

# Calibration of Cameras with Radially Symmetric Distortion

Jean-Philippe Tardif, Peter Sturm, *Member, IEEE Computer Society*,  
Martin Trudeau, and Sébastien Roy

**Abstract**—We present algorithms for plane-based calibration of general radially distorted cameras. By this, we understand cameras that have a distortion center and an optical axis such that the projection rays of pixels lying on a circle centered on the distortion center form a right viewing cone centered on the optical axis. The camera is said to have a single viewpoint (SVP) if all such viewing cones have the same apex (the optical center); otherwise, we speak of NSVP cases. This model encompasses the classical radial distortion model [5], fisheyes, and most central or noncentral catadioptric cameras. Calibration consists in the estimation of the distortion center, the opening angles of all viewing cones, and their optical centers. We present two approaches of computing a full calibration from dense correspondences of a single or multiple planes with known euclidean structure. The first one is based on a geometric constraint linking viewing cones and their intersections with the calibration plane (conic sections). The second approach is a homography-based method. Experiments using simulated and a broad variety of real cameras show great stability. Furthermore, we provide a comparison with Hartley-Kang's algorithm [12], which, however, cannot handle such a broad variety of camera configurations, showing similar performance.

**Index Terms**—Calibration, omnidirectional vision, fisheye, catadioptric camera.

## 1 INTRODUCTION

IN the last few years, we have seen an increasing interest in nonconventional cameras and projection models, going beyond affine or perspective projection. There exists a large diversity of camera models, many of which specific to certain types of projections [14], [17], others applicable to families of cameras such as central catadioptric systems [2], [3], [7], [9]. All of these models are described by a few intrinsic parameters, much like the classical pinhole model, possibly enhanced with radial or decentering distortion coefficients. Calibration methods exist for all these models, and they are usually tailor-made for them, i.e., they cannot be used for any other projection model [15]. Several works address the calibration problem from an opposite point of view by adopting a very generic imaging model that incorporates most commonly used cameras [6], [10], [11], [20], [22].

In the most general case, cameras are modeled by attributing an individual projection ray to each pixel. Such a model is highly expressive, but it is difficult to obtain a stable calibration of cameras with it, at least with few input images taken in an uncontrolled environment. Finally,

several researchers have proposed a compromise between parametric and such generic models. They assume radial symmetry around the distortion center (often considered as coinciding with the principal point) but with a general distortion function [12], [24], [25], [26], [28], [29].

In this paper, we use a model of this category which we find sufficiently general to model many common types of cameras. By having fewer parameters than the fully generic model, calibration remains easy and stable. It encompasses many common camera models, such as pinhole (modulo aspect ratio and skew), the classical polynomial radial distortion model, fisheyes, or any catadioptric system whose mirror is a surface of revolution, and for which the optical axis of the perspective (or affine) camera looking at it is aligned with the mirror's revolution axis. We describe our model in the following and refer to Fig. 1 for an illustration. Cameras are modeled using the notion of a **distortion center** in the image whose back-projection yields the **optical axis** in 3D. For cameras with **radially symmetric distortion**, the projection rays associated with pixels lying on the same **distortion circle** centered on the distortion center lie on a right **viewing cone** centered on the optical axis. In the following, we denote by  $d$  the radius of a distortion circle; for ease of expression, we also speak of distortion circle  $d$ , meaning the distortion circle of radius  $d$ .

For a perspective camera with focal length  $f$ , the opening angle of a distortion circle's viewing cone is given by  $2 \arctan \frac{d}{f}$ . Hence, the opening angles of all viewing cones are "dictated" by the camera's focal length. In our model, this is generalized: In the most general case, one may not assume any relation between opening angles of different viewing cones. In other words, we may have one individual focal length  $f_d$  per distortion circle  $d$ . We may define a **focal length function**  $f_d = f(d)$ . In practice, we handle such a general model in two different ways: by assuming a polynomial form for the focal length function or by subsampling it and

- J.-P. Tardif is with the Department of Computer and Information Science, University of Pennsylvania, Levine Hall L402, 3330 Walnut Street, Philadelphia, PA 19104-6228. E-mail: tardif@seas.upenn.edu.
- P. Sturm is with INRIA Grenoble-Rhône-Alpes, 655 Avenue de l'Europe, 38330 Montbonnot, France. E-mail: Peter.Sturm@inrialpes.fr.
- M. Trudeau is with Mercer, 1981 McGill College Suite 800, Montréal, QC, H3A 3T5, Canada. E-mail: martin.trudeau@mercercor.com.
- S. Roy is with the Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, 2920 chemin de la Tour, Montréal QC, H3T 1J4, Canada. E-mail: roys@iro.umontreal.ca.

Manuscript received 14 Aug. 2007; revised 6 Apr. 2008; accepted 18 July 2008; published online 30 July 2008.

Recommended for acceptance by F. Kahl.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-2007-08-0505.

Digital Object Identifier no. 10.1109/TPAMI.2008.202.

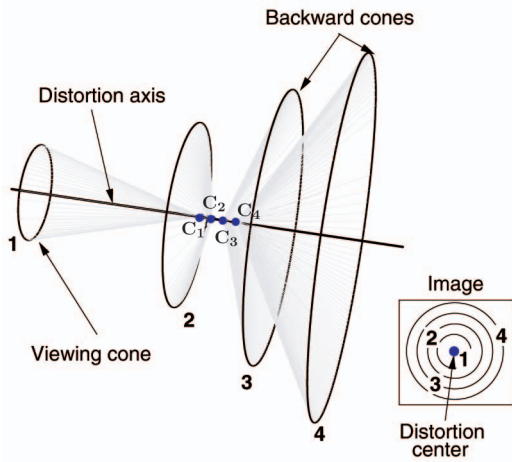


Fig. 1. Our camera model in the case of a noncentral projection (see text for explanations). The inlayed illustrations show the distortion center (in blue) and four distortion circles and their corresponding viewing cone and optical center.

estimating/using one focal length per discrete sample of  $d$ . We also consider its reciprocal, the **distortion function** that maps the opening angle of a viewing cone to the radius of the associated distortion circle.

Let us introduce a few additional notations. A line spanned by a pixel and the distortion center is called a **radial line** and the plane spanned by the projection ray associated with the pixel and the optical axis is called a **radial plane**. In our camera model, we specify that angles between radial lines and associated radial planes are identical (otherwise, it would be a very uncommon camera). Our model comprises **central** cameras [single viewpoint (SVP)], where all viewing cones have the same apex (the **optical center**), but also **noncentral** ones (NSVP), for which the viewing cones' apexes lie anywhere on the optical axis. In the latter case, we may speak of one optical center per viewing cone. For convenience, these terms are summarized in Table 1.

**Problem statement.** We want to calibrate cameras based on the above model from one or several images of a calibration plane in unknown positions. Calibration consists in estimating the camera's focal length function or, equivalently, its distortion function. Further, for the NSVP case, we have to estimate the optical centers associated with all distortion circles.

The problem of calibration from a planar scene of known geometry has been studied intensively in computer vision.

It is widely accepted that one seeks a solution that minimizes the sum of reprojection errors of the points on the calibration planes. This being a nonconvex function, iterative methods must be used. Besides the problem of convergence to a local minimum, they require some estimate for the internal parameters of the camera, generally provided by the camera manufacturer [1], [13]. Both of these difficulties can be overcome by noniterative approaches relying on algebraic error functions as long as they provide good accuracy. Then, these results can be refined iteratively using a geometric error.

**Contributions.** Our contributions take the form of two noniterative calibration algorithms. Their input is the euclidean structure of the plane(s) and a dense or sparse matching between the plane(s) and the camera image. The first approach is based on algebraic constraints relating the projection of the points on the calibration plane on the image. It is based on the observation that each distortion circle and the associated viewing cone can be considered as an individual perspective camera. Hence, the projection of a calibration plane to the image, when reduced to a single distortion circle, can be expressed by a homography and we can readily apply plane-based calibration algorithms designed for perspective cameras. Our second approach is based on a detailed geometric analysis. Consider one image of a calibration plane; each viewing cone of the camera intersects the calibration plane in a conic, which we call a **calibration conic**. We describe geometric constraints relating calibration conics to the orientation and position of the camera as well as its intrinsic parameters; these are at the basis of our second calibration approach. Our first approach usually performs best in practice. Furthermore, it can accurately estimate the projection model for a noncentral camera. Based on our experimental results, we argue that, in the case of a general model such as the one we propose, an approach directly estimating a noncentral projection gives better accuracy than an approach based on: 1) estimating an approximated central projection and 2) iterative refinement using a noncentral model [15].

**Organization.** A geometric study of our model is presented in Section 2. Our first calibration approach, akin to plane-based calibration of perspective cameras, is described in Section 3, followed by our second, more geometric, approach in Section 4. Our algorithms assume a known position of the distortion center, but we also show how to estimate it, using repeated calibrations for different candidates, cf. Section 5. Several practical issues and experimental results, including a comparison with

TABLE 1  
Summary of the Terms Related to Our Camera Model

Terms	Description
Optical axis	A line perpendicular to the image plane
Distortion center	The intersection between the image plane and the optical axis
Distortion circle	A circle in the image centered in the distortion center
Viewing cone	Back-projection of a distortion circle; a right cone centered in the optical axis
Calibration conic	Intersection of a viewing cone and a calibration plane
Radial line	A line passing through the distortion center
Radial plane	Back-projection of a radial line
Central projection	All viewing cones have identical apex, the optical center
Non-central projection	The viewing cones only share the same (optical) axis

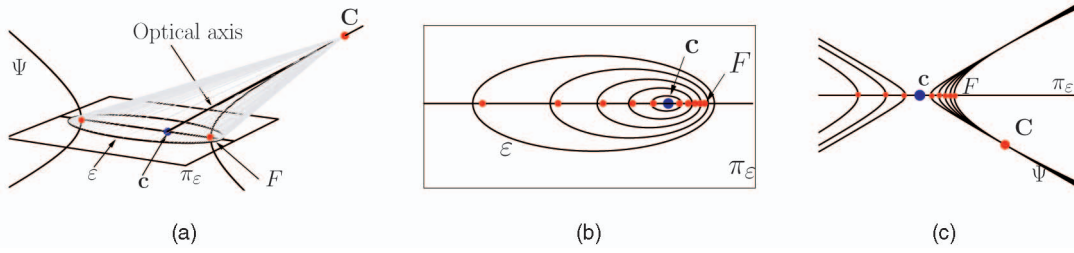


Fig. 2. Illustrations of the geometry of viewing cones, calibration conics (here ellipses), and location of optical center in the SVP case. (a) Complete illustration for one viewing cone. (b) View of the calibration plane, showing many cones' calibration ellipses. Note that their major axes are collinear. (c) Side view of the viewpoint conics (hyperbolas in this case) associated with many calibration conics.

Hartley-Kang's (HK) methods [12], are presented in Sections 6 and 7, respectively. Finally, we conclude in Section 8.

## 2 GEOMETRY

In this section, we present geometric constraints for our camera model. Readers interested in implementing the homography-based approach may wish to skip to Sections 2.3 and 3.

