HAL
open science

Plane-based self-calibration of radial distortion

Jean-Philippe Tardif, Peter Sturm, Sébastien Roy

To cite this version:

Jean-Philippe Tardif, Peter Sturm, Sébastien Roy. Plane-based self-calibration of radial distortion. ICCV 2007 - 11th IEEE International Conference on Computer Vision, Oct 2007, Rio de Janeiro, Brazil. pp.1-8, 10.1109/ICCV.2007.4409113 . inria-00384287

HAL Id: inria-00384287

https://inria.hal.science/inria-00384287v1

Submitted on 14 May 2009

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.

In proceedings of the Eleventh IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil, October 14-20, 2007.

Plane-based self-calibration of radial distortion

Jean-Philippe Tardif[†] Peter Sturm[‡] Sébastien Roy[†]
[†]Université de Montréal, Canada [‡]INRIA Rhône-Alpes, France
{tardifj,roys}@iro.umontreal.ca peter.sturm@inrialpes.fr

Abstract

We present an algorithm for plane-based self-calibration of cameras with radially symmetric distortions given a set of sparse feature matches in at least two views. The projection function of such cameras can be seen as a projection with a pinhole camera, followed by a non-parametric displacement of the image points in the direction of the distortion center. The displacement is a function of the points' distance to the center. Thus, the generated distortion is radially symmetric. Regular cameras, fish-eyes as well as the most popular central catadioptric devices can be described by such a model.

Our approach recovers a distortion function consistent with all the views, or estimates one for each view if they are taken by different cameras. We consider a least squares algebraic solution for computing the homography between two views that is valid for rectified (undistorted) point correspondences. We observe that the terms of the function are bilinear in the unknowns of the homography and the distortion coefficient associated to each point. Our contribution is to approximate this non-convex problem by a convex one. To do so, we replace the bilinear terms by a set of new variables and obtain a linear least squares problem. We show that like the distortion coefficients, these variables are subject to monotonicity constraints. Thus, the approximate problem is a convex quadratic program. We show that solving it is sufficient for accurately estimating the distortion parameters. We validate our approach on simulated data as well as on fish-eye and catadioptric cameras. We also compare our solution to three state-of-the-art algorithms and show similar performance.

1. Introduction

Cameras with general distortions include pinhole, fish-eyes and many single viewpoint catadioptric devices. The projection function of such cameras can be seen as the projection from a (central) perspective camera, followed by a non-parametric displacement of the imaged point in the direction of the distortion center. This displacement induces a non-linear distortion of the image, *i.e.* straight lines in space are not imaged as straight lines unless passing through the distortion center. A very common assumption about the distortion is its radial symmetry [1, 4, 6, 11, 12, 15, 17, 24, 25, 27]. It implies that the image displacement of a point is a function of its distance to the distortion center (irrespective of radial orientation). Two other assumptions are the alignment of the distortion center with the principal point and a unit aspect ratio. These allow an interpretation of the distortion function not only in terms of image displacement, but in terms of viewing angle with respect to the optical axis. It is an effective way of including cameras that have a field of view larger than $180°$.

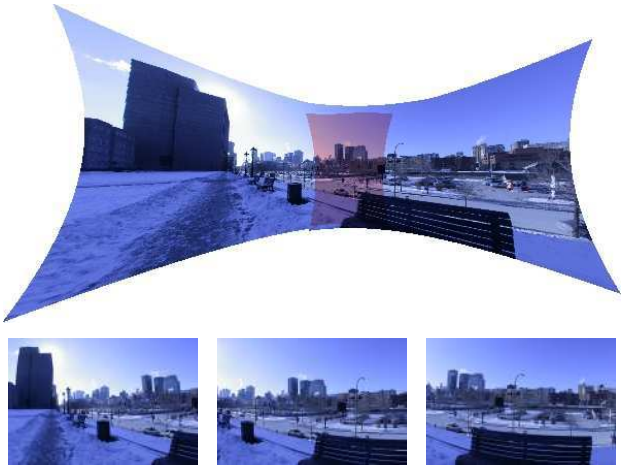


Figure 1. **Top**: Planar stitching with automatic distortion correction. Red/dark overlay shows the mosaic's portion common to all images. **Bottom**: Original images taken with a wide-angle lens.

Problem statement. We present an approach for self-calibrating this general distortion model under the above assumptions. The algorithm uses sparse point matches from at least two views of a plane of unknown geometry, or equivalently, from cameras with coinciding optical centers observing a general scene. We assume *known distortion centers* for the considered cameras.

Organization. In the next section, we present the most im-

portant related work and discuss the main differences with ours. Then, the camera/distortion model we use is reviewed in §3. Our main contribution is given in §4 to §6. Our experiments and results are discussed in §8, followed by the conclusion in §9.

Notation. Matrices are in *sans-serif*, *e.g.* M and vectors in bold, *e.g.* $\mathbf{v}$. We use homogeneous coordinates unless otherwise stated; a bar indicates a vector containing affine coordinates, *e.g.* $\bar{\mathbf{p}}$.

2. Previous work

Traditionally, radial distortions have been treated as a lens aberration to be corrected in the image. Very wide angle lenses had limited applications because of the cameras' small image resolution. In recent years, this limitation has been overcome. It has led to different omnidirectional devices such as fish-eye and catadioptric cameras that can capture large portions of a scene with a high level of detail. In these cameras however, radial distortion is no longer an aberration, but the result of particular designs to increase their field of view.

