

# A Generic Concept for Camera Calibration

Peter Sturm, Srikumar Ramalingam

► **To cite this version:**

Peter Sturm, Srikumar Ramalingam. A Generic Concept for Camera Calibration. 8th European Conference on Computer Vision (ECCV '04), May 2004, Prague, Czech Republic. pp.1-13, 10.1007/978-3-540-24671-8\_1 . inria-00524411

**HAL Id: inria-00524411**

**<https://hal.inria.fr/inria-00524411>**

Submitted on 25 May 2011

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

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

# A Generic Concept for Camera Calibration

Peter Sturm<sup>1</sup> and Srikumar Ramalingam<sup>2</sup>

<sup>1</sup> INRIA Rhône-Alpes, 38330 Montbonnot, France

Peter.Sturm@inrialpes.fr • <http://www.inrialpes.fr/movi/people/Sturm/>

<sup>2</sup> Dept. of Computer Science, University of California, Santa Cruz, CA 95064, USA

**Abstract.** We present a theory and algorithms for a generic calibration concept that is based on the following recently introduced general imaging model. An image is considered as a collection of pixels, and each pixel measures the light travelling along a (half-) ray in 3-space associated with that pixel. Calibration is the determination, in some common coordinate system, of the coordinates of all pixels' rays. This model encompasses most projection models used in computer vision or photogrammetry, including perspective and affine models, optical distortion models, stereo systems, or catadioptric systems – central (single viewpoint) as well as non-central ones. We propose a concept for calibrating this general imaging model, based on several views of objects with known structure, but which are acquired from unknown viewpoints. It allows in principle to calibrate cameras of any of the types contained in the general imaging model using one and the same algorithm. We first develop the theory and an algorithm for the most general case: a non-central camera that observes 3D calibration objects. This is then specialized to the case of central cameras and to the use of planar calibration objects. The validity of the concept is shown by experiments with synthetic and real data.

## 1 Introduction

We consider the camera calibration problem, i.e. the estimation of a camera's intrinsic parameters. A camera's intrinsic parameters (plus the associated projection model) give usually exactly the following information: for any point in the image, they allow to compute a ray in 3D along which light travels that falls onto that point (here, we neglect point spread).

Most existing camera models are parametric (i.e. defined by a few intrinsic parameters) and address imaging systems with a single effective viewpoint (all rays pass through one point). In addition, existing calibration procedures are tailor-made for specific camera models.

The aim of this work is to relax these constraints: we want to propose and develop a calibration method that should work for any type of camera model, and especially also for cameras without a single effective viewpoint. To do so, we first renounce on parametric models, and adopt the following very general model: a camera acquires images consisting of pixels; each pixel captures light that travels along a ray in 3D. The camera is fully described by:

- the coordinates of these rays (given in some local coordinate frame).
- the mapping between rays and pixels; this is basically a simple indexing.

This general imaging model allows to describe virtually any camera that captures light rays travelling along straight lines<sup>3</sup>. Examples (cf. figure 1):

- a camera with any type of optical distortion, such as radial or tangential.
- a camera looking at a reflective surface, e.g. as often used in surveillance, a camera looking at a spherical or otherwise curved mirror [10]. Such systems, as opposed to central catadioptric systems [3] composed of cameras and parabolic mirrors, do not in general have a single effective viewpoint.
- multi-camera stereo systems: put together the pixels of all image planes; they “catch” light rays that definitely do not travel along lines that all pass through a single point. Nevertheless, in the above general camera model, a stereo system (with rigidly linked cameras) is considered as a **single** camera.
- other acquisition systems, see e.g. [4, 14, 19], insect eyes, etc.

*Relation to previous work.* See [9, 17] for reviews and references on existing calibration methods and e.g. [6] for an example related to *central* catadioptric devices. A calibration method for certain types of *non-central* catadioptric cameras (e.g. due to misalignment of mirror), is given in [2].

The above imaging model has already been used, in more or less explicit form, in various works [8, 12–16, 19, 23–25], and is best described in [8], where also other issues than sensor geometry, e.g. radiometry, are discussed. There are conceptual links to other works: acquiring an image with a camera of our general model may be seen as sampling the plenoptic function [1], and a light field [11] or lumigraph [7] may be interpreted as a single image, acquired by a camera of an appropriate design.