### 2.1 One Distortion Circle in One Image of a Calibration Plane

Let us consider one distortion circle in the image plane. Its associated viewing cone cuts the calibration plane in a **calibration conic**. From dense matches between image and calibration plane, we can compute this calibration conic (see Section 6 for more details on the dense matching). This conic can be either an ellipse or a hyperbola (the parabola is only a theoretically interesting case). If we knew the position of the camera's optical center relative to the calibration plane, then we could directly compute the viewing cone of our distortion circle, i.e., the cone that has the optical center as apex and that contains the above calibration conic. As mentioned, that cone has several useful properties: Its axis is the camera's optical axis and it is a right cone, i.e., rotationally symmetric with respect to its axis. From the cone, the focal length of the considered distortion circle can be easily computed: The cone's opening angle is equal to the field of view.

In practice, we do not know the optical center's position relative to the calibration plane. In the following, we show geometrical relations between the calibration conic, the optical center, and the optical axis of the camera, which allow to compute the optical center. Recall that, when talking about optical center, we mean the optical center per distortion circle; they all lie on the optical axis and, in the SVP case, they are identical.

Without loss of generality, we assume that the calibration plane is the plane  $Z = 0$  and that the calibration conic  $\varepsilon$  is given by the matrix  $\varepsilon \propto \text{diag}(a, b, -1)$  with  $|b| \geq a > 0$  ( $\propto$  represents equality up to scale), i.e., the  $X$ -axis is the conic's major axis. The type of the calibration conic  $\varepsilon$  depends on  $a$  and  $b$  as follows:

$$\begin{cases} b \geq a > 0 & \text{ellipse} \\ b < 0 \text{ and } |b| > a > 0 & \text{hyperbola.} \end{cases} \quad (1)$$

Our aim is to provide constraints on the position of the optical center, as well as on the orientation of the optical

axis, from this conic. We first do this for the case of a calibration ellipse, then for the case of a hyperbola.

**The case of calibration ellipses.** Let us first state a well-known result. Consider a right cone whose apex is a point with real coordinates, and its intersection with a plane. For now, we assume that the intersection is an ellipse (the case of the hyperbola will be discussed later). It is easy to prove that the orthogonal projection of the cone's apex onto the plane lies on the ellipse's major axis (cf. Fig. 2a). This implies that the cone's apex lies in the plane that is orthogonal to the ellipse's supporting plane and that contains its major axis.

For our problem, this means that the optical center must lie in the plane  $Y = 0$  (since the ellipse lies in plane  $Z = 0$  and has the  $X$ -axis as major axis). We may further constrain its position  $\mathbf{C} = (X, 0, Z, 1)^T$ , as follows [4]. The cone with  $\mathbf{C}$  as apex and that contains the calibration ellipse  $\varepsilon$  is given by

$$\Lambda \propto \begin{bmatrix} aZ^2 & 0 & -aXZ & 0 \\ 0 & bZ^2 & 0 & 0 \\ -aXZ & 0 & aX^2 - 1 & Z \\ 0 & 0 & Z & -Z^2 \end{bmatrix}.$$

For this cone to be a right one, its upper left  $3 \times 3$  matrix  $\bar{\Lambda}$  must have a double eigenvalue. The three eigenvalues are

$$bZ^2, \frac{a(X^2 + Z^2) - 1 \pm \sqrt{4aZ^2 + (1 - a(X^2 + Z^2))^2}}{2}. \quad (2)$$

The second and third eigenvalues cannot be equal for real values of  $X$  and  $Z$  (besides, in the trivial case,  $X = Z = 0$ , which is excluded since it would correspond to the optical center lying in the calibration plane). The first eigenvalue is equal to the third one if  $Z = 0$  (excluded for the same reason as above) and to the second one if

$$a b X^2 + b(a - b) Z^2 + (a - b) = 0. \quad (3)$$

This equation tells us that the optical center lies on a **viewpoint conic** given by the following matrix and the associated equation:

$$\Psi \propto \begin{bmatrix} a & b & 0 \\ 0 & b(a - b) & 0 \\ 0 & 0 & a - b \end{bmatrix}, \quad (X \ Z \ 1) \Psi \begin{pmatrix} X \\ Z \\ 1 \end{pmatrix} = 0. \quad (4)$$

This is a hyperbola, due to  $b \geq a > 0$  (cf. (1)); it is sketched in Fig. 2a. Furthermore, its asymptotes correspond to the directions of the two circular cylinders that contain the calibration ellipse.



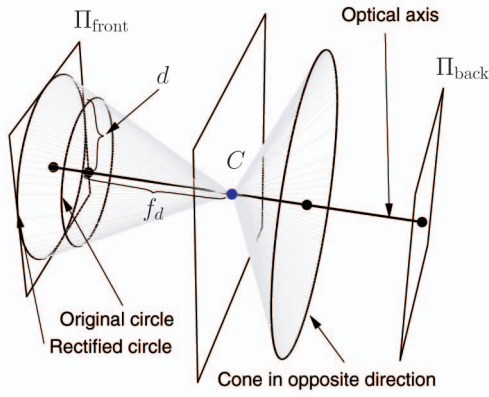


Fig. 3. Viewing cones can also be seen as individual perspective cameras with different focal length. A rectified image can be obtained by projecting the distortion circles (which lie in different planes) on one plane  $\Pi_{\text{front}}$  (or  $\Pi_{\text{back}}$  for a field of view larger than  $180^\circ$ ).

Let us now consider the orientation of the optical axis. Due to (3), we have an optical center given by

$$Z = \pm \sqrt{\frac{a b X^2 + a - b}{b(b - a)}}. \quad (5)$$

Here, we exclude the case  $a = b$ , which would correspond to the camera looking straight at the calibration plane.

The direction of the cone's axis is given by the eigenvector associated with the single eigenvalue of  $\bar{\Lambda}$ , i.e., the third one, augmented with a homogeneous coordinate 0:

$$\begin{pmatrix} \pm \sqrt{b(b - a)(a b X^2 + a - b)} & 0 & a b X & 0 \end{pmatrix}^\top. \quad (6)$$

We now show that the cone's axis is identical to the tangent of the hyperbola  $\Psi$  in the optical center  $C$ , which is given by the line (in the plane  $Y = 0$ )

$$\Psi \begin{pmatrix} X \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} a b X \\ \mp \sqrt{b(b - a)(a b X^2 + a - b)} \\ a - b \end{pmatrix}.$$

Its point at infinity is (still in the plane  $Y = 0$ )

$$\begin{pmatrix} \pm \sqrt{b(b - a)(a b X^2 + a - b)} & a b X & 0 \end{pmatrix}^\top,$$

i.e., it is identical to the point given in (6). Hence, for an optical center on  $\Psi$ , the optical axis is directly given by the associated tangent.

Coming back to (2), the eigenvalues of the cone can be used to compute the focal length  $f_d$  associated to a distortion circle of radius  $d$  (see Fig. 3). The value  $f_d/d$  is the tangent of half the opening angle of the viewing cone. Furthermore, it can be shown that  $(f_d/d)^2$  is equal to the negative of the ratio of the double and the single eigenvalue of  $\bar{\Lambda}$ , i.e., the negative of the ratio of the first and the third eigenvalues given in (2). Using (5), we obtain

$$\left(\frac{f_d}{d}\right)^2 = -\frac{b(abX^2 + a - b)}{a(a - b)}. \quad (7)$$

This relation will prove useful in Section 4.2.

**The case of calibration hyperbolas.** The case where the intersection between a cone and the calibration plane yields a hyperbola is accounted for automatically by the previous formulation. All of the above findings hold here, with the exception that the viewpoint conic  $\Psi$  is an ellipse. This case typically occurs with very wide-angle cameras or when the angle between the camera's optical axis and the calibration plane is large.

## 2.2 Multiple Distortion Circles

So far, we have shown that, for an individual distortion circle, the associated viewing cone can be determined from the associated calibration conic, up to one degree of freedom (location on the viewpoint conic  $\Psi$  and associated orientation of the optical axis). We now show how to get a unique solution, when considering several distortion circles simultaneously. Let us first note that calibration conics corresponding to different distortion circles are not independent: their major axes are collinear (cf. Fig. 2b), even in the NSVP case. Their centers are not identical, however, unless they are all circles, which can only happen when the camera looks straight at the calibration plane.

Let  $\Psi_d$  be the viewpoint conic associated with distortion circle (of radius  $d$ ), all  $\Psi_d$  being given in the same coordinate frame. In the case of an SVP camera, the optical center must lie on all these conics. Furthermore, the optical axis is tangent to all of them. This implies that all conics touch each other (have a double intersection point) in the optical center. This is illustrated in Fig. 2c. A naive algorithm would compute the viewpoint conic for all calibration conics and seek their single intersection/contact point. However, very little noise can cause two viewpoint conics to have no real intersection point at all, instead of a double one.

Interestingly, this constraint gives a geometric explanation of the ambiguity, or correlation, between camera position and focal length that often occurs when calibrating a camera from a single view. Considering perfectly recovered viewpoint conics (such as in Fig. 2c), the ambiguity is observed as the area where the viewpoint conics are "very close" to each other. Clearly, a very low perturbation of the curves can result in significant but correlated errors on the optical center and the focal length.

In the NSVP case, a separate optical center corresponds to each distortion circle and viewing cone. Hence, the viewpoint conics will not have a single contact point anymore. However, the optical axis is shared by all viewing cones. Hence, it is the single (in general) line that is tangent to all viewpoint conics. Furthermore, each contact point with the optical axis is the associated optical center.

## 2.3 Model Parameterization

It is possible to consider the distortion circles and associated viewing cones as individual *perspective cameras*, with different focal lengths but identical principal points [24], [25], [26]. In the SVP case, extrinsic parameters of all these cameras are identical, whereas, in the NSVP case, they all share the same orientation and the optical centers are merely displaced along the optical axis.

Let us consider the distortion circle of radius  $d$  and one image of a calibration plane. From point correspondences between pixels on this circle and points on the calibration plane (on the calibration conic), we can compute a plane-to-image homography  $H_d$ . For simplicity, let us assume that

image coordinates have been translated to put the distortion center at the origin. The homography can then be decomposed such that

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \propto H_d \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = K_d R \begin{bmatrix} 1 & 0 \\ 0 & 1 & -\mathbf{t} + t_d \mathbf{r}_3 \\ 0 & 0 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad (8)$$

where  $\propto$  means equality up to scale,  $(x, y)$  is a calibration point,  $(u, v)$  is a pixel on the distortion circle, and  $R$  and  $\mathbf{t}$  are the rotation matrix and translation vector representing camera pose, respectively (same for all  $d$ ). The scalar  $t_d$  allows us to model translational displacement of individual viewing cones along the optical axis (given by  $\mathbf{r}_3^\top$ , the third row of  $R$ ), which is needed for NSVP cameras. For SVP cameras,  $t_d$  is set to 0 for all  $d$ .