For these new devices, new camera models are needed. They can be divided into two classes: parametric and non-parametric. Typically, parametric models have been designed for specific acquisition devices. Examples of such models are the classical polynomial model [19], the division model [4], the field of view model [2], the rational model [3], stereographic projection [5] and the unified catadioptric model [6]. A recent tendency has been to apply models originally designed for specific types of cameras to others. It was shown that the unified catadioptric model could also be applied to regular fish-eyes [1, 27] and that the polynomial division model could represent catadioptric cameras [23]. Non-parametric camera models take an opposite point of view. In their most general form, each pixel is associated to one sampling ray [9, 13, 16, 22]. Some researchers have also proposed compromises between parametric and fully general models [11, 12, 23]. The one we use fits into this category. It assumes radial symmetry around a distortion center, but no parametric function is used to describe the distortion.

Self-calibration of cameras is the problem of estimating the cameras' internal and external parameters without using objects of known geometry. One must make assumptions about the camera's internal parameters, *e.g.* constant parameters or unit aspect ratio, or about its external parameters, *e.g.* pure rotation or translation. This paper considers another common assumption, that the observed scene is planar, which, in the context of our approach, is equivalent to seeing a general scene from a purely rotating camera. We show that a general radially symmetric distortion function can be estimated using two or more images. The proposed algorithms can be used in the context of 3D reconstruction and image stitching, *i.e.* mosaic building.

Table 1. Summary of different radial distortion self-calibration approaches. 'Para' refers to parametric distortion model, 'mixed' to algorithms that handle images with different distortions, 'dense' to methods requiring dense matches and $*$ indicates the methods included in our experimental comparison in §8.

References	non-para.	para.	mixed	dense
Barreto *et al.* [1]		×	×	
Claus *et al.* [3]		×		
Fitzgibbon [4]		×		
Geyer *et al.* [7]		×		
Mičušik-Pajdla [15]		×		
Sturm [21]		×	×	
Ramalingam *et al.* [17]	×			×
Tardif *et al.* [23]*	×	×	×	×
Thirthala *et al.* [24]*		×	×	
Thirthala *et al.* [25]*	×		×	
Zhang [28]		×	×	
Ours	×	×	×	

The characteristics of the most closely related self-calibration algorithms are summarized in table 1. We discuss the difference with the works closest to ours, that of Thirthala and Pollefeys [24] and that of Tardif *et al.* [23]. An empirical comparison is also given in §8. The model proposed in [24] will be discussed below. Thirthala and Pollefeys propose a tensor method to perform a projective 3D reconstruction of the scene. Given the reconstruction, the distortion parameters can be recovered. Feature matches in three images of a planar scene are required to compute a trifocal tensor. In [25], this approach is generalized to a general scene and even non-central cameras by using a quadrifocal tensor. In some applications, matching pixels between three views can be unwieldy. A typical situation is in mosaic stitching, as seen in figure 1, where the portion of the region common to all images is very small.

In [23], a plumbline approach is proposed. It can be extended to use dense point features by discovering collinear points. This is the most important weakness of the approach, making it difficult to refine the estimate of the distortion center. Furthermore, it is assumed that many points have (approximately) equal radii, i.e. distances from the distortion center, which simplifies the problem, or a polynomial division model is used directly.

3. Camera model

Our camera model follows the one of Tardif *et al.* [23] and Thirthala and Pollefeys [24]. The discrete representation of the distortion function is also related to the calibration algorithm of Hartley and Kang [11]. We give a brief summary of the geometry of such cameras. There are two complementary ways to describe their sampling function in

the case of radial symmetry and a single effective viewpoint. Keeping in mind both representations gives the intuition behind the algebraic derivation we use.

The description below refers to figure 2. The representation relies on the concept of **radial 1D camera** [24]. We consider one image point to which is associated a **radial line** $\mathbf{l}_i$ joining the point and the distortion center. Let us define the plane Π_i formed by this radial line and the optical axis of the camera. Since the deviation from the pinhole model occurs along the radial line, the sampling ray associated to the point must be located somewhere in this plane. The benefit of modeling the whole camera with a set of such sampling planes is that this circumvents the effect of the distortion of the camera. On the other hand, the distortion parameters (as well as the associated constraints like radial symmetry) cannot be recovered in a single step.

A second, more restrictive, description is possible in terms of **distortion circles** centered in the distortion center [23]. Each circle of radius r_j is associated to a right **viewing cone** C_j in space. These cones are centered in the optical axis and have the camera's optical center as vertex. The **distortion function** relates their opening angle to the radius of the associated distortion circle. An equivalent and more convenient representation however is by using a single pinhole camera for each r_j. For a camera in canonical position, the projection function associated to a point (x, y) is given by:

$$\begin{bmatrix} f_r & 0 & c_x & 0 \\ 0 & f_r & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad r = \|(x - c_x, y - c_y)\|, \qquad (1)$$

where (c_x, c_y) is the position of the distortion center. Thus, the image distortion is described as a dependency of the focal length on the radius, or distance to the distortion center. A zero focal length corresponds to a **viewing plane** Π_p, *i.e.* a cone with opening angle equal to $180°$. A negative focal length models a backward looking cone such as C_3. It comes up when the viewing angle is larger than $180°$ like for some catadioptric or fish-eye cameras. Assuming the distortion center at the origin, the point can be "undistorted" by dividing it by f_r, or equivalently by letting $(x, y, f_r)^\mathsf{T}$ be its rectified homogeneous coordinates.[1] Note that for perspective rectification, it is sufficient to recover the set of f_r up to scale. For the most common cameras, f_r is a smooth monotonically decreasing function. This guarantees that the field of view is strictly monotonically increasing with respect to the radius. In the following, we will call the f_r sometimes focal lengths, sometimes simply distortion coefficients.