To our knowledge, the only previously proposed calibration approaches for the general imaging model, are due to Swaminathan, Grossberg and Nayar [8, 22]. The approach in [8] requires the acquisition of two or more images of a calibration object with known structure, and knowledge of the camera or object motion between the acquisitions. In this work, we develop a completely general approach, that requires taking three or more images of calibration objects, from arbitrary and **unknown viewing positions**. The approach in [22] does not require calibration objects, but needs to know the camera motion. Calibration is formulated as a non-linear optimization problem. In this work, “closed-form” solutions are proposed (requiring to solve linear equation systems).

Other related works deal mostly with epipolar geometry estimation and modeling [13, 16, 24] and motion estimation for already calibrated cameras [12, 15].

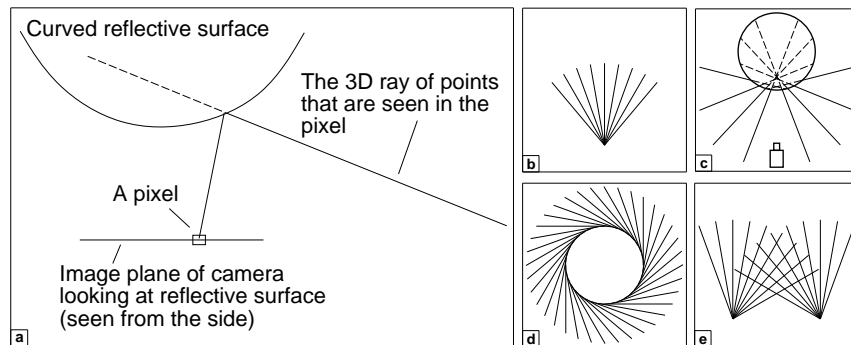
*Organization.* In §2, we explain the camera model used and give some notations. For ease of explanation and understanding, the calibration concept is first introduced for 2D cameras, in §3. The general concept for 3D cameras is described in §4 and variants (central vs. non-central camera and planar vs. 3D calibration objects) are developed in §5. Some experimental results are shown in §6, followed by discussions and conclusions.

<sup>3</sup> However, it would not work for example with a camera looking from the air, into water: still, to each pixel is associated a refracted ray in the water. However, when the camera moves, the refraction effect causes the set of rays to move non-rigidly, hence the calibration would be different for each camera position.

## 2 Camera Model and Notations

We give the definition of the (purely geometrical) camera model used in this work. It is essentially the same as the model of [8] where in addition other issues such as point spread and radiometry are treated. We assume that a camera delivers images that consist of a set of pixels, where each pixel captures/measures the light travelling along some half-ray. In our calibration method, we do not model half-rays explicitly, but rather use their infinite extensions – **camera rays**. Camera rays corresponding to different pixels need not intersect – in this general case, we speak of **non-central cameras**, whereas if all camera rays intersect in a single point, we have a **central camera** with an **optical center**.

Furthermore, the physical location of the actual photosensitive elements that correspond to pixels, does in general not matter at all. On the one hand, this means that the camera ray corresponding to some pixel, needs not pass through that pixel, cf. figure 1. On the other hand, neighborhood relations between pixels are in general not necessary to be taken into account: the set of a camera’s photosensitive elements may lie on a single surface patch (image plane), but may also lie on a 3D curve, on several surface patches or even be placed at completely isolated positions. In practice however, we do use some continuity assumption, useful in the stage of 3D-2D matching, as explained in §6: we suppose that pixels are indexed by two integer coordinates like in traditional cameras and that camera rays of pixels with neighboring coordinates, are “close” to one another.