As for  $K_d$ , it is a calibration matrix<sup>1</sup>  $\text{diag}(f_d, f_d, 1)$ , where  $f_d$  is the focal length associated with the considered distortion circle. We may interpret the relation between  $d$  and  $f_d$  as a distortion function applied to a perspective camera whose undistorted image plane is  $\pi_{\text{front}}$  (cf. Fig. 3).

Note that this parameterization only accounts for viewing cones with field of view smaller than  $180^\circ$ . Larger fields of view can be modeled by adding a reflection to the rotational component of the pose,  $R' = \text{diag}(1, 1, -1)R$ , and a corresponding image plane  $\pi_{\text{back}}$ . Equivalently, one may describe these cones as cameras with negative focal length as presented in [25].

## 2.4 Overview of the Approaches

We briefly describe our two calibration approaches in terms of the concepts previously introduced in this section. Both proceed in two steps. In the homography-based algorithm (Section 3), the optical axis for each view of the calibration plane is first estimated, each one up to four possibilities. The correct choice is made at the second step, which consists of simultaneously estimating the position of the camera on the optical axis for all views and the opening angle (equivalently the focal length) of all the cones. The second approach is based on the above introduced geometric constraints, although ultimately some algebraic manipulations are required to avoid an iterative solution. We discuss two solutions for estimating the pose of the camera for each view, using the calibration conics. Then, the opening angles of the cones are estimated by refitting the calibration conics using a parameterization enforcing the fit to be consistent with the pose of the camera.

## 3 HOMOGRAPHY-BASED CALIBRATION

### 3.1 Unconstrained Approach

In [24], an approach based on recovering the *Image of the Absolute Conic*, associated to each distortion circle, from one or many images of a calibration plane, is introduced. Once the internal parameters are recovered, the camera external parameters are estimated using the approach presented in [19], [21], [31]. The weakness of this approach is that, since the calibration is individually performed on each distortion circle, it does not enforce SVP or NSVP constraints. However, it is useful for estimating the distortion center

as described in Section 5. This algorithm will be referred to as “IAC.”

### 3.2 Constrained Approach

We present another homography-based approach that can enforce the constraints directly. It is related to HK approach, but it also handles noncentral omnidirectional cameras.

**SVP case, one image.** In the SVP case,  $t_d$  is zero for all  $d$ . It follows that

$$K_d^1 \mathbf{q} \propto \begin{bmatrix} 1 & & \\ & 1 & \\ & & f_d \end{bmatrix} \mathbf{q} \propto R \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 & -\mathbf{t} \\ 0 & 0 \end{bmatrix}}_M \mathbf{Q} \quad (9)$$

holds up to scale. If we divide the first by the second coordinates of both sides, we obtain an equation that is independent from the distortion circle  $d$ , i.e.,

$$\frac{q_1}{q_2} = \frac{(M\mathbf{Q})_1}{(M\mathbf{Q})_2}. \quad (10)$$

We finally obtain a linear equation on the six coefficients in the first two rows of the pose matrix  $M$ , i.e.,

$$\begin{aligned} q_1(M_{21}Q_1 + M_{22}Q_2 + M_{23}Q_3) - \\ q_2(M_{11}Q_1 + M_{12}Q_2 + M_{13}Q_3) = 0. \end{aligned} \quad (11)$$

The above equation being linear homogeneous, we may use all point correspondences between the calibration plane and the image to estimate the first two rows of  $M$ , up to scale. Until here, our algorithm is very similar to [12], but the following differs significantly. The pose, i.e.,  $R$  and  $\mathbf{t}$ , can be partially estimated from  $M$ . There exists a scalar  $\lambda$  such that

$$\begin{bmatrix} R_{11} & R_{12} & -\mathbf{r}_1^\top \mathbf{t} \\ R_{21} & R_{22} & -\mathbf{r}_2^\top \mathbf{t} \end{bmatrix} = \lambda \bar{M}, \quad (12)$$

where  $\mathbf{r}_i^\top$  is the  $i$ th row of  $R$  and  $\bar{M}$  is the upper  $2 \times 3$  part of  $M$ . This leads to the following observations:

- The rotation matrix  $R$  can be estimated up to four solutions, from the left  $2 \times 2$  part of  $\bar{M}$ . They differ by a choice of sign for columns and rows of the solution: If  $R$  is one correct solution, then the other three are given by  $DR$ ,  $RD$ ,  $DRD$ , with  $D = \text{diag}(-1, -1, 1)$ . Distinguishing the valid rotation matrix will be done later when estimating the full camera position and internal parameters.
- The intersection  $\mathbf{t}_0$  between the optical axis of the camera and the calibration plane can be recovered as the right nullspace of  $\bar{M}$  since it is the point on the plane projecting onto the distortion center (the origin). Hence, the translation vector  $\mathbf{t}$  can be estimated up to one degree of freedom, a displacement  $\mu$  along the optical axis, i.e.,

$$\mathbf{t} = \mathbf{t}_0 + \mu \mathbf{r}_3. \quad (13)$$

Estimating  $\mu$  can be done simultaneously with estimating the focal lengths of all distortion circles. By inserting the solution for  $\mathbf{t}$  into (9), we obtain

1. As mentioned in Section 1, this model does not include a skew between pixel axes or an aspect ratio different from 1. Also, it assumes that the distortion center is at the principal point.

$$\mathbf{q} \propto \begin{bmatrix} f_d & & \\ & f_d & \\ & & 1 \end{bmatrix} \mathbf{R} \begin{bmatrix} 1 & 0 & \\ 0 & 1 & -\mathbf{t}_0 - \mu \mathbf{r}_3 \\ 0 & 0 & \end{bmatrix} \mathbf{Q}.$$

Due to the orthonormality of  $\mathbf{R}$ , we obtain

$$\mathbf{q} \propto \begin{bmatrix} f_d & & \\ & f_d & \\ & & 1 \end{bmatrix} \left( \underbrace{\mathbf{R} \begin{bmatrix} 1 & 0 & \\ 0 & 1 & -\mathbf{t}_0 \\ 0 & 0 & \end{bmatrix}}_{\mathbf{T}_0} - \begin{bmatrix} 0 & & \\ & 0 & \\ & & \mu \end{bmatrix} \right) \mathbf{Q}.$$

Let us denote:  $\mathbf{S} = \mathbf{R}\mathbf{T}_0\mathbf{Q}$ , which is a known vector. We thus can write the above equation as

$$\mathbf{q} \propto \begin{bmatrix} f_d S_1 \\ f_d S_2 \\ S_3 - \mu Q_3 \end{bmatrix} = \begin{bmatrix} S_1 & \\ & S_2 \\ & & Q_3 \end{bmatrix} \begin{pmatrix} f_d \\ f_d \\ S_3/Q_3 - \mu \end{pmatrix} \quad (14a)$$

and obtain the following set of linear equations on the unknowns  $f_d$  and  $\mu$ :

$$[\mathbf{q}]_{\times} \begin{bmatrix} S_1 & & \\ & S_2 & \\ & & -Q_3 \end{bmatrix} \begin{pmatrix} f_d \\ f_d \\ \mu - S_3/Q_3 \end{pmatrix} = \mathbf{0} \quad (14b)$$

or

$$\begin{bmatrix} q_3 S_2 & q_2 Q_3 \\ q_3 S_1 & q_1 Q_3 \\ q_1 S_2 - q_2 S_1 & 0 \end{bmatrix} \begin{pmatrix} f_d \\ \mu \end{pmatrix} = \begin{pmatrix} q_2 S_3 \\ q_1 S_3 \\ 0 \end{pmatrix}. \quad (14c)$$

Note that the equation system is nonhomogeneous, i.e., the solution for the  $f_d$  and  $\mu$  is computed exactly, not only up to scale. Also note that the third equation is useless: In the absence of noise, the term  $q_1 S_2 - q_2 S_1$  is zero and gives no constraint on  $f_d$ . With noise, however, the third equation admits random coefficients and using it was found to bias  $f_d$  toward small values. We thus only use the first two equations of (14c).

All point correspondences, from all distortion circles, can be used simultaneously: Each correspondence contributes to estimating  $\mu$  and to the focal length of the distortion circle it belongs to. Hence, overall, we have a linear system of size  $2n \times (D+1)$ , where  $n$  is the number of point correspondences and  $D$  is the number of distortion circles.

This assumed a known rotation. All rotation matrices among the four possibilities shown above give the same solution for  $f_d$  and  $\mu$ , up to different signs. Let  $d_0$  be distortion circle of smallest radius. Then, the correction rotation is the one with positive  $f_{d_0}$  and  $\mu$ .

**Using many images of a calibration plane.** The above equations can be used in a slightly modified form to simultaneously use many images. Let  $v$  be the index for each view out of  $V$ . Each associated  $2 \times 3$  partial homography  $\mathbf{M}_v$  can be computed using (11). Then,  $\mathbf{t}_{v0}$  and  $\mathbf{R}_v$  can be estimated individually from each of them. Finally, (14c) is extended naturally to account for many views by simultaneously solving for all displacements  $\mu_v$  and focal lengths  $f_d$ . The resulting equation system is of size  $2n \times (D+V)$ .

**NSVP case, many views.** In the NSVP case, the previous algorithm can be applied nearly without modification. Let us consider the form of the plane-to-image homography for an individual distortion circle, given in (8), for the NSVP

case. In comparison to the SVP case (9), there is an additional term  $t_d$  for the position of the displaced optical center on the optical axis:

$$\begin{aligned} \mathbf{q} &\propto \mathbf{H}_{vd}\mathbf{Q} = \mathbf{K}_d \mathbf{R}_v \begin{bmatrix} 1 & 0 & \\ 0 & 1 & -(\mathbf{t}_{v0} + (\mu_v + t_d)\mathbf{r}_{v3}) \\ 0 & 0 & \end{bmatrix} \mathbf{Q} \\ &= \mathbf{K}_d (\mathbf{R}_v \mathbf{T}_{v0} - \text{diag}(0, 0, \mu_v + t_d)) \mathbf{Q}. \end{aligned} \quad (15)$$

Note that  $t_d$  is the same for all views. Since  $\mathbf{K}_d$  is a diagonal matrix, the first two coordinates of the right-hand side do not depend on  $\mu_v$  and  $t_d$ . Hence, like in the SVP case, (11) can be used to estimate linearly the first two rows of the pose matrix  $\mathbf{M}_v$ , using all point correspondences, for all distortion circles. The pose parameters  $\mathbf{R}_v$  and  $\mathbf{t}_v$  can also be extracted in the same manner, with  $\mathbf{t}_v$  being obtained up to a displacement along the optical axis  $\mathbf{r}_{v3}$ .

We now consider how to estimate the focal lengths  $f_d$  and the optical center positions  $\mu_v + t_d$ . The equations are identical to those in the SVP case, with the difference that  $\mu_v$  has to be replaced by  $\mu_v + t_d$ , i.e., it is not the same for all distortion circles. The set of  $\mu_v$  and  $t_d$  is an overparameterization, since subtracting a value from all  $\mu_v$  and adding it to all  $t_d$  leaves (15) unchanged. Hence, we may set one of them to any fixed value. The equation system is thus of size  $2n \times (2D - 1 + V)$ .