[1]However we will refer to f_r as a distortion coefficient.

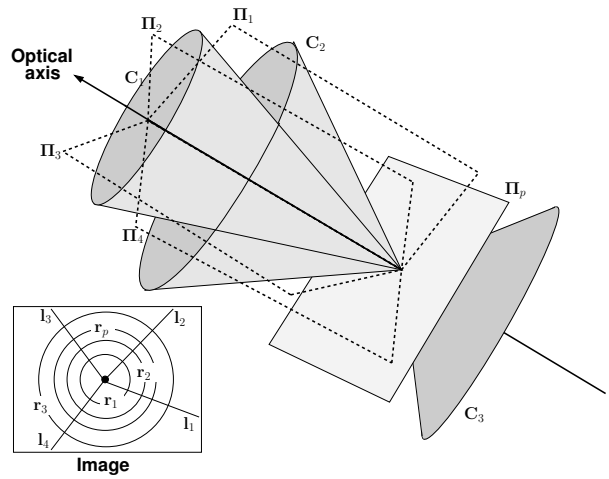


Figure 2. Illustration of our camera model. See text for details.

4. Projective point transfer

We are given a set of correspondences between two images. Our goal is to recover the distortion coefficient associated with the radius of each of the points of the two images. Once these are recovered, it is reasonable to assume that the distortion coefficients for other values of the radius can be computed by interpolation, *e.g.* using a polynomial.

We assume that the distortion center is known. Thus, we can change the coordinate system of our images so the origin coincides with this center. If both cameras see an image of a plane, matching points $\bar{\mathbf{p}}$ and $\bar{\mathbf{q}}$, once rectified, will obey the classical homography image transfer:

$$\begin{pmatrix} \bar{\mathbf{p}} \\ g \end{pmatrix} \propto \mathsf{H} \begin{pmatrix} \bar{\mathbf{q}} \\ f \end{pmatrix}, \text{ with } \bar{\mathbf{p}} = \begin{pmatrix} x \\ y \end{pmatrix} \text{ and } \bar{\mathbf{q}} = \begin{pmatrix} u \\ v \end{pmatrix}, \qquad (2)$$

where g and f are the distortion coefficients associated to $\bar{\mathbf{p}}$ and $\bar{\mathbf{q}}$. Note that this is valid even when f and/or g are zero or negative.

It is well known that two points of $\mathbb{P}^2$ are identical if their cross product vanishes. This is typically used to linearly estimate the homography between two views using point correspondences [10]. However, in our case, this yields two trilinear and one bilinear equations in the elements of H and the distortion coefficients:

$$\begin{pmatrix} x & y & g \end{pmatrix}^\mathsf{T} \times \left(\mathsf{H} \begin{pmatrix} u & v & f \end{pmatrix}^\mathsf{T} \right) = \begin{pmatrix} uyh_{31} + vyh_{32} + fyh_{33} - guh_{21} - gvh_{22} - fgh_{23} \\ guh_{11} + gvh_{12} + fgh_{13} - uxh_{31} - vxh_{32} - fxh_{33} \\ uxh_{21} + vxh_{22} + fxh_{23} - uyh_{11} - vyh_{12} - fyh_{13} \end{pmatrix} = \mathbf{0}. \qquad (3)$$

For reasons that will become clear shortly, we use only the third equation. This is equivalent to considering the first camera as a radial 1D and effectively eliminating the distortion coefficient g. Indeed, we can define the radial line $\mathbf{l} = (0, 0, 1)^\top \times (x, y, 1)^\top = (-y, x, 0)^\top$ and verify that $\left(\bar{\mathbf{p}}^\mathsf{T}, g\right) \mathbf{l} = (x, y, g)(-y, x, 0)^\top = 0$. Hence, we have:

$$\mathbf{l}^\top \begin{pmatrix} \bar{\mathbf{p}} \\ g \end{pmatrix} = \mathbf{l}^\top \mathsf{H} \begin{pmatrix} \bar{\mathbf{q}} \\ f \end{pmatrix} =$$

$$uxh_{21} + vxh_{22} + fxh_{23} - uyh_{11} - vyh_{12} - fyh_{13} \;=\; 0. \quad (4)$$

Note that g does not appear in the equation anymore. Neither do many parameters of H. This is not critical however, since recovering the homography between the two views can be done *a posteriori*. One could formulate self-calibration as a least squares problem, *i.e.* as the minimization of the sum of squares of the term (4) over the available point correspondences. This is a non-convex problem since (4) is bilinear in f, h_{13} and h_{23}. In the next section, we show how this sum of squares can be approximated by a convex quadratic program with inequality constraints.