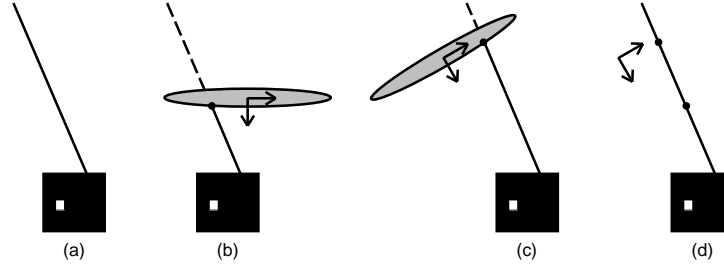
**Fig. 1.** Examples of imaging systems. (a) Catadioptric system. Note that camera rays do not pass through their associated pixels. (b) Central camera (e.g. perspective, with or without radial distortion). (c) Camera looking at reflective sphere. This is a non-central device (camera rays are not intersecting in a single point). (d) Omnivergent imaging system [14, 19]. (e) Stereo system (non-central) consisting of two central cameras.

## 3 The Calibration Concept for 2D Cameras

We consider here a camera and scene living in a 2D plane, i.e. camera rays are lines in that plane. Two images are acquired, while the imaged object undergoes some motion. Consider a single pixel and its camera ray, cf. figure 2. Figures 2 (b) and (c) show the two points on the object that are seen by that pixel in the two images. We suppose to be able to determine the coordinates of these two points, in some local coordinate frame attached to the object (“matching”).

*The case of known motion.* If the object’s motion between image acquisitions is known, then the two object points can be mapped to a single coordinate frame, e.g. the object’s coordinate frame at its second position, as shown in figure 2 (d). Computing our pixel’s camera ray is then simply done by joining the two points. This summarizes the calibration approach proposed by Grossberg and Nayar [8], applied here for the 2D case. Camera rays are thus initially expressed in a coordinate frame attached to the calibration object. This does not matter (all that counts are the relative positions of the rays), but for convenience, one would typically choose a better frame. For a central camera for example, one would choose the optical center as origin or for a non-central camera, the point that minimizes the sum of distances to the set of camera rays (if it exists).

Note that it is not required that the two images be taken of the same object; all that is needed is knowledge of point positions relative to coordinate frames of the objects, and the “motion” between the two coordinate frames.



**Fig. 2.** (a) The camera as black box, with one pixel and the associated camera ray. (b) The pixel sees a point on a calibration object, whose coordinates are identified in a frame associated with the object. (c) Same as (b), for another position of the object. (d) Due to known motion, the two points on the calibration object can be placed in the same coordinate frame. The camera ray is then determined by joining them.

*The case of unknown motion.* This approach is no longer applicable and we need to estimate, implicitly or explicitly, the unknown motion. We show how to do this, given three images. Let  $\mathbf{Q}$ ,  $\mathbf{Q}'$  and  $\mathbf{Q}''$  be the points on the calibration objects, that are seen in the same pixel. These are 3-vectors of homogeneous coordinates, expressed in the respective local coordinate frame. Without loss of generality, we choose the coordinate frame associated with the object’s first position, as common frame. The unknown relative motions between the second and third frames and the first one, are given by  $2 \times 2$  rotation matrices  $\mathbf{R}'$  and  $\mathbf{R}''$  and translation vectors  $\mathbf{t}'$  and  $\mathbf{t}''$ . Note that  $R'_{11} = R'_{22}$  and  $R'_{12} = -R'_{21}$  (same for  $\mathbf{R}''$ ). Mapping the calibration points to the common frame gives points

$$\mathbf{Q} \quad \begin{pmatrix} \mathbf{R}' & \mathbf{t}' \\ \mathbf{0}^T & 1 \end{pmatrix} \mathbf{Q}' \quad \begin{pmatrix} \mathbf{R}'' & \mathbf{t}'' \\ \mathbf{0}^T & 1 \end{pmatrix} \mathbf{Q}'' \quad .$$

They must lie on the pixel’s camera ray, i.e. must be collinear. Hence, the determinant of the matrix composed of their coordinate vectors, must vanish:

$$\begin{vmatrix} Q_1 & R'_{11}Q'_1 + R'_{12}Q'_2 + t'_1Q'_3 & R''_{11}Q''_1 + R''_{12}Q''_2 + t''_1Q''_3 \\ Q_2 & R'_{21}Q'_1 + R'_{22}Q'_2 + t'_2Q'_3 & R''_{21}Q''_1 + R''_{22}Q''_2 + t''_2Q''_3 \\ Q_3 & Q'_3 & Q''_3 \end{vmatrix} = 0 \quad . \quad (1)$$

