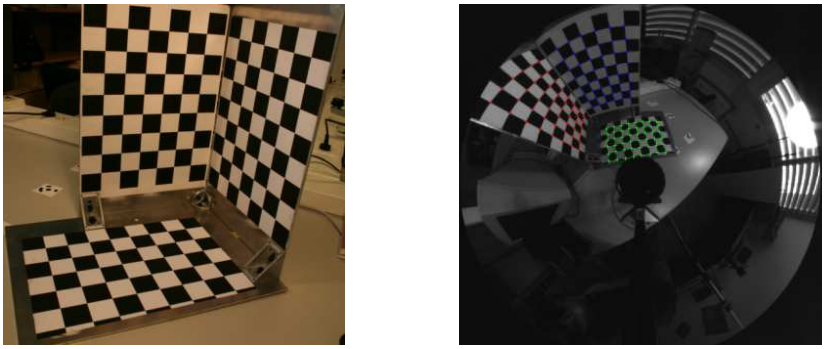
Table 4 Comparison between our method and Mei's.

Catadioptric camera calibration using tilting gives a better RMSE but the intrinsic values obtained are far from the real ones. Estimation using distortion parameters increase slightly the RMSE but the intrinsic parameters are close to the real ones, except for ξ but this error can be attached to the configuration of the system (the optical center of the perspective camera may not be exactly located at the other focal point of the hyperbola describing the mirror) and not to the model.

After these results, we decided to use tangential distortion because it gives better results and depicts better the real catadioptric system.

In order to verify our approach we compare our intrinsic parameter estimates to the ones obtained by Mei's [20] calibration approach (Table 4). As we can see neither Mei's approach nor P_{cata} approach can estimate the theoretic f and ξ parameters but they give a good estimation to c_x and c_y. Mei computes the initial values directly from the inner circle of the omnidirectional image and using information given by the user. Our approach computes all the initial values from P_{cata} in closed form.

[3] Neovision H3S with XCD-X710 SONY camera



(a) (b)

Fig. 5 (a) 3D pattern, (b) Omnidirectional image of the 3D pattern (1024×768 pixels).

6.2 Extrinsic Parameters

To obtain ground truth extrinsic parameters we have taken two additional images with the high resolution camera, observing the omnidirectional camera and the pattern. These images are added to the ones used to measure the 3D pattern. From this set of images the orientation and translation of the camera with respect to the pattern are computed. Location of the focal point was difficult since the points are not easy to identify in the images and indeed inside the mirror.

We performed experiments with 3 different camera locations. Table 5 shows the rotations and translations obtained from these experiments. Using PhotoModeler software we were just able to compute the direction of the z-axis but not the rotation around it. So we just show rotation estimations for the x and y axis. We can observe that the extrinsic parameter estimation is performed with a good accuracy having an average error of 0.0096 radians for rotations and 0.0022 meters for translations.

6.3 Structure from Motion

The second experiment to evaluate the accuracy of our approach consists of obtaining the Structure and Motion (SfM) from two omnidirectional images observing the 3D pattern. Fig. 6(a) shows the 3D pattern with the angles between the planes composing it. Fig. 6(b) depicts the configuration used to perform the SfM experiment. Using the internal calibration provided by our method we compute the corresponding 3D rays from each omnidirectional image. We use these correspondences of 3D rays to compute the essential matrix E which relates them. From this matrix we compute two projection matrices $\mathsf{P}_1 = [\mathtt{I}|\mathbf{0}]$ and $\mathsf{P}_2 = [\mathsf{R}|\mathbf{t}]$. Then, with these projection matrices and the 3D rays as input for a linear triangulation method [5] we compute an initial 3D reconstruction. Both the 3D reconstruction and the camera location are later refined by a non-linear optimization process. We use 144 points which were extracted manually from the images. We measure the average error between the real 3D points and their estimations and the angle between the planes. We use as ground truth the data computed by the photogrammetric software. The angles between the planes as depicted in Fig. 6(a) are $\alpha = 90.06°$, $\beta = 89.60°$ and $\gamma = 90.54°$. The estimated values are $\alpha = 89.22°$, $\beta = 90.55°$ and $\gamma = 89.73°$. We have an average error of 0.86°. We also measure the accuracy of the 3D points. The dimensions of the planar grids used in the 3D pattern are 210mm $\times$ 294mm. We compute the Euclidean distance between each reconstructed point and the ground truth. The average error is 1.03mm.