5. A convex approximation

In this section, we describe our approximation scheme and provide some theoretical insight to justify our approach in §5.1. We assume that both h_{13} and h_{23} are non-zero and fix the scale of H by setting $h_{13} = 1$. Degenerate cases are discussed in §5.2. Thus, (4) simplifies to:

$$uxh_{21} + vxh_{22} + fxh_{23} - uyh_{11} - vyh_{12} - fy = 0. \quad (5)$$

Let us replace the only remaining bilinear term fh_{23} by a new variable: $fh_{23} \rightarrow \alpha$. This gives:

$$uxh_{21} + vxh_{22} + x\alpha - uyh_{11} - vyh_{12} - fy \;=\; 0, \quad (6)$$

$$\text{subject to } \; \alpha = fh_{23}. \quad (7)$$

With n point correspondences, we get n linear equations and n bilinear constraints involving $h_{11}, h_{12}, h_{21}, h_{22}$ and[2] $f_i, \alpha_i, i...n$. So far, we thus have a linear least squares problem with bilinear constraints, which is still non-convex. Our approximation is to replace the bilinear constraints with monotonicity constraints on the f_i and α_i. Let us reorder our correspondence indices i so the $\bar{\mathbf{q}}_i$ are in ascending order of their distance to the origin. We observe that the monotonicity constraint on the f_i also applies to the α_i since they are equal to the f_i up to a scale h_{23}. Since the sign of h_{23} is unknown, we have either $\alpha_i \geq \alpha_{i+1}$ or $\alpha_i \leq \alpha_{i+1}$ for all i. Combining all equations in matrix form, we get:

$$\underbrace{\begin{bmatrix} y_1\bar{\mathbf{q}}_1^\top & -x_1\bar{\mathbf{q}}_1^\top & \bar{\mathbf{p}}_1^\top\mathsf{D} & & \\ \vdots & \vdots & & \ddots & \\ y_n\bar{\mathbf{q}}_n^\top & -x_n\bar{\mathbf{q}}_n^\top & & & \bar{\mathbf{p}}_n^\top\mathsf{D} \end{bmatrix}}_{\mathsf{A}} \underbrace{\begin{pmatrix} h_{11} \\ h_{12} \\ h_{21} \\ h_{22} \\ \alpha_1 \\ f_1 \\ \vdots \\ \alpha_n \\ f_n \end{pmatrix}}_{\mathbf{x}} = \mathbf{0} \quad (8)$$

$$\begin{aligned} \text{subject to} \quad & f_1 \geq f_2 \geq \ldots \geq f_n \\ \text{and either} \quad & \alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_n \\ \text{or} \quad & \alpha_1 \leq \alpha_2 \leq \ldots \leq \alpha_n \end{aligned} \quad (9)$$

where $\mathsf{D} = \mathrm{diag}(-1, 1)$. We assume the general case of sparse matches, where none of the points $\bar{\mathbf{q}}_i$ share the same radius. Consequently, A has more columns than rows and (8) is underconstrained. However, enforcing the monotonicity constraints provides sufficient constraints as explained in §5.1. We compute:

$$\min_{\mathbf{x}} \|\mathsf{A}\mathbf{x}\|^2 \quad (10)$$

$$\text{subject to (9) and } f_j = 1.$$

The index j is a choice from any of the n points. The constraint $f_j = 1$ avoids the trivial solution $\mathbf{x} = \mathbf{0}$ and also fixes the overall scale of the distortion coefficients.[3] The system (10) represents a sparse convex quadratic program since $\mathsf{A}^\top\mathsf{A}$ is positive semi-definite. The optimization of this problem is relatively easy using a modern numerical package. We used the Matlab CVX interface [8] to SeDuMi [20] which implements a sparse interior point method. The choice between $\alpha_i \geq \alpha_{i+1}$ and $\alpha_i \leq \alpha_{i+1}$ depends on the sign of h_{23}. Naturally, it is not known *a priori*. We thus minimize both systems and keep the solution giving the smallest residual error.

In general, solving this problem yields a satisfying distortion function. However, the constraints in (7) are not enforced and, under noise, each α_i/f_i will give a slightly different value. One can estimate h_{23} as the average of these ratios. However, we prefer to use the median as it is more robust to errors. Once h_{23} is estimated, (5) becomes linear in the f_i and the other entries of H. We can re-estimate them with a system similar to (10) with inequality constraints for the f_i only.

One could perform non-linear optimization using the norm of (3) or another meaningful geometric error such as the reprojection error. Note however that the distortion function is not invertible if the distortion coefficients are (close) to zero. In this case, the reprojection error in the original images cannot be computed and we are stuck with using the rectified pixel coordinates. It is preferable to use the projective angle between a point and its transferred point from the other image to perform the optimization.

Finally, full self-calibration of the camera can be done with already known techniques for plane-based self-calibration [26] or for a purely rotating camera [14, 18]. Note that the recovered principal point of the camera need not be identical to the distortion center. Then, a full 3D reconstruction of the points on the plane (or the plane at infinity) can be performed, followed by Euclidean bundle adjustment.

[2] To simplify notations, i in f_i is a point index and not the radius as in f_r above.

[3] Two views of a plane do not allow self-calibration of the 'absolute' focal length.

5.1. Justification

Let us first consider the case where some of the considered points have the same radius. In this case, the number of variables f_i and α_i goes down. With enough points with equal radius, the system becomes overconstrained. In the absence of noise, its solution is clearly the one we are seeking, and as such, it satisfies the constraints (7). With noise, points with equal radius trivially have the same ratio α/f.