i	$C_i$	$V_i$
1	$Q_1 Q_1' Q_3'' + Q_2 Q_2' Q_3''$	$R_{21}'$
2	$Q_1 Q_2' Q_3'' - Q_2 Q_1' Q_3''$	$R_{22}'$
3	$Q_1 Q_3 Q_1'' + Q_2 Q_3 Q_2''$	$-R_{21}''$
4	$Q_1 Q_3 Q_2'' - Q_2 Q_3 Q_1''$	$-R_{22}''$
5	$Q_3 Q_1' Q_1'' + Q_3 Q_2' Q_2''$	$R_{11}' R_{21}'' - R_{11}'' R_{21}'$
6	$Q_3 Q_1' Q_2'' - Q_3 Q_2' Q_1''$	$R_{11}' R_{22}'' - R_{12}' R_{21}''$
7	$Q_1 Q_3 Q_3''$	$t_2' - t_2''$
8	$Q_2 Q_3 Q_3''$	$-t_1' + t_1''$
9	$Q_3 Q_1' Q_3''$	$R_{11}' t_2'' - R_{21}' t_1''$
10	$Q_3 Q_2' Q_3''$	$R_{12}' t_2'' - R_{22}' t_1''$
11	$Q_3 Q_3 Q_1''$	$R_{21}'' t_1' - R_{11}'' t_2'$
12	$Q_3 Q_3 Q_2''$	$R_{22}'' t_1' - R_{12}'' t_2'$
13	$Q_3 Q_3 Q_3''$	$t_1' t_2'' - t_1'' t_2'$

**Table 1.** Non-zero coefficients of the trifocal calibration tensor for a general 2D camera.

This equation is trilinear in the calibration point coordinates. The equation's coefficients may be interpreted as coefficients of a trilinear matching tensor; they depend on the unknown motions' coefficients, and are given in table 1. In the following, we sometimes call this the *calibration tensor*. It is somewhat related to the *homography tensor* derived in [18]. Among the  $3 \cdot 3 \cdot 3 = 27$  coefficients of the calibration tensor, 8 are always zero and among the remaining 19, there are 6 pairs of identical ones. The columns of table 1 are interpreted as follows: the  $C_i$  are trilinear products of point coordinates and the  $V_i$  are the associated coefficients of the tensor. The following equation is thus equivalent to (1):

$$\sum_{i=1}^{13} C_i V_i = 0 \quad . \quad (2)$$

Given triplets of points  $\mathbf{Q}$ ,  $\mathbf{Q}'$  and  $\mathbf{Q}''$  for at least 12 pixels, we may compute the trilinear tensor up to an unknown scale  $\lambda$  by solving a system of linear equations of type (2). Note that we have verified using simulated data, that we indeed can obtain a unique solution (up to scale) for the tensor. The main problem is then that of extracting the motion parameters from the calibration tensor. In [21] we give a simple algorithm for doing so<sup>4</sup>. Once the motions are determined, the approach described above can be readily applied to compute the camera rays and thus to finalize the calibration.

*The special case of central cameras.* It is worthwhile to specialize the calibration concept to the case of central cameras (but which are otherwise general, i.e. not perspective). A central camera can already be calibrated from two views. Let  $\mathbf{Z}$  be the homogeneous coordinates of the optical center (in the frame associated with the object's first position). We have the following collinearity constraint:

$$\begin{vmatrix} Z_1 Q_1 & R_{11}' Q_1' + R_{12}' Q_2' + t_1' Q_3' \\ Z_2 Q_2 & R_{21}' Q_1' + R_{22}' Q_2' + t_2' Q_3' \\ Z_3 Q_3 & Q_3' \end{vmatrix} = \mathbf{Q}'^T \begin{pmatrix} R_{21}' Z_3 & -R_{22}' Z_3 & R_{22}' Z_2 - R_{21}' Z_1 \\ R_{22}' Z_3 & R_{21}' Z_3 & -R_{21}' Z_2 - R_{22}' Z_1 \\ Z_3 t_2' - Z_2 Z_1 - Z_3 t_1' & Z_2 t_1' - Z_1 t_2' & \end{pmatrix} \mathbf{Q} = 0$$

The bifocal calibration tensor in this equation is a  $3 \times 3$  matrix and somewhat similar to a fundamental or essential matrix. It can be estimated linearly from calibration points associated with 8 pixels or more. It is of rank 2 and its right null vector is the optical center  $\mathbf{Z}$ , which is thus easy to compute. Once this is done, the camera ray for a pixel can be determined e.g. by joining  $\mathbf{Z}$  and  $\mathbf{Q}$ .