**Polynomial model.** Calibrating a general model requires many data points to obtain precise and accurate results, i.e., to avoid overfitting. Otherwise, relying on a more restricted model may give better results. In [25], [26], [29], a polynomial model was used to represent the relationship between focal length and radius  $d$ . Our formulation can also be trivially modified to use polynomials for  $f_d$  as well as  $t_d$ . For NSVP cameras, it turns out that in practice, estimating the fully general model can be unstable because of the focal length-displacement ambiguity mentioned in Section 2.2. The result is that both  $f_d$  and  $t_d$  are not smooth functions for small radius values  $d$ . Instead of relying on smoothing terms (whose weights are difficult to set) to circumvent this, we prefer using polynomials.

The above method is summarized in Fig. 4 and referred to as “HB” in the following. In practice, this is the most accurate and flexible of the approaches presented here (see Section 7).

## 4 GEOMETRIC APPROACH

### 4.1 Solving the Calibration and the Pose

In the following, we present a second approach based on the geometric constraints discussed in Section 2. We first propose a naive calibration approach; it is not optimal though and a better one will be discussed in the next section. Recall the constraints relating calibration conics, viewpoint conics, and position of the camera illustrated in Fig. 2. In the SVP case, one easily deduces a calibration algorithm consisting in first estimating the optical center (relative to the calibration plane) as the 3D point that is closest on average to all viewpoint conics (see below). Then, (7) can be used to compute the focal lengths for all distortion circles. In the NSVP case, a possibility would be to compute the optical axis: the line  $\mathbf{L}$  that minimizes the sum of squared distances to the viewpoint conics. However, this is not very accurate as discussed below.



**INPUT:** Dense or Sparse plane-image correspondences

**ALGORITHM:**

For each calibration plane  $v$ :

- Estimate  $\bar{M}_v$  using (11) and all the correspondences.
- Obtain the rotation  $R_v$  up to 4 solutions using (12).
- Obtain  $\mathbf{t}_{0v}$  as the right nullspace of  $\bar{M}_v$ ,  
The four possible optical axes are given by (13).
- Disambiguate  $R_v$  and optical axis by estimating  $\forall d : f_d$  and  $\mu_v$  using (14c)

**DENSE CORRESPONDENCES:**

- Simultaneously estimate  $\forall d : f_d$  and  $\forall v : \mu_v$  combining (14c) for each view.

**SPARSE CORRESPONDENCES:**

- Apply the previous step replacing  $f_d$  by a polynomial.

Perform iterative refinement of the reprojection error.

Fig. 4. Complete algorithm for the HB calibration approach for an SVP camera.

**SVP case: Computing the closest point to the viewpoint conics.** This is the original approach presented in [24]. Computing the orthogonal distance of a point to a general conic requires solving a fourth degree polynomial [30]. Using this to compute the closest point to our set of viewpoint conics is not very practical. Instead, we iteratively minimize a cost function subject to constraints. The closest point  $\mathbf{q}$  is found by solving

$$\min_{\mathbf{q}, \mathbf{q}_d} \sum_d \text{dist}(\mathbf{q}, \mathbf{q}_d)^2, \quad \text{subject to } \mathbf{q}_d^\top \Psi_d \mathbf{q}_d = 0,$$

i.e., we also estimate one point per conic  $\Psi_d$  that will, after convergence, be the orthogonal projection of  $\mathbf{q}$  on  $\Psi_d$ . Since the function and constraints are polynomial, this problem can be optimized using an algorithm relying on Cylindrical Algebraic Decomposition (CAD), which guarantees a global minimum [18]. Such an algorithm is available through the *Minimize* function of *Mathematica*. In this work, we present another solution that gives much better result. Instead of proceeding into two steps, we directly use the fitted calibration conics to estimate the camera position, as described below.

## 4.2 A Formulation Enforcing the Constraints Directly

In our first experiments, we found the previous approach to be unstable, even though good results could be obtained in

some cases [24]. This is because the formulation has several drawbacks. First, although CAD optimization is algorithmic, it becomes computationally intractable when the number of viewpoint conics increases. Second, it appears that finding the closest point to the set of the recovered viewpoint conics is not the optimal criterion. As discussed above and shown in Fig. 5, when the noise is large, the shape and position of the viewpoint conics may become very perturbed.

For an SVP camera, a better formulation would directly enforce that the viewpoint conics all touch in one location and have the same tangent at this point. Then, CAD optimization could be avoided. Before giving our solution, we come back to our calibration conics. As mentioned above, they share an axis (in the absence of noise). We assume that their common axis can be estimated with high accuracy despite noise in the data, due to using many calibration conics. Once the conics' axis is estimated, it is convenient to change the coordinate system of the calibration plane such that the conics are aligned with the  $X$ -axis. Then, each one of them can be parameterized by its major and minor axis lengths  $b_d$  and  $a_d$ , as well as by a displacement  $k_d$  along the  $X$ -axis:

$$\varepsilon_d \propto \begin{bmatrix} a_d & 0 & -a_d k_d \\ 0 & b_d & 0 \\ -a_d k_d & 0 & a_d k_d^2 - 1 \end{bmatrix}. \quad (16)$$

The parameters  $a_d$ ,  $b_d$ , and  $k_d$  can be estimated much like in a classical conic fitting algorithm. We perform all subsequent computations with those axis-aligned conics. This parameterization guarantees that all viewpoint conics lie in the same plane ( $Y = 0$ ), in which they can be expressed as

$$\Psi_d \propto \begin{bmatrix} a_d b_d & 0 & -a_d b_d k_d \\ 0 & b_d(a_d - b_d) & 0 \\ -a_d b_d k_d & 0 & a_d b_d k_d^2 + a_d - b_d \end{bmatrix}. \quad (17)$$

However, this parameterization does not guarantee other properties given in Section 2.2, especially that these viewpoint conics all touch in a single point. To achieve this, we state a new result. For a central radially symmetric camera, the intersections of all its viewing cones with the calibration plane are calibration conics  $\hat{\varepsilon}_d$  given by

$$\hat{\varepsilon}_d \propto \begin{bmatrix} \gamma^2 \rho_d & 0 & \epsilon \rho_d \gamma + \gamma \\ 0 & \delta^2(2\rho_d + 1) & 0 \\ \epsilon \rho_d \gamma + \gamma & 0 & \rho_d \epsilon^2 + 2\epsilon - 2 \end{bmatrix}, \quad (18)$$

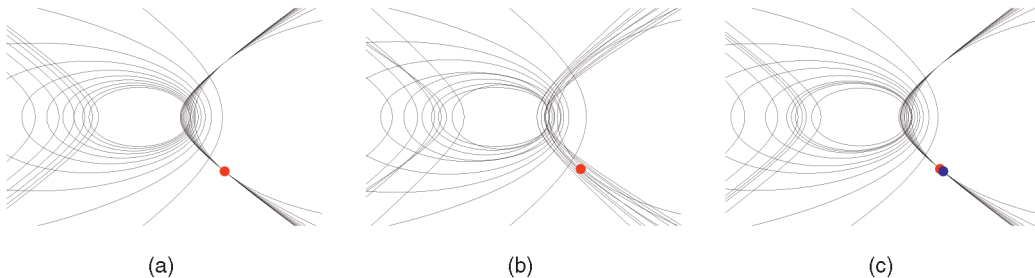


Fig. 5. The effect on viewpoint conics of errors on the calibration conics (shown in the same plane for visualization). (a) Original configuration with the optical center. (b) Added noise to the calibration conics and resulting viewpoint conics. Obviously, finding the closest point to these curves will yield a large error. (c) Recovered optical center in dark/blue and corrected conics given by our approach, cf. Section 4.2.

TABLE 2  
Computing the External Parameters of the Camera  
from the Parameters

Information	Formula
X-coordinate of the optical center	$-\frac{\gamma}{\delta^2} - \frac{\epsilon}{\gamma} + \frac{2}{\gamma}$
Z-coordinate of the optical center	$-\sqrt{-\frac{\gamma^2 - 2\delta^2}{\delta^4}}$
Intersection of optical axis and calibration plane	$-\epsilon/\gamma$
Intersection of calibration plane and principal plane <sup>1</sup> (setting $\phi_d = 0$ )	$y = (2 - \epsilon)/\gamma$

<sup>1</sup> The principal plane refers to the plane passing through the optical center and being parallel to the image plane.

where  $\gamma$ ,  $\epsilon$ , and  $\delta$  encode the external parameters of the camera and  $\rho_d$  is a parameter for the distortion circle of radius  $d$ . For each calibration conic  $\tilde{\epsilon}_d$ , the corresponding viewpoint conic is given by

$$\hat{\Psi}_d \propto \begin{bmatrix} \gamma & 0 & \epsilon + \frac{1}{\rho_d} \\ 0 & \frac{\gamma^2 - 2\delta^2 - \frac{\epsilon^2}{\rho_d}}{\gamma} & 0 \\ \epsilon + \frac{1}{\rho_d} & 0 & \frac{\gamma^2 + 2\delta^2(\epsilon - 1) + \delta^2\epsilon^2\rho_d}{\gamma\delta^2\rho_d} \end{bmatrix}. \quad (19)$$

The derivation and the proof of this formulation are available in [27]. Note that, in practice, we are only interested in the estimation of  $\tilde{\epsilon}_d$ . Much like (7) can be obtained for  $\Psi_d$ , we can compute  $(f_d/d)^2$  for  $\hat{\Psi}_d$ . After some algebraic manipulation using that result (not shown here), we substitute  $\rho_d$  in (18) with

$$\rho_d = -\frac{\phi_d\gamma^2}{2(\gamma^2 - 2\delta^2)} - \frac{1}{2}, \quad (20)$$

where  $\phi_d$  is chosen so to be equal to  $(f_d/d)^2$ .

The formulas for estimating the external parameters of the camera are summarized in Table 2 and briefly derived below. The position of the camera is obtained by solving

$$(X, Z, 1)\Psi_i(X, Z, 1)^\top = (X, Z, 1)\Psi_j(X, Z, 1)^\top$$

for  $(i \neq j)$ . The value  $-\frac{\epsilon}{\gamma}$  gives the X-coordinate of the intersection point of the optical axis with the calibration plane:

$$\forall d : (-\epsilon/\gamma, 0, 1) \Psi_d(X, Z, 1)^\top = 0,$$

where  $\Psi_d(X, Z, 1)^\top$  is the tangent to the viewpoint conic at the optical center, i.e., the optical axis. One may wonder why the extrinsics were not directly “encoded” in the parameterization. Indeed, such parameterizations exist and were investigated. However, they were abandoned because the resulting calibration conics  $\tilde{\epsilon}_d$  were not as “simple” as the one we propose. When a calibration algorithm could be deduced from any of these formulations, it was less stable than the one we show next.

With this global formulation, all calibration conics  $\tilde{\epsilon}_d$  should be fitted at the same time. However, this task is difficult since the function is nonlinear and requires an initial estimate of the parameters.