In practice, interest points may have similar radius, but usually never exactly identical ones. With the knowledge that the f is smooth and monotonically decreasing, adding constraints (9) provides a reasonable approximation to the above overconstrained situation.

5.2. Degenerate cases

A first degenerate case occurs when either h_{13} or h_{23} are (very close to) zero. In this case (5) simplifies to an equation linear in f and the parameters of H. We now discuss the case where both h_{13} and h_{23} are zero. A first occurrence of this degeneracy is when the camera performs a *pure rotation* around its optical axis. Note, although the distortion is observable in the image, it is not in terms of point transfer. That is, an homography (precisely an image rotation) may satisfy the point transfer for unrectified images.

A more interesting case occurs in *plane-based* self-calibration. In this case, $h_{13} = h_{23} = 0$ implies that $\begin{pmatrix}0 & 0 & 1\end{pmatrix}^\mathsf{T} \propto \mathsf{H}\begin{pmatrix}0 & 0 & 1\end{pmatrix}^\mathsf{T}$ i.e. that the centers of distortion are matching points with respect to the plane homography. Hence, the optical axes intersect in a point on the scene plane. The converse is also true: if the optical axes intersect in the scene plane then $h_{13} = h_{23} = 0$. One observes that (5) simplifies to a linear relation in the upper-left 2×2 elements of H. This implies that H can be estimated up to 3 degrees of freedom. However, it can be shown that no constraint can be obtained on f and g, thus self-calibration is not possible in this case.

6. Regularization

6.1. Intervals for monotonicity constraints

Under noise, the monotonicity constraints of (9) can result in instability for the optimization problem. Intuitively, this happens when the distance between two points, say $\bar{\mathbf{q}}_i$ and $\bar{\mathbf{q}}_j$, is smaller than the noise in their coordinates. The effect of the monotonicity constraints on the results is that "stairs" can appear for f_i and α_i. We thus examined different regularization schemes. For instance, replacing $f_i \geq f_j$ with $f_i \geq f_j - \beta_i, \beta_i \geq 0$ and minimizing these new variables as part of the problem. But this did not resolve the issue and increased the computational burden.

We propose a simpler solution: the idea is to replace the hard monotonicity constraints with interval constraints. The intervals are defined using points with closest absolute difference of radius above a certain threshold ϵ. Formally, the interval for the coefficient f_i corresponding to point $\bar{\mathbf{q}}_i$ is $f_k \geq f_i \geq f_j$, with k the largest index such that $\|\bar{\mathbf{q}}_i\| - \|\bar{\mathbf{q}}_k\| \geq \epsilon$ and with j the smallest index such that $\|\bar{\mathbf{q}}_j\| - \|\bar{\mathbf{q}}_i\| \geq \epsilon$. The same is applied to the α_i. A rule of thumb for selecting ϵ is to set it larger than the maximal error of the point transfer. In practice, we set it to 10 image pixels in all our tests.

6.2. Polynomials and robust computation

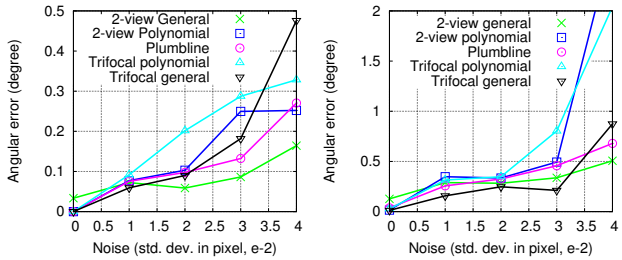
Our approach can be easily modified to directly fit a parametric distortion function instead of recovering a general one. Our tests suggest however that doing so is not as accurate as performing the fitting of the model on the recovered f_i in a second step (see results in §8). Nevertheless, the computation time is significantly reduced, which proves very useful as explained below.

The parametric function can take any form as long as it is linear in its parameters. A typical example is a polynomial. In this case, the relaxation is performed by replacing f_i and α_i by two polynomials $f(r_i) = 1 + \sum_{j=2} \lambda_j r_i^j$ and $\alpha(r_i) = \gamma_0 + \sum_{j=2} \gamma_j r_i^j$ with $r_i = \|\bar{\mathbf{q}}_i\|$ and modify (8) accordingly. Constraint (7) requires that the coefficients of the two polynomials be equal up to a scaling factor h_{23}. Similarly as before, this is replaced by monotonicity constraints on both polynomials with respect to the radii of the considered image points. Thus, using polynomials also implies solving a convex quadratic program. Once again, one can estimate h_{23} as the median of the ratios $\alpha(r_i)/f(r_i)$ at every point and use it to re-estimate $f(r_i)$ and $h_{11}, h_{12}, h_{21}, h_{22}$.

A very important difference with using a discrete function is that $\mathsf{A}^\mathsf{T}\mathsf{A}$ is positive definite. Dropping the constraints and given a sufficient number of points, we obtain an over-constrained linear least square problem. In fact, without noise and given that the model is appropriate, the minimum of this problem will automatically respect the constraints. It is natural to ask whether solving this linear problem would also be sufficient under noise. Our tests suggest that it is indeed a reasonable compromise that reduces significantly the computation time. We could successfully use this fast approximation inside a robust algorithm based on random sampling, *e.g.* RANSAC.