7 Conclusions

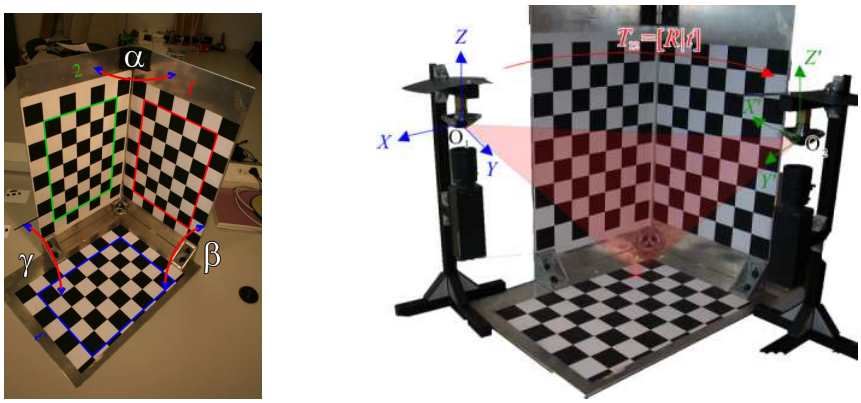
We presented a calibration technique based on the sphere camera model which is able to represent every single-viewpoint catadioptric system. We employed a generic 6×10 projection matrix, which uses *lifted coordinates* for image and 3D points. We estimated this projection matrix using 3D-2D correspondences (from a 3D calibration pattern), and decomposed it to obtain intrinsic and extrinsic parameters. We used this parameter estimation followed by a non-linear optimization to calibrate various types of cameras. We tested this method both with simulations and real images. We also present a Structure from Motion experiment to test the accuracy of our calibration method.

8 Acknowledgements

The authors are grateful for researcher exchange support of TUBITAK and Spanish MICINN. This work was also supported by project DPI2009-14664-C02-01.

	Experiment 1		Experiment 2		Experiment 3	
	Real	Estimated	Real	Estimated	Real	Estimated
R_x	-0.01	-0.02	-0.01	-0.003	-0.01	-0.002
R_y	0.02	0.02	0.02	0.01	0.02	0.03
R_z	—	—	—	—	—	—
t_x	0.39	0.39	0.39	0.39	0.39	0.38
t_y	0.21	0.21	0.33	0.33	0.23	0.23
t_z	-0.18	-0.18	-0.18	-0.18	-0.18	-0.18
RMSE	0.26		0.20		0.26	

Table 5 Rotation and translation of the camera with respect to the 3D pattern. Rotation angles are in radians. Translations are in meters. Real values were computed by the PhotoModeler software and a high resolution camera.



(a) (b)

Fig. 6 (a) 3D pattern with the angles between the planes. (b) SfM configuration.

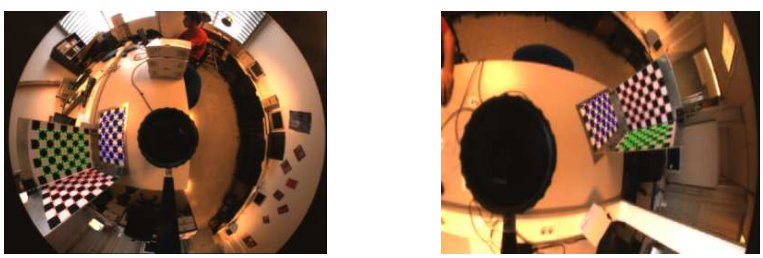


Fig. 7 Images used in the SfM experiment.

Peter Sturm acknowledges support by the French ANR project CAVIAR. João P. Barreto is grateful for support from the Portuguese Science Foundation through grant PTDC/ EEA-ACR/ 68887/ 2006.

References

1. Scotti, G., Marcenaro, L., Coelho, C., Selvaggi, F., Regazzoni, C.: Dual camera intelligent sensor for high definition 360 degrees surveillance. IEEE Proceedings on Vision, Image and Signal Processing. **152** (2005) 250–257
2. Nagahara, H., Yagi, Y., Yachida, M.: Wide field of view head mounted display for tele-presence with an omnidirectional image sensor. In: Conference on Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Volume 7. (2003) 86–86
3. Chahl, J., Srinivasan, M.: A complete panoramic vision system, incorporating imaging, ranging, and three dimensional navigation. In: IEEE Workshop on Omnidirectional Vision, 2000. Proceedings. (2000) 104–111
4. Lhuillier, M.: Toward flexible 3d modeling using a catadioptric camera. In: IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07. (2007) 1–8
5. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Second edn. Cambridge University Press, ISBN: 0521540518 (2004)
6. Baker, S., Nayar, S.K.: A theory of single-viewpoint catadioptric image formation. International Journal of Computer Vision **35** (1999) 175–196
7. Swaminathan, R., Grossberg, M., Nayar, S.: Non single-viewpoint catadioptric cameras. In: Technical Report, Department of Computer Science, Columbia University. (2001)
8. Derrien, S., Konolige, K.: Approximating a single viewpoint in panoramic imaging devices. In: Proceedings. IEEE Workshop on Omnidirectional Vision, 2000. (2000) 85–90
9. Faugeras, O.: Three-Dimensional Computer Vision (Artificial Intelligence). The MIT Press (1993)
10. Kang, S.B.: Catadioptric self-calibration. In: IEEE Conference on Computer Vision and Pattern Recognition, 2000. Volume 1. (2000) 201–207