Perhaps surprisingly, there exists an analytic solution to computing the extrinsic parameters once the values of  $a_d$ ,  $b_d$ , and  $k_d$  for all calibration conics are estimated. The position of the camera can be estimated, while ignoring the focal length at each circle. We exploit the fact that without noise  $\forall d : \epsilon_d = \tilde{\epsilon}_d$ , i.e., the unconstrained and constrained formulations should

give identical results.<sup>2</sup> We solve this equation for each parameter  $\epsilon$ ,  $\gamma$ , and  $\phi_d$  and obtain

$$\phi_d = 2\delta^{-1/2}a_d^{-1}b_d(b_d - a_d) - 1, \quad (21)$$

$$\tilde{\epsilon}_d = -a_d^{-1/2}\sqrt{\gamma^2 + a_d} - \gamma k_d + 1, \quad (22)$$

$$\tilde{\gamma}_d = \pm\delta b_d^{-1}\sqrt{a_d}\sqrt{\delta^2 + 2b_d}, \quad (23)$$

for each calibration conic. Note that the two last equations involve only the extrinsic parameters. We rename them with a  $\tilde{\cdot}$  and a  $d$  subscript because, in the presence of noise, they are different for each calibration conic. This is because each calibration conic is fitted individually. Therefore, an estimate for  $\delta$  can be obtained by minimizing the variance of the  $\tilde{\gamma}_d$ . In our implementation, a minimization over  $\tilde{\gamma}_d^2$  is preferred, since it eliminates the square root in the expression. This results in solving

$$\delta^2 = \arg \min_{\delta} \sigma^2, \text{ and } \sigma^2 = \sum_d (\mu - \tilde{\gamma}_d^2)^2, \quad (24)$$

where  $\mu$  is the mean. The expression  $\sigma^2$  is a fourth degree polynomial in  $\delta^2$ , i.e., it admits only three extrema in  $\delta^2$ . One of them corresponds to  $\delta = 0$  and is a maximum and the other two are minima with identical absolute value but opposite sign. The one larger than zero is our solution for  $\delta^2$  and  $\delta$  can always be taken positive. Given this value, we can estimate the other parameters with  $\epsilon = \text{Mean}(\tilde{\epsilon}_d)$  and  $\gamma = \pm \text{Mean}(\tilde{\gamma}_d)$  using (22) and (23). This sign ambiguity will be resolved later. In the presence of noise, we perform a final optimization:

$$\arg \min_{\delta, \epsilon, \gamma} \sum_d (\epsilon - \tilde{\epsilon}_d)^2 + (\gamma - \tilde{\gamma}_d)^2 \quad (25)$$

using a standard Gauss-Newton algorithm.

Once the external parameters of the camera are known, that is, the value of  $\delta$ ,  $\epsilon$ , and  $\gamma$ , we are seeking the intrinsics  $\phi_d$ . Using (21) did not give satisfying results in practice. Instead, we find  $\tilde{\epsilon}_d$  that best fits the original data points. Given the external parameters, each calibration conic can be fitted using a least square algebraic error function resulting in a second degree polynomial involving the variable  $\phi_d$ . We try both signs for  $\gamma$  and keep the one that best fits the calibration conics.

**Many calibration planes.** If many calibration planes are available, the focal length of each circle must be consistent over all the views. This is accomplished by simultaneously fitting the calibration conics  $\tilde{\epsilon}_d$  for different planes. This global Linear Least Square problem is a second degree polynomial, where  $\phi_d$  is the only variable.

The steps of this algorithm are summarized in Fig. 6. In the following, it is referred to as the Right Cone Constraint (RCC) method.

### 4.3 NSVP Extension

Our previous SVP model can be generalized allowing an NSVP. Due to lack of space, it is not described here. Our tests demonstrated that such a parameterization is not that useful in practice. Indeed, the parameters can only be recovered by

2. We scale  $\epsilon_d$  and  $\tilde{\epsilon}_d$  such that they have unity at the upper left coordinate.



**INPUT:** Dense plane–image correspondences

**ALGORITHM:**

For each calibration plane  $v$ :

- Estimate individually every calibration conic  $\varepsilon_d$ .
- Compute the average major axis  $\mathbf{m}$  and select a point, denoted  $\mathbf{g}$ , on the axis.
- Change the coordinate system of the original data to set  $\mathbf{m}$  on the  $X$ -axis and  $\mathbf{g}$  at the origin.
- Estimate the new  $\varepsilon_d$  of the form:  $\propto \begin{bmatrix} a & 0 & c \\ 0 & b & 0 \\ c & 0 & d \end{bmatrix}$ .
- Solve (24) to obtain  $\delta_v$ .
- Minimize (25) and obtain  $\epsilon_v, \gamma_v$ .

Compute, for all distortion circles  $d$ ,  $\phi_d$  using (18).

Perform iterative refinement of the reprojection error.

Fig. 6. Complete algorithm for the RCC calibration approach.

means of optimization that would require initial estimates of the parameters, which are difficult to obtain.

## 5 COMPUTING THE DISTORTION CENTER

Until now, we have assumed, for both algorithms, that the distortion center was known; this information was used to select the distortion circles. Recall that the distortion center is also the principal point of the camera in our model. Tests with noiseless simulated data showed that the calibration may be quite sensitive to a bad choice of distortion center; indeed, like for real cameras, using the image center as an approximation was not satisfying in general. Hence, the distortion center must be estimated as part of the calibration process. Note that using HK algorithm is not satisfactory in general since the criteria for choosing this point is in terms of image rectification. Besides not being applicable to noncentral cameras, these criteria do not correspond to our model. Below, we propose one where the recovered distortion center is identical to the principal point. The sensitivity of calibration that we observed in simulations suggests that it should be possible to estimate the distortion center rather reliably, which was confirmed in practice.

**Algorithm.** We used the following heuristics to define an optimization criterion for the distortion center. Let us apply the IAC approach of Section 3.1 with several images as input. The plane-based calibration for each distortion circle is then capable of estimating a principal point, besides the focal length  $f_d$ . It seems plausible that the better the assumed distortion center was, the closer the estimated principal points will be to it. Since plane-based calibration is applied on images centered on the assumed distortion center, we can consider the average distance of the estimated principal points (one per distortion circle) as a measure for the goodness of the center.

Fig. 7 shows the values of this measure, computed for distortion center positions on a  $60 \times 60$  grid around the image center, for real cameras. The shape of the cost surface indicates that we can find the optimum distortion center using a simple steepest descent type method. We implemented such an approach that accurately finds the distortion center within a couple of minutes of computation. Note that the second column of Fig. 7 shows that, although the principal points used to plot it were computed individually

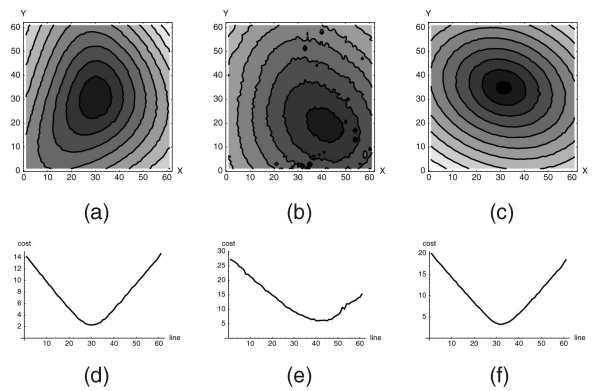


Fig. 7. Plots of the goodness measure for the distortion center, obtained for three tested lenses (cf. Section 7). (a), (c), (e)  $60 \times 60$  grid around the image center (yellow/dark meaning smaller). (b), (d), (f) One slice per plot, through the respective minimum. (a) 3.5 mm. (b) Cata-12 mm. (c) 8 mm. (d) 3.5 mm. (e) Cata-12 mm. (f) 8 mm.

per distortion circle, they are very densely clustered (average distance to assumed distortion center of less than 3 pixels). This suggests a high stability of the calibration.

**Discussion.** Compared to other approaches, our optimization criterion is chosen to find the best optical axis. Thus, the distortion center is identical to a principal point in a pinhole camera. This formulation is also used in [9], [28], [29]. Our criterion is different from the one in image-based distortion functions, where the distortion center (together with the distortion function) is chosen to maximize the linearity of rectified line images [8], [12], [25], [31]. However, the latter can result in instability for the estimation under very low distortion and, more importantly, is not optimal when the camera is NSVP since image rectification is not possible. In theory, ours is not subject to these problems even when no distortion is present. This is confirmed by the nice shape of the error function for the 8.0 mm camera, as shown in Fig. 7. For these reasons, the distortion center computation will not be included in the comparison with HK approach.

## 6 PRACTICAL ISSUES

### 6.1 Dense Plane-Image Correspondences

The easiest approach we found to get dense correspondences between the calibration plane and the camera is to use a flat screen. We used a simple coded structured light algorithm [16], which consists of displaying a sequence of patterns of horizontal and vertical black and white stripes of varying thickness on the screen to encode the position of each screen pixel (cf. Fig. 8). Then, for each camera pixel, we identify the corresponding position on the calibration plane by decoding the observed intensities in each pattern. We found that, when performed in a controlled environment (low-constant ambient lighting, screen of high contrast and resolution), the accuracy of such a method is good enough for calibration. Indeed, we only used correspondences of pixel precision (see [23] for details). Since the points located on the distortion circles are given in floating-point coordinates, we compute their correspondences by a weighted sum of the correspondences recovered for the four closest image pixels. Besides allowing dense correspondences with the calibration plane, these approaches

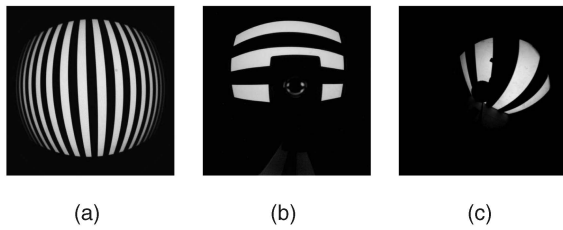


Fig. 8. Projected patterns for correspondences are horizontal and vertical black and white stripes. Images taken with (a) the Goyo 3.5 mm, (b) catadioptric, and (c) paracatadioptric camera (cata-12 mm).

render trivial the problem of recovering the structure of the calibration plane. This is as opposed to using calibration grid images where grid points must be automatically extracted and identified. This is especially difficult when the distortion is very large.

## 6.2 Omnidirectional Cameras

There are several issues worth mentioning for omnidirectional cameras. If the field of view is larger than  $180^\circ$ , then some distortion circles will have viewing cones that actually approach planes. For the RCC approach, this means that fitting the calibration conic may become unstable. These cases can be detected as the ones whose correspondences on the calibration plane are close to collinear. In practice, they are discarded from the actual calibration procedure. In the case of the homography-based algorithm that uses all matches simultaneously, no special attention is needed.