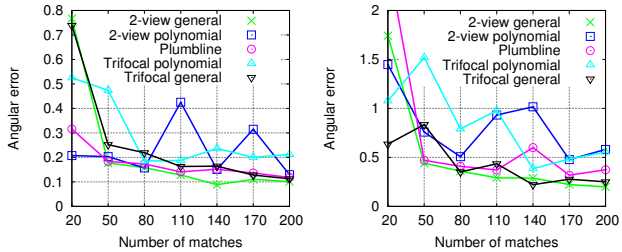
7. The case of multiple views

When many image pairs between two different cameras are available or when both images were taken from the same one, it is useful to estimate all 2-view relations in a single problem. The benefit is that the distortion coefficients of the features from all images can be combined. To do so, one sorts the distortion coefficients of all the views and applies constraints such as in (9) or intervals as explained in



(a) Fish-eye (b) Catadioptric

Figure 3. Comparison of the algorithms with respect to noise.



(a) Fish-eye (b) Catadioptric

Figure 4. Comparison with respect to the number of features.

§6.1. Let $\mathsf{H}^1, ..., \mathsf{H}^V$ be the homographies. The only issue concerns the sign of the h^v_{23} which induces a choice of constraints in (9). Remind that for one image pair, both positive and negative values of h^v_{23} must be tested. With V relations, this yields 2^V minimization to solve. A simpler method is to individually solve each homography in order to get the sign of h^v_{23}. The joint minimization is solved only once with the obtained signs.

8. Experiments

We compared our approach on simulated datasets with both Thirthala-Pollefeys trifocal tensor methods [24, 25] (referred to as 'trifocal polynomial' and 'trifocal general') and the plumbline method of Tardif *et al.* [23]. For our method, we recovered both (discretely sampled) general distortion functions and polynomials of degree four. In the case of the general function, the interpolation was done by fitting such a polynomial to the estimated f_i.

8.1. Simulation

We simulated several fish-eye and catadioptric lenses. Their distortion functions were randomly generated with monotonically decreasing polynomials. Catadioptric cameras had larger distortions than the fish-eyes and we also made sure their viewing angle was larger than $180°$. Note that the tested algorithms take slightly different input. For our methods and the trifocal tensor based ones, we generated respectively 2-view and 3-view correspondences. The data for the plumbline method was generated from the projection of points located on 3D lines. After the distortion function was recovered, we assumed that the other internal parameters could be recovered with a conventional self-calibration algorithm. The error was computed by measuring the angle between the real back-projection rays and the one from the estimated models, for 100 newly generated image points.

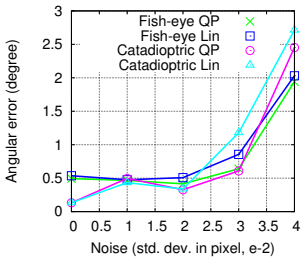


Figure 5. Computation using a polynomial function and the minimum number of required matches. Comparison between incorporating monotonicity constraints (QP) or not (Lin) for a fish-eye lens and a catadioptric device.

Two experiments tested the sensitivity to noise and to the size of the dataset. The first one, illustrated in figure 3, used 100 features per view (2-view or 3-view correspondences, or points from 3D lines), with added Gaussian noise of varying level. The second one, illustrated in figure 4, used a fixed noise level of $\sigma = 3 \times 10^{-2}$ with different number of features. Note that a larger error for catadioptric cameras is expected since their field of view is larger. Our method provides accurate and stable results and even outperforms the others in many tests, although using only two views. A result of our approximation scheme is that the error is not exactly zero in the absence of noise. However, it remains very low even with high noise. Another important conclusion is that solving a general distortion function is better than directly recovering the polynomial, except when only a small number of correspondences are available.

Another experiment refers to §6.2: we compared the linear and constrained quadratic formulation for solving a third degree polynomial distortion function using 9 image correspondences (the minimal requirement being $8\frac{1}{2}$). Our results, illustrated in figure 5, strongly suggest that this approximation does not substantially worsen the results.

8.2. Real images

Several real cameras were tested. In each case, between 400 and 700 correspondences were automatically found (except in one case) between two images and outliers were removed with RANSAC using the linear formulation for estimating a third degree polynomial.

Figure 1 shows a 3-view mosaic built using a 15mm wide angle lens mounted on a Canon SLR. Pairwise correspondences were extracted and a global distortion function

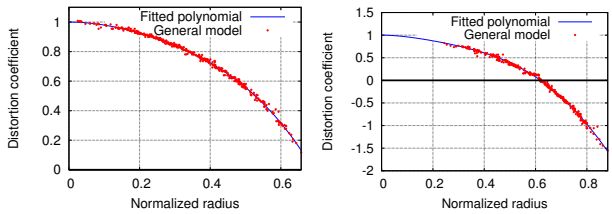


Figure 6. Distortion function for **left:** fish-eye (see also figure 7) **right:** first catadioptric camera (see also figure 8).