<sup>4</sup> This is similar, though more complicated than extracting (ego-)motion of perspective cameras from the classical essential matrix [9].

*The special case of a linear calibration object.* This is equally worthwhile to investigate. We propose an algorithm in [21], which works but is more complicated than the algorithm for general calibration objects.

## 4 Generic Calibration Concept for 3D Cameras

This and the next section describe our main contributions. We extend the concept described in §3 to the case of cameras living in 3-space. We first deal with the most general case: non-central cameras and 3D calibration objects.

In case of **known motion**, two views are sufficient to calibrate, and the procedure is equivalent to that outlined in §3, cf. [8]. In the following, we consider the practical case of **unknown motion**. Input are now, for each pixel, three 3D points  $\mathbf{Q}$ ,  $\mathbf{Q}'$  and  $\mathbf{Q}''$ , given by 4-vectors of homogeneous coordinates, relative to the calibration object's local coordinate system. Again, we adopt the coordinate system associated with the first image as global coordinate frame. The object's motion for the other two images is given by  $3 \times 3$  rotation matrices  $\mathbf{R}'$  and  $\mathbf{R}''$  and translation vectors  $\mathbf{t}'$  and  $\mathbf{t}''$ . With the correct motion estimates, the aligned points must be collinear. We stack their coordinates in the following  $4 \times 3$  matrix:

$$\begin{pmatrix} Q_1 & R'_{11}Q'_1 + R'_{12}Q'_2 + R'_{13}Q'_3 + t'_1Q'_4 & R''_{11}Q''_1 + R''_{12}Q''_2 + R''_{13}Q''_3 + t''_1Q''_4 \\ Q_2 & R'_{21}Q'_1 + R'_{22}Q'_2 + R'_{23}Q'_3 + t'_2Q'_4 & R''_{21}Q''_1 + R''_{22}Q''_2 + R''_{23}Q''_3 + t''_2Q''_4 \\ Q_3 & R'_{31}Q'_1 + R'_{32}Q'_2 + R'_{33}Q'_3 + t'_3Q'_4 & R''_{31}Q''_1 + R''_{32}Q''_2 + R''_{33}Q''_3 + t''_3Q''_4 \\ Q_4 & & Q''_4 \end{pmatrix} \quad (3)$$

The collinearity constraint means that this matrix must be of rank less than 3, which implies that all sub-determinants of size  $3 \times 3$  vanish. There are 4 of them, obtained by leaving out one row at a time. Each of these corresponds to a trilinear equation in point coordinates and thus to a trifocal calibration tensor whose coefficients depend on the motion parameters.

Table 2 gives the coefficients of the first two calibration tensors (all 4 are given in the appendix of [21]). For both, 34 out of 64 coefficients are always zero. One may observe that the two tensors share some coefficients, e.g.  $V_8 = W_1 = R'_{31}$ .

The tensors can be estimated by solving linear equation system, and we verified using simulated random experiments that in general unique solutions (up to scale) are obtained, if 3D points for sufficiently many pixels (29 at least) are available. In the following, we give an algorithm for computing the motion parameters. Let  $V'_i = \lambda V_i$  and  $W'_i = \mu W_i$ ,  $i = 1 \dots 37$  be the estimated tensors (up to scale). The algorithm proceeds as follows.