## 6.3 Nonlinear Optimization

Because  $f_d$  is a function of the radius in the distorted image, it is not straightforward to perform the projection of a 3D point into the image (as opposed to the back-projection of image points to 3D). This means that (8) cannot be used directly to perform a nonlinear optimization of the calibration unknowns. Since the focal length function cannot be inverted when it crosses zero, it is preferable to define the distortion in terms of view angle  $\theta_d$ . As seen in Fig. 14b, this function is generally simple, so easily invertible. Given this function, the radius of the image of a 3D point is computed from the angle between the optical axis and the line spanned by the optical center and the 3D point. The nonlinear optimization is then performed with  $\theta_d$  instead of  $f_d$ . The NSVP case cannot be handled as simply since there is no single optical center relative to which to compute the angle  $\theta$ . In this case, we use (8) and estimate a radius  $d_i$  associated to each 3D-2D correspondence, along with the other parameters. This yields a sparse nonlinear problem. We also enforce monotonicity on  $f_d$  and  $\theta_d$ . For example, this can be done approximatively by adding terms to the cost function, of the form:  $(|f_d - f_{d+s}| - (f_d - f_{d+s}))^2$ ,  $s > 0$ , which is a quadratic penalty if the constraints are not enforced but gives 0 otherwise.

## 7 EXPERIMENTS

We tested our approaches using different types of cameras with simulated and real data. They were also compared to the HK algorithm [12]. Since the view angle of some of our camera is larger than  $180^\circ$ , we implement the algorithm for a spherical retina.

### 7.1 Simulation

**SVP cameras.** We simulated two types of cameras: wide angle with small radial distortion and fisheyes with large distortion.<sup>3</sup> The focal length (and distortion) function of each type of camera was built randomly via monotonically decreasing polynomials of fifth degree. We used an image size comparable to our own camera: 1 megapixel (see real data). The shapes of focal length functions of the wide-angle cameras were similar to our 15 mm (Fig. 14) and we used  $f_d$  with  $f_0 = 1,000 \pm 200$  pixels. The simulated fisheyes were analogous to our 3.5 mm, so we used  $f_0 = 400 \pm 100$  pixels, and the shape of the focal length function yielded a field of view close to  $180^\circ$ . In these tests, we assumed a known distortion center and also made sure the camera was never placed in a near fronto-parallel position with respect to the calibration plane.

We compared the reprojection error, the error on the pose, and error on the calibration. We define the latter as the average difference between the recovered focal length function  $f_d$  and the ground truth. Our tests showed that all of them are highly related, so we only show results for the reprojection error. The reprojection errors for the three algorithms with respect to noise and the number of used cameras are shown in Fig. 9. We added Gaussian noise of standard deviation up to 4 pixels to the original data, which consisted of 50 points per distortion circle (which is rather small compared to the several hundred usually available from a structured light dense mapping). In all of our tests, data on the distortion circles are not necessarily evenly distributed. Especially in the case of a very large field of view, only a portion of the image effectively sees the calibration plane. We made sure to properly simulate this effect. Finally, our tests were performed using 3, 7, and 10 views.

In general, all three algorithms performed similarly. The RCC obtains results for the wide-angle camera similar to the other two. However, this is not the case for the fisheye camera because the data points were not uniformly distributed around the distortion circles. Hence, the pose estimation was not as stable. This exposes a weakness of this approach: If the pose for one view is badly estimated, it can potentially destroy the estimation of the focal length function even using many views. This effect is not as important for the HB since the focal length function is estimated with the position of the camera on the optical axis using all the views.

**NSVP cameras.** We performed an in-depth analysis of the performance of our algorithms for NSVP cameras. Three aspects were considered. First, how well can the displacement along the optical axis be recovered under noise? Second, is the linear NSVP algorithm useful in practice? Third, does the NSVP model overfit when the camera is actually SVP? We performed our tests using the homography-based approach because it is the only one that naturally enforces both SVP and NSVP constraints. Two approaches were tested:

1. Initialization using linear calibration based on an SVP assumption, followed by nonlinear optimization of the NSVP model.
2. Initialization as well as optimization using the NSVP model.

3. We did not compare catadioptric cameras with the HK approach because the comparison would have been unfair.

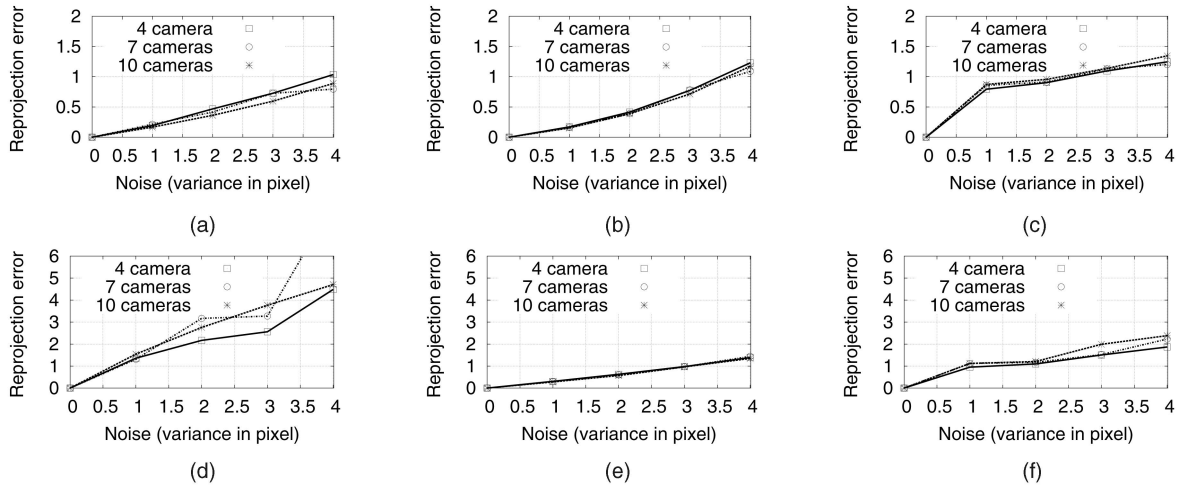


Fig. 9. Reprojection errors (in pixels) for the simulated *wide-angle* and *fish-eye* cameras. (a) RCC wide angle. (b) HB wide angle. (c) HK wide angle. (d) RCC fisheye. (e) HB fisheye. (f) HK fisheye.

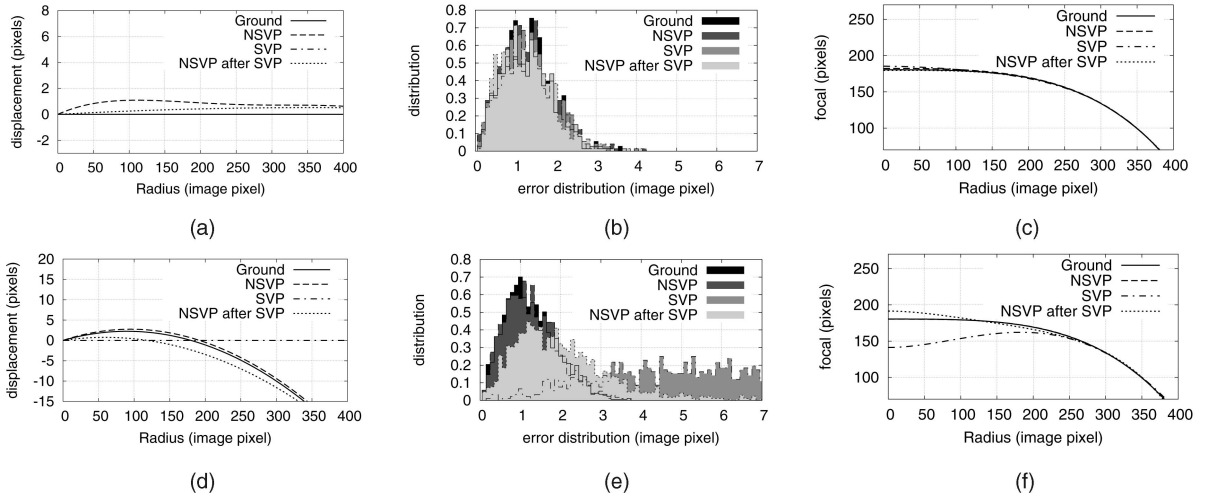


Fig. 10. The two optimization algorithms (see text for details) with simulated data. (a), (d) The recovered displacement  $t_d$ . (b), (e) The error distribution. (c), (f) The focal length functions compared to the ground truth. (a) SVP. (b) SVP. (c) SVP. (d) NSVP. (e) NSVP. (f) NSVP.

The tests were performed on simulated SVP and NSVP catadioptric cameras with viewpoints moving along the optical axis (Figs. 10a and 10d). We used 10 different positions for the plane and added Gaussian noise of standard deviation 1 pixel to the data. Beyond this noise level, we found that our approach could not accurately estimate the viewpoint displacement. Three hundred points per image were used and polynomial models of degree 5 for the calibration were used. A typical behavior of the two approaches is shown in Fig. 10. This leads to the following observations:

- For an SVP camera, both optimizations should lead to similar results (negligible NSVP).
- For an NSVP camera, enforcing an SVP yields a biased focal length function (cf. Fig. 10f). In some cases, this can be satisfying in terms of reprojection error, like for one of our real cameras (cf. Fig. 13).
- When the model is refined to include  $t_d$ , the optimization might not converge to a satisfying minimum.

- If the camera is NSVP, the second approach should perform better, but only if the noise is sufficiently low (cf. Fig. 10d). Otherwise, it can result in worse results because solving with (15) is not as stable as with (9).

## 7.2 Real Data

Several camera configurations were tested. First, a CCTV Basler A201bc was combined with a fisheye Goyo 3.5 mm lens, to an 8 mm Cosmocar lens with small distortion and to a RemoteReality catadioptric lens combined with a 12.5 mm Cosmocar lens (referred to as “cata-12 mm”). Second, a Canon SLR was combined with a fisheye 15 mm lens and to a 0-360 catadioptric lens combined with a 30 mm lens (referred to as “cata-30 mm”). In all cases, the calibration plane of known euclidean structure was a 20-inch LCD screen. The number of calibration views was between 8 and 10 for the different experiments.

Fig. 14a gives the computed focal length of the 15 mm, 3.5 mm, cata-12 mm, and cata-30 mm, with respect to the distance  $d$  to the distortion center, using all methods. These are the functions that were recovered without further optimization based on the reprojection error. Table 3 shows



TABLE 3  
Comparison of the Average Reprojection Errors

Cameras	Algorithms					
	RCC	RCC+NL	HB	HB+NL	HK	HK+NL
cata-12mm	4.60	1.21	1.28	1.14	1.02	0.96
cata-30mm	—	—	1.42	1.33	—	—
3.5mm	6.89	1.15	1.46	1.06	2.53	1.16
8mm	10.74	3.01	6.64	3.08	10.92	3.10
Canon 15mm	0.56	0.48	0.51	0.48	0.51	0.48

NL refers to nonlinear optimization.