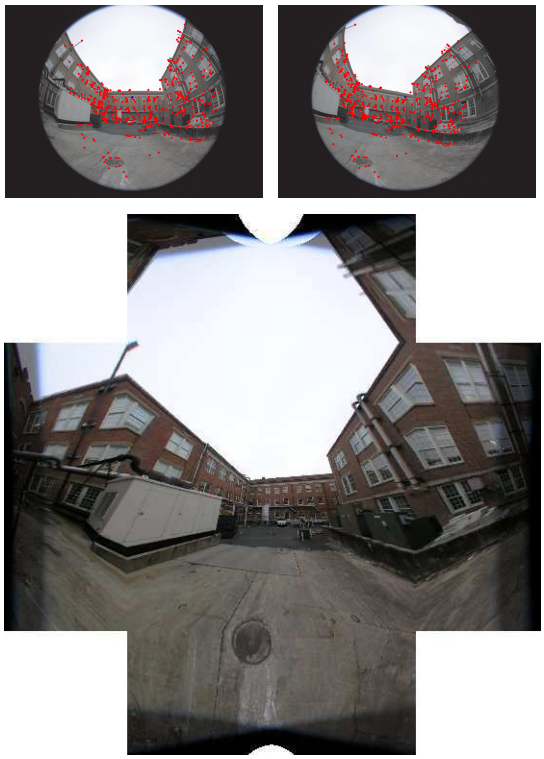


Figure 7. Fish-eye images. **Top:** Two of the images under pure rotation. **Bottom:** The images combined together in a cubemap.

was recovered by estimating the homographies relating the views in both directions.

We also calibrated three omnidirectional cameras: a Nikon 8mm fish-eye lens mounted on Nikon Coolpix camera, and two different catadioptric cameras. In all three cases, two images under pure rotation were used to recover the distortion function, followed by Euclidean upgrade using the method proposed in [18]. The recovered distortion functions of the first two cameras are shown in figure 6. In all cases, the two images could be combined together in undistorted cubemaps (*cf*. figures 7, 8 and 9). Non-linear optimization was not required to obtain very good results on image rectification, but was used for accurate stitching of the images.

Finally, our second catadioptric camera was also fully calibrated from two images of a plane (*cf*. figure 10). In

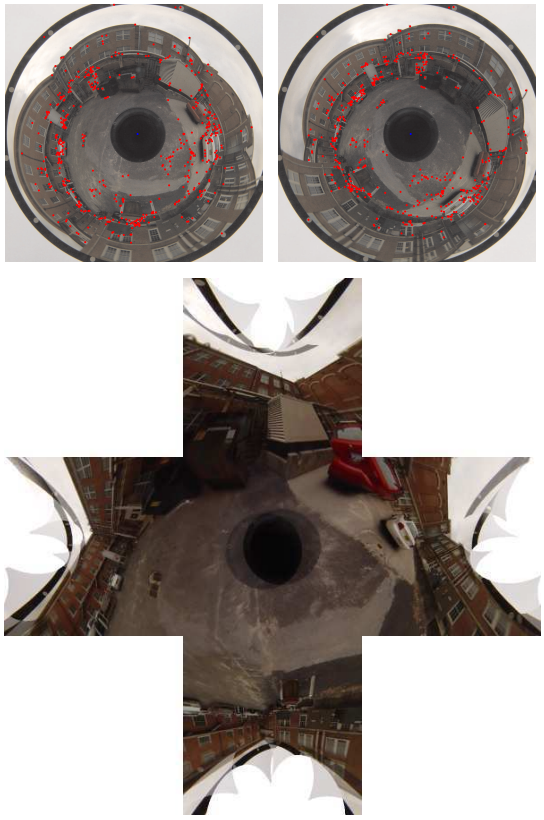


Figure 8. First catadioptric camera. **Top:** Two images under pure rotation. **Bottom:** The images combined together in a cubemap.

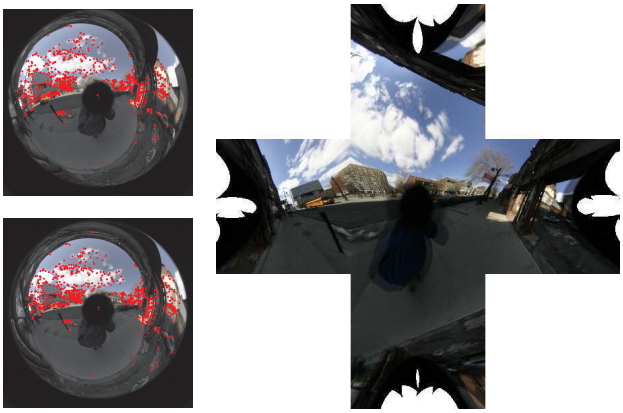


Figure 9. Second catadioptric camera. **Top:** two images under pure rotation. **Bottom:** The images combined together in a cubemap.

this case, we had to manually remove the correspondences outside of the plane since the camera displacement was not large enough to disambiguate perfectly between pure rotation and planar self-calibration. Image rectification is very good, but not as good as in the pure rotation case. Indeed, features from the plane can cover at most half of each image and are more difficult to obtain near borders.