11. Svoboda, T., Pajdla, T.: Epipolar geometry for central catadioptric cameras. Int. J. Comput. Vision **49** (2002) 23–37
12. Scaramuzza, D., Martinelli, A., Siegwart, R.: A toolbox for easily calibrating omnidirectional cameras. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006. (2006) 5695–5701
13. Toepfer, C., Ehlgen, T.: A unifying omnidirectional camera model and its applications. In: Proceedings of the 11th International Conference on Computer Vision, ICCV 2007. (2007) 1–5
14. Geyer, C., Daniilidis, K.: A unifying theory for central panoramic systems and practical applications. In: Proceedings of the 6th European Conference on Computer Vision-Part II, London, UK (2000) 445–461
15. Geyer, C., Daniilidis, K.: Paracatadioptric camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence **24** (2002) 687–695
16. Orghidan, R., Salvi, J., Mouaddib, E.M.: Calibration of a structured light-based stereo catadioptric sensor. In: Conference on Computer Vision and Pattern Recognition Workshop, 2003. CVPRW '03. Volume 7. (2003) 70–70
17. Cauchois, C., Brassart, E., Drocourt, C., Vasseur, P.: Calibration of the omnidirectional vision sensor: Syclop. In: IEEE International Conference on Robotics and Automation, 1999. Proceedings. 1999. Volume 2. (1999) 1287–1292 vol.2
18. Kannala, J., Brandt, S.: A generic camera calibration method for fish-eye lenses. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. Volume 1. (2004) 10–13 Vol.1
19. Tardif, J.P., Sturm, P., Roy, S.: Self-calibration of a general radially symmetric distortion model. In: Proceedings of the 9th European Conference on Computer Vision. Volume 4 of Lecture Notes in Computer Science., Springer (2006) 186–199
20. Mei, C., Rives, P.: Single view point omnidirectional camera calibration from planar grids. In: IEEE International Conference on Robotics and Automation, 2007. (2007) 3945–3950
21. Svoboda, T., Pajdla, T.: Epipolar geometry for central catadioptric cameras. International Journal of Computer Vision **49** (2002) 23–37
22. Geyer, C., Daniilidis, K.: Structure and motion from uncalibrated catadioptric views. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001. Volume 1. (2001) I–279–I–286 vol.1
23. Geyer, C., Daniilidis, K.: Properties of the catadioptric fundamental matrix. In: Proceedings of the 7th European Conference on Computer Vision. (2002) 140–154
24. Sturm, P.: Mixing catadioptric and perspective cameras. In: Proceedings. Third Workshop on Omnidirectional Vision, 2002. (2002) 37–44
25. Barreto, J.: A unifying geometric representation for central projection systems. Computer Vision and Image Understanding **103** (2006) 208–217
26. Barreto, J., Daniilidis, K.: Epipolar geometry of central projection systems using veronese maps. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006. Volume 1. (2006) 1258–1265
27. Sturm, P., Barreto, J.: General imaging geometry for central catadioptric cameras. In: Proceedings of the 10th European Conference on Computer Vision. Volume 4., Springer (2008) 609–622
28. Geyer, C., Daniilidis, K.: A unifying theory for central panoramic systems and practical applications. In: Proceedings of the 6th European Conference on Computer Vision-Part II, London, UK, Springer-Verlag (2000) 445–461
29. Bastanlar, Y., Puig, L., Sturm, P., Guerrero, J., Barreto, J.: Dlt-like calibration of central catadioptric cameras. In: Proceedings of the Eight Workshop on Omnidirectional Vision, 2008. (2008)
30. Abdel-Aziz, Y., Karara, H.: Direct linear transformation from comparator coordinates into object space coordinates in close-range photogrammetry. In: Symposium on Close-Range Photogrammetry. (1971) 1–18
31. Horn, R., Johnson, C.: Topics in Matrix Analysis. 2nd edn. Cambridge Univ. Press (1991)
32. Horn, R., Johnson, C.: Matrix Analysis. Cambridge Univ. Press (1985)
33. Heikkila, J., Silven, O.: A four-step camera calibration procedure with implicit image correction. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997. (1997) 1106–1112
34. Buchanan, T.: The twisted cubic and camera calibration. Computer Vision, Graphics and Image Processing **42** (1988) 130–132
35. Barreto, J.P., Araújo, H.: Geometric properties of central catadioptric line images and their application in calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 1327–1333