TABLE 4  
Comparison of the Average Reprojection Error  
for Different Constraints on the Viewpoint

Cameras	Algorithms-Constraints				
	SVP-L	SVP-NL	NSVP after SVP-NL	NSVP-L	NSVP-NL
cata-12mm	1.65	1.08	1.06	1.58	1.12
cata-30mm	2.22	1.93	1.73	1.42	1.33

"L" refers to the linear algorithms and "NL" to nonlinear optimization of the parameters.

the average reprojection errors of the three algorithms for all cameras. In many cases, the cameras could be calibrated from a single image of the screen (cf. Fig. 12 for the RCC), although, in general, we recommend using at least five images for good stability. Recall that our structured light-based matching provides a large number of correspondences. The cata-12 mm was calibrated with all the approaches (except for HK approach where only the portion of the image corresponding to forward cones was used) with very similar results (cf. Fig. 14). As for the cata-30 mm, since it was found to be noncentral, as described later in this section. The RCC gave very accurate results, however only with a limited number of planes. The difficulties arose when not enough data were available to allow a good fitting of the calibration conics. We handled this by dropping these planes and use only the other ones.

If discrete values for  $f_d$  are computed instead of, e.g., a polynomial function, then only a subset of distortion circles are used for calibration; others can then be extrapolated or interpolated from a polynomial fitting of the data. Let us define this polynomial  $p$ ; from the camera model, it is best to ensure that its derivative at 0 (corresponding to the distortion center) is 0. This constraint is due to the symmetry of the distortion model. Another criterion is that the function should be monotonically decreasing. This last constraint is not directly enforced in our algorithms. However, this did not seem to be an issue in our tests. In

TABLE 5  
Result for Pose Estimation

Algorithms	Position			Angle		
	$p_{01}$	$p_{12}$	$p_{02}$	$a_{01}$	$a_{12}$	$a_{02}$
Ground truth	5	5	10	0°	0°	0°
RCC	4.94	5.00	9.93	0.83°	0.1°	0.92°
HB	4.90	4.94	9.85	0.79°	0.79°	1.6°
HK	4.90	4.96	9.86	0.84°	0.68°	1.51°

The camera was moved to three positions with known relative motion. Coefficients  $p_{ij}$  and  $a_{ij}$  denote the distance (in centimeters) and relative angle (in degrees) between camera positions  $i$  and  $j$ .

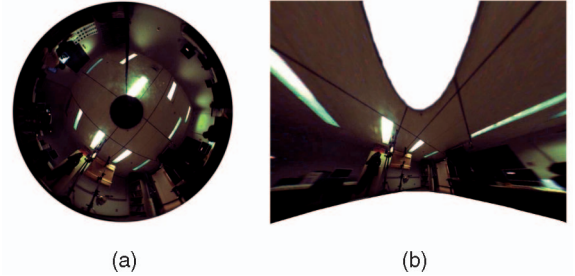


Fig. 11. Image rectification for the Basler camera with the 12.5-mm lens and the RemoteReality lens (cata-12 mm). (a) Original image. (b) Rectified image for a rotated view.

practice, polynomials of degree 5 appeared to be sufficient. To handle the case of omnidirectional cameras more appropriately, the interpolation is carried out with the view angle instead of the focal length. In this case, a monotonically increasing polynomial passing through 0 can also be fitted (see Fig. 14b).

Both catadioptric cameras cata-12 mm and cata-30 mm are typical examples of configurations yielding multiple view-points. Indeed, both mirrors are parabolic and the mounted lenses are perspective [22]. However, only the second one was found to be NSVP (cf. Fig. 13 and Table 4). We conjecture that, although our 12.5 mm camera is not orthographic, it has a field of view sufficiently small to provide a locus of view-points very close to a single effective viewpoint. To verify our hypothesis, it would be useful to perform the test with more specialized equipment like in [22].

Evaluating the results based on the reprojection error can lead to biased conclusions in the case of a generic model. Indeed, the model offers more freedom, which allows to fit the data better. Meaningful quantitative results were obtained for the Goyo 3.5-mm lens, using a pose estimation procedure. Using a translation stage, the camera was moved to three positions with known relative motion (no

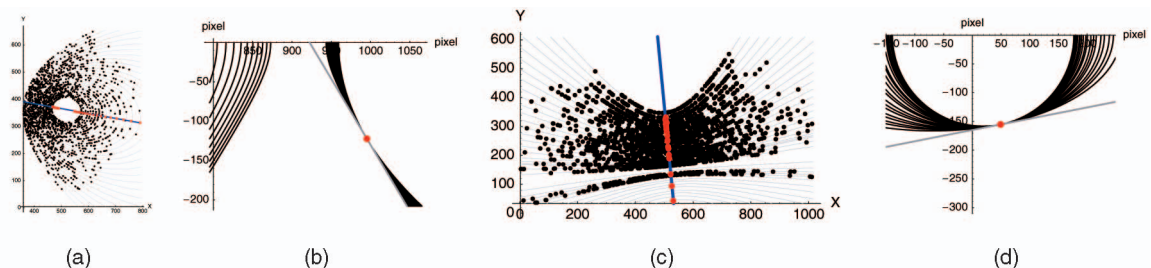


Fig. 12. Calibration with the RCC approach. (a) Fitted ellipses for the Goyo 3.5-mm lens and (b) corresponding hyperbolas, computed intersection, and optical axis (gray line). (c) For the cata-12 mm camera, the intersection between the calibration plane and the cones yielded ellipses and hyperbolas, constraining the viewpoint to lie, respectively, on hyperbolas and ellipses. (d) Intersection of the viewpoint conics.

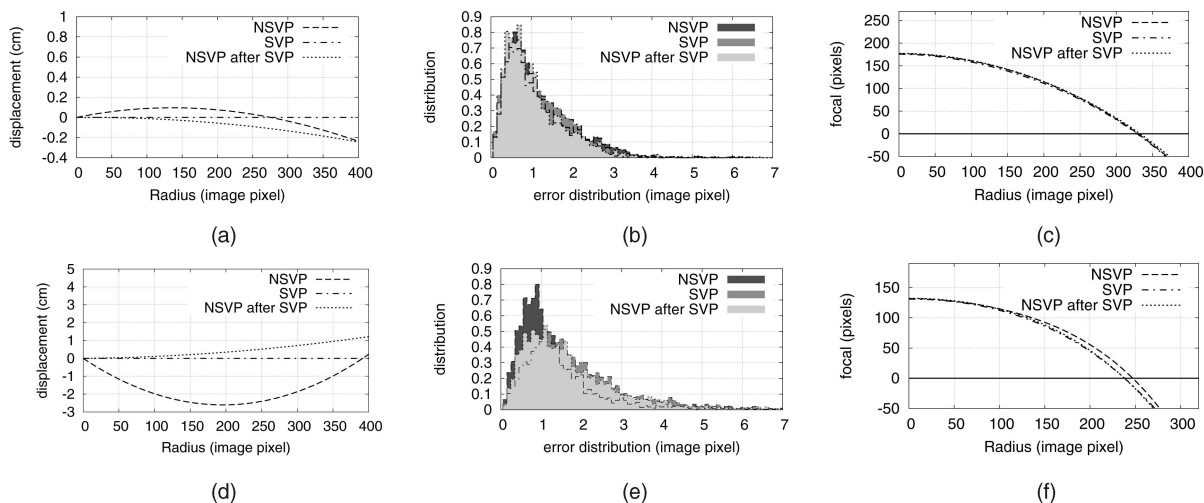


Fig. 13. The two optimization algorithms (see text for details) with our catadioptric cameras. (a), (d) The recovered displacement  $t_d$ . (b), (e) The distribution of reprojection errors. (c), (f) The recovered focal length functions. (a) Cata-12 mm. (b) Cata-12 mm. (c) Cata-12 mm. (d) Cata-30 mm. (e) Cata-30 mm. (f) Cata-30 mm.

rotation, known translation). Using the calibration information (obtained using other images), the pose of the camera relative to the calibration plane was computed for all three positions. From this, the relative motions were computed and compared to the ground truth. The results presented in Table 5 show a good stability for all methods.

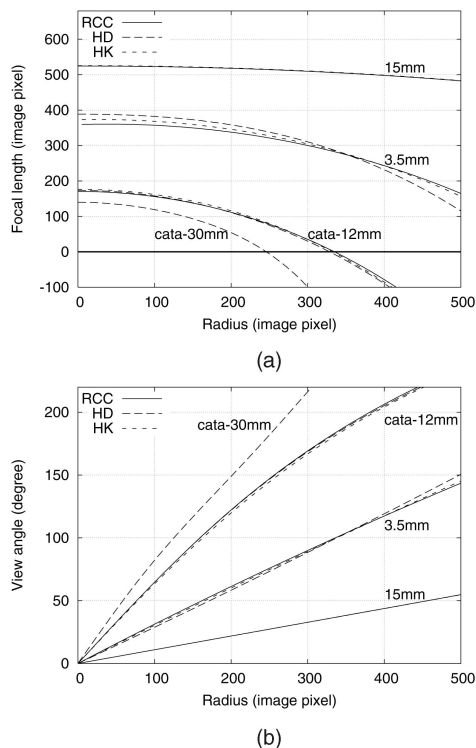


Fig. 14. (a) Recovered focal length (in pixels) for the three algorithms (after polynomial fitting of the data). (b) Recovered view angle (in degrees). These calibration curves were obtained without optimization based on the reprojection error. Performing such an optimization led to very similar results in general. Observe that, for the catadioptric cameras, there are negative focal lengths, meaning that their view angle is larger than  $180^\circ$ .

Images from three panoramic cameras were rectified based on the calibration results (cf. Figs. 16, 11a, and 11b). For the wide-angle Goyo lens and the cata-12 mm, the results seem to be very good, even toward the image borders (cf. Fig. 16b and the inset images in Fig. 11b).

Finally, a homemade catadioptric device built from a Fujinon 12.5 mm lens pointed at a roughly spherical mirror was tested (cf. Fig. 15). Although its radial configuration was not perfect, the distortion center could be found and a satisfying calibration could be obtained with our methods. The “HB” approach gave the best results because it could take advantage of up to eight images, which is more robust to the imperfect configuration of the camera. The rectification is surprisingly good for a large part of the image, especially around the borders (cf. Figs. 16a and 16c). The remaining distortions in the center were found to be caused by a small bump on the “mirror’s” surface.

## 8 SUMMARY AND CONCLUSION

We have proposed new calibration approaches for a camera model that may be a good compromise between flexibility and stability for many camera types, especially wide-angle ones. Previous work showed that the RCC approach might have a limited practical usability because of stability issues [24]. This was because only one calibration plane could be used directly and because camera position was recovered in two steps: conic fitting and finding the closest point to a set of viewpoint conics. Both issues were addressed in this paper and we showed that the use of the RCC approach is



Fig. 15. Homemade catadioptric camera built from a Basler A201bc camera with a Fujinon 12.5 mm lens pointed at a Christmas ornament representing a roughly spherical mirror.