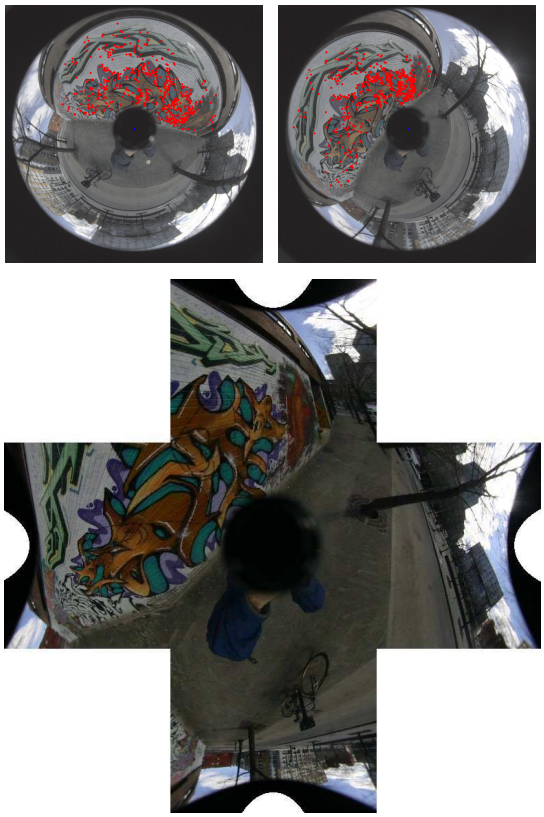


Figure 10. Plane-based self-calibration of the second catadioptric camera. **Top:** The original images. **Bottom:** Right image rectified.

9. Conclusion

We have demonstrated a practical solution to plane-based self-calibration of radially symmetric distortion. Our contribution is that it can use sparse point matches from only two views. Our method provides accurate and stable results based on a convex approximation of an initially non-convex problem.

Our future work will be focused on applying this idea to perform two-view self-calibration under general motion and scene structure.

References

[1] J.P. Barreto, K. Daniilidis. Epipolar geometry of central projection systems using veronese maps. CVPR 2006.

[2] F. Devernay, O. Faugeras. Straight lines have to be straight: Automatic calibration and removal of distortion from scenes of structured environments. MVA 2001.

[3] D. Claus, A.W. Fitzgibbon. Rational Function Model for Fish-eye Lens Distortion. CVPR 2005.

[4] A.W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. CVPR 2001.

[5] M.M. Fleck. Perspective Projection: The Wrong Imaging Model. TR 95–01, University of Iowa, 1995.

[6] C. Geyer, K. Daniilidis, Catadioptric Projective Geometry. IJCV, 45(3), 2001.

[7] C. Geyer, K. Daniilidis. Structure and Motion from Uncalibrated Catadioptric Views. CVPR 2001.

[8] M. Grant, S. Boyd, Y. Ye. Disciplined Convex Programming. Global Optimization: From Theory to Implementation, Kluwer, 2005.

[9] E. Grossmann, E-J Lee, P. Hislop, D. Nistér, H. Stewénius, Are two rotational flows sufficient to calibrate a smooth non-parametric sensor?, CVPR 2006.

[10] R. Hartley, A. Zisserman. Multiple View Geometry in Computer Vision Cambridge University Press 2000.

[11] R.I. Hartley, S.B. Kang. Parameter-free Radial Distortion Correction with Centre of Distortion Estimation. ICCV 2005.

[12] J. Kannala, S.S. Brandt. A Generic Camera Model and Calibration Method for Conventional, Wide-Angle, and Fish-Eye Lenses. PAMI, 28(8):1335-1340, 2006.

[13] D. Nistér, H. Stewénius, E. Grossman. Non-Parametric Self-Calibration. ICCV 2005.

[14] S. Maybank, O. Faugeras. A Theory of Self-Calibration of a Moving Camera. IJCV, 8(2):123-151, 1992.

[15] B. Micusik, T. Pajdla , Structure from Motion with Wide Circular Field of View Cameras, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 28(7), 2006.

[16] S. Ramalingam, P. Sturm, S.K. Lodha. Towards Generic Self-Calibration of Central Cameras. OMNIVIS 2005.

[17] S. Ramalingam, P. Sturm, E. Boyer. A Factorization Based Self-Calibration for Radially Symmetric Cameras, 3DPVT 2006.

[18] H.-Y. Shum, R. Szeliski. Systems and Experiment Paper: Construction of Panoramic Image Mosaics with Global and Local Alignment. IJCV, 36(2):101-130, 2000.

[19] C. Slama, editor. Manual of Photogrammetry. American Society of Photogrammetry, Falls Church, VA, 4th edition, 1980.

[20] J.F. Sturm. Using SeDuMi 1.02, A Matlab toolbox for optimization over symmetric cones. Optimization Methods and Software, 1999.

[21] P. Sturm. Mixing Catadioptric and Perspective Cameras. OMNIVIS 2002.

[22] P. Sturm, S. Ramalingam. A generic concept for camera calibration. ECCV, 1-13, 2004.

[23] J.-P. Tardif, P. Sturm, S. Roy. Self-calibration of a general radially symmetric distortion model ECCV, 186-199, 2006.

[24] S. Thirthala, M. Pollefeys. The Radial Trifocal Tensor. A tool for calibrating the radial distortion of wide-angle cameras. CVPR, 321-328, 2005.

[25] S. Thirthala, M. Pollefeys. Multi-View Geometry of 1D Radial Cameras and its Application to Omnidirectional Camera Calibration. ICCV, 1539-1546, 2005.

[26] B. Triggs. Autocalibration from Planar Scenes. ECCV 1998.

[27] X. Ying, Z. Hu. Can We Consider Central Catadioptric Cameras and Fisheye Cameras within a Unified Imaging Model. ECCV 2004.

[28] Z. Zhang. On the Epipolar Geometry Between Two Images With Lens Distortion. ICPR 1996.