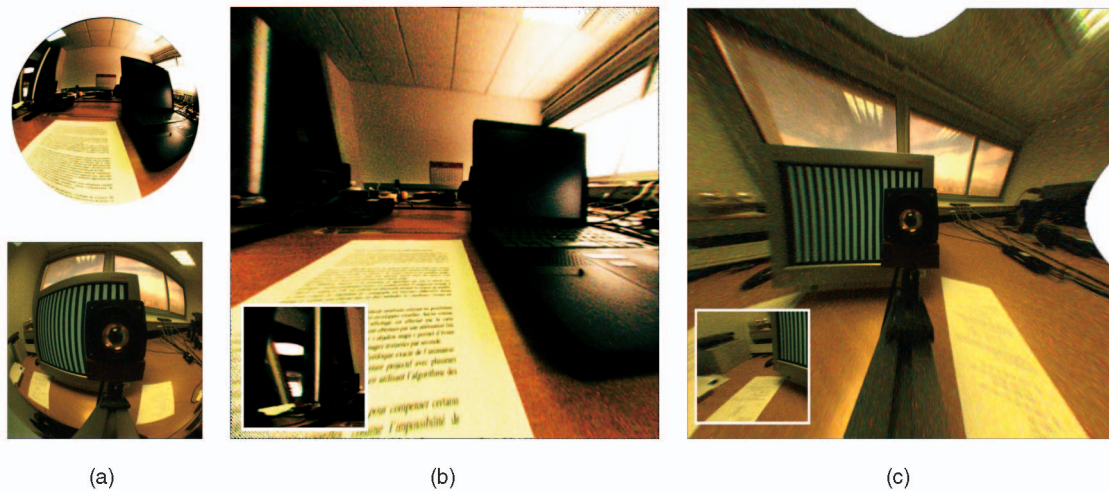


Fig. 16. Image rectification. (a) Original images. (b) Rectified image for the Goyo 3.5 mm. (c) Rectified image for the homemade catadioptric camera. Small inset images show rectification of the border regions.

very well suited when performed with only few camera poses. However, our homography-based approach introduced in this paper is preferable. It can be adapted to using a polynomial distortion model and extended to NSVP configurations. This allows one to perform calibration without a dense plane-to-image matching, unlike the previous approach. Those reasons, and the fact it was the most reliable in our experiments, led us to recommend the HB approach over RCC.

The HK approach gives very good results with the benefit that it has an elegant solution for the distortion center estimation. Our approach gives a similar accuracy but can also deal with NSVP cameras, which is one of the main goals of this work.

## ACKNOWLEDGMENTS

Jean-Philippe Tardif and Peter Sturm gratefully acknowledge the support of the French ANR under Project CAVIAR. Funding was also provided to Jean-Philippe Tardif by the Natural Sciences and Engineering Research Council of Canada. The authors would like to thank the reviewers for useful comments.

## REFERENCES

- [1] Intel Open Source Computer Vision Library, <http://www.intel.com/technology/computing/opencv/>, 2008.
- [2] S. Baker and S.K. Nayar, "A Theory of Single-Viewpoint Catadioptric Image Formation," *Int'l J. Computer Vision*, vol. 35, pp. 175-196, 1999.
- [3] J.P. Barreto and K. Daniilidis, "Unifying Image Plane Liftings for Central Catadioptric and Dioptric Cameras," *Proc. IEEE Int'l Workshop Omnidirectional Vision and Camera Networks*, pp. 151-162, 2004.
- [4] W. Boehm and H. Prautzsch, *Geometric Concepts for Geometric Design*. AK Peters, 1994.
- [5] D.C. Brown, "Close-Range Camera Calibration," *Photogrammetric Eng.*, vol. 37, pp. 855-866, 1971.
- [6] G. Champelboux, S. Lavallee, P. Sautot, and P. Cinquin, "Accurate Calibration of Cameras and Range Imaging Sensor: The Npbs Method," *Proc. IEEE Int'l Conf. Robotics and Automation*, pp. 1552-1557, 1992.
- [7] D. Claus and A.W. Fitzgibbon, "A Rational Function Lens Distortion Model for General Cameras," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 213-219, 2005.
- [8] F. Devernay and O. Faugeras, "Straight Lines Have to Be Straight," *Machine Vision and Applications*, vol. 13, pp. 14-24, 2001.
- [9] C. Geyer and K. Daniilidis, "A Unifying Theory for Central Panoramic Systems and Practical Implications," *Proc. Sixth European Conf. Computer Vision*, vol. 29, pp. 159-179, 2000.
- [10] K.D. Gremban, C.E. Thorpe, and T. Kanade, "Geometric Camera Calibration Using Systems of Linear Equations," *Proc. IEEE Int'l Conf. Robotics and Automation*, pp. 562-567, 1988.
- [11] M.D. Grossberg and S.K. Nayar, "The Raxel Imaging Model and Ray-Based Calibration," *Int'l J. Computer Vision*, vol. 61, pp. 119-137, 2005.
- [12] R. Hartley and S.B. Kang, "Parameter-Free Radial Distortion Correction with Center of Distortion Estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 8, pp. 1309-1321, Aug. 2007.
- [13] 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*, vol. 28, no. 8, pp. 1335-1340, Aug. 2006.
- [14] S.S. Lin and R. Bajcsy, "True Single View Point Cone Mirror Omni-Directional Catadioptric System," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 102-107, 2001.
- [15] B. Micusik and T. Pajdla, "Structure from Motion with Wide Circular Field of View Cameras," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1135-1149, July 2006.
- [16] J. Salvi, J. Pags, and J. Battle, "Pattern Codification Strategies in Structured Light Systems," *Pattern Recognition*, vol. 37, pp. 827-849, 2004.
- [17] D.E. Stevenson and M.M. Fleck, "Nonparametric Correction of Distortion," TR 95-07, Univ. of Iowa, 1995.
- [18] A. Strzebonski, *Cylindrical Algebraic Decomposition*, <http://mathworld.wolfram.com/CylindricalAlgebraicDecomposition.html>, 2008.
- [19] P. Sturm, "Algorithms for Plane-Based Pose Estimation," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 1, 2000.
- [20] P. Sturm and S. Ramalingam, "A Generic concept for Camera Calibration," *Proc. Eighth European Conf. Computer Vision*, vol. 2, pp. 1-13, 2004.
- [21] P.F. Sturm and S.J. Maybank, "On Plane-Based Camera Calibration: A General Algorithm, Singularities, Applications," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 432-437, 1999.
- [22] R. Swaminathan, M.D. Grossberg, and S.K. Nayar, "Caustics of Catadioptric Cameras," *Proc. Eighth IEEE Int'l Conf. Computer Vision*, pp. 2-9, 2001.
- [23] J.P. Tardif and S. Roy, "A MRF Formulation for Coded Structured Light," *Proc. Fifth Int'l Conf. 3-D Digital Imaging and Modeling*, pp. 22-29, 2005.
- [24] J.P. Tardif and P. Sturm, "Calibration of Cameras with Radially Symmetric Distortion," *Proc. IEEE Int'l Workshop Omnidirectional Vision and Camera Networks*, pp. 44-51, 2005.



- [25] J.P. Tardif, P. Sturm, and S. Roy, "Self-Calibration of a General Radially Symmetric Distortion Model," *Proc. Ninth European Conf. Computer Vision*, pp. 186-199, 2006.
- [26] J.P. Tardif, P. Sturm, and S. Roy, "Plane-Based Self-Calibration of Radial Distortion," *Proc. 11th IEEE Int'l Conf. Computer Vision*, pp. 1-8, 2007.
- [27] J.-P. Tardif, P. Sturm, M. Trudeau, and S. Roy, "Calibration of Cameras with Radially Symmetric Distortion," technical report, INRIA, 2008.
- [28] S. Thirithala and M. Pollefeys, "Multi-View Geometry of 1D Radial Cameras and Its Application to Omnidirectional Camera Calibration," *Proc. 10th IEEE Int'l Conf. Computer Vision*, pp. 1539-1546, 2005.
- [29] S. Thirithala and M. Pollefeys, "The Radial Trifocal Tensor: A Tool for Calibrating the Radial Distortion of Wide-Angle Cameras," *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, pp. 321-328, 2005.
- [30] Z. Zhang, "Parameter Estimation Techniques: A Tutorial with Application to Conic Fitting," *Image and Vision Computing*, vol. 15, pp. 59-76, 1997.
- [31] Z. Zhang, "A Flexible New Technique for Camera Calibration," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330-1334, Nov. 2000.



**Jean-Philippe Tardif** received the MSc and PhD degrees in computer science from the Université de Montréal in 2003 and 2007, respectively. In 2007, he was a postdoctoral researcher in the GRASP Laboratory at the University of Pennsylvania. He is currently a postdoctoral fellow in the Center for Intelligent Machines at McGill University. His research interests include structure from motion, calibration and self-calibration of omnidirectional cameras, passive 3D reconstruction, and place recognition.



**Peter Sturm** received the MSc degree from the National Polytechnic Institute of Grenoble (INPG), Grenoble, France, in 1994, the MSc degree from the University of Karlsruhe in 1994, and the PhD degree from INPG in 1997. His PhD thesis was awarded the SPECIF Award (given to one French PhD thesis in computer science per year). After a two-year postdoctorate at Reading University, working with S. Maybank, he joined INRIA in a permanent research position as chargé de recherche in 1999. Since 2006, he has been a directeur de recherche (professor). He has been a member of program committees of the major conferences in computer vision, image processing, and pattern recognition. He will be a program chair of the International Conference on Computer Vision (ICCV '11) and area chair of ICCV '09 and IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '09) and was an area chair for the European Conference on Computer Vision (ECCV '06). He is on the editorial board of *Image and Vision Computing*, the *Journal of Computer Science and Technology*, and the *Information Processing Society of Japan (IPSI) Transactions on Computer Vision and Applications*. He is an organization cochair of ECCV '08 and has organized workshops and given tutorials and invited lectures at several conferences. His main research topics are in computer vision, specifically related to camera (self-) calibration, 3D reconstruction, and motion estimation, both for traditional perspective cameras and omnidirectional sensors. During his undergraduate studies, he had his own one-person software company, within which he was mainly writing and selling software for the organization of sports events. He was involved in the organization of the 2001 Judo World Championships, the 1999 Sumo Amateur World Championships (the first ever to be held outside Japan), the 1994 Judo University World Championships, two European Championships, and numerous other international and national events. He is a member of the IEEE Computer Society.



love was number theory.

**Martin Trudeau** received the BSc degree in mathematics from the Université de Montréal in 1987 and the MSc degree in mathematics from Cornell University in 1993. He currently works on financial models for Mercer. He was previously a researcher in the Vision3D Laboratory, Université de Montréal. Prior to this, he worked on signal processing algorithms for NMS Communications. His interests include almost anything involving mathematics, pure or applied. His first



Being a founding member of the Institut Arts Cultures et Technologies, Université de Montréal, he dedicates considerable attention to the artistic and cultural applications of his research and collaborates actively in the fields of music, theater, architecture, and contemporary arts.

**Sébastien Roy** is an associate professor in the Computer Science Department (Département d'Informatique et de Recherche Opérationnelle) and is the head of the Vision3D Laboratory, Université de Montréal. His research is in the field of 3D computer vision. His interests focus on 3D reconstruction from images, motion, and trajectory analysis from video sequences, as well as using multiple projectors to create a large seamless image over arbitrary-shaped surfaces.

► For more information on this or any other computing topic, please visit our Digital Library at [www.computer.org/publications/dlib](http://www.computer.org/publications/dlib).