1. Estimate scale factors:  $\lambda = \sqrt{V'^2_8 + V'^2_9 + V'^2_{10}}$  and  $\mu = \sqrt{W'^2_1 + W'^2_2 + W'^2_3}$ .
2. Compute  $V_i = \frac{V'_i}{\lambda}$  and  $W_i = \frac{W'_i}{\mu}$ ,  $i = 1 \dots 37$
3. Compute  $\mathbf{R}'$  and  $\mathbf{R}''$ :

$$\mathbf{R}' = \begin{pmatrix} -W_{15} & -W_{16} & -W_{17} \\ -V_{15} & -V_{16} & -V_{17} \\ V_8 & V_9 & V_{10} \end{pmatrix} \quad \mathbf{R}'' = \begin{pmatrix} W_{18} & W_{19} & W_{20} \\ V_{18} & V_{19} & V_{20} \\ -V_{11} & -V_{12} & -V_{13} \end{pmatrix} \quad .$$

They will not be orthonormal in general. We “correct” this as shown in [21].

4. Compute  $\mathbf{t}'$  and  $\mathbf{t}''$  by solving a straightforward linear least squares problem, which is guaranteed to have a unique solution, see [21] for details.

Using simulations, we verified that the algorithm gives a unique and correct solution in general.

$i$	$C_i$	$V_i$	$W_i$
1	$Q_1 Q'_1 Q''_4$	0	$R'_{31}$
2	$Q_1 Q'_2 Q''_4$	0	$R'_{32}$
3	$Q_1 Q'_3 Q''_4$	0	$R'_{33}$
4	$Q_1 Q'_4 Q''_1$	0	$-R'_{31}$
5	$Q_1 Q'_4 Q''_2$	0	$-R'_{32}$
6	$Q_1 Q'_4 Q''_3$	0	$-R'_{33}$
7	$Q_1 Q'_4 Q''_4$	0	$t'_3 - t''_3$
8	$Q_2 Q'_1 Q''_4$	$R'_{31}$	0
9	$Q_2 Q'_2 Q''_4$	$R'_{32}$	0
10	$Q_2 Q'_3 Q''_4$	$R'_{33}$	0
11	$Q_2 Q'_4 Q''_1$	$-R'_{31}$	0
12	$Q_2 Q'_4 Q''_2$	$-R'_{32}$	0
13	$Q_2 Q'_4 Q''_3$	$-R'_{33}$	0
14	$Q_2 Q'_4 Q''_4$	$t'_3 - t''_3$	0
15	$Q_3 Q'_1 Q''_4$	$-R'_{21}$	$-R'_{11}$
16	$Q_3 Q'_2 Q''_4$	$-R'_{22}$	$-R'_{12}$
17	$Q_3 Q'_3 Q''_4$	$-R'_{23}$	$-R'_{13}$
18	$Q_3 Q'_4 Q''_1$	$R'_{21}$	$R'_{11}$
19	$Q_3 Q'_4 Q''_2$	$R'_{22}$	$R'_{12}$
20	$Q_3 Q'_4 Q''_3$	$R'_{23}$	$R'_{13}$
21	$Q_3 Q'_4 Q''_4$	$t''_2 - t'_2$	$t''_1 - t'_1$
22	$Q_4 Q'_1 Q''_1$	$R'_{21} R'_{31} - R'_{22} R'_{31}$	$R'_{11} R'_{31} - R'_{12} R'_{31}$
23	$Q_4 Q'_1 Q''_2$	$R'_{21} R'_{32} - R'_{22} R'_{31}$	$R'_{11} R'_{32} - R'_{12} R'_{31}$
24	$Q_4 Q'_1 Q''_3$	$R'_{21} R'_{33} - R'_{22} R'_{31}$	$R'_{11} R'_{33} - R'_{13} R'_{31}$
25	$Q_4 Q'_1 Q''_4$	$R'_{21} t'_3 - R'_{31} t''_2$	$R'_{11} t'_3 - R'_{31} t''_1$
26	$Q_4 Q'_2 Q''_1$	$R'_{22} R'_{31} - R'_{21} R'_{32}$	$R'_{12} R'_{31} - R'_{11} R'_{32}$
27	$Q_4 Q'_2 Q''_2$	$R'_{22} R'_{32} - R'_{22} R'_{32}$	$R'_{12} R'_{32} - R'_{12} R'_{32}$
28	$Q_4 Q'_2 Q''_3$	$R'_{22} R'_{33} - R'_{23} R'_{32}$	$R'_{12} R'_{33} - R'_{13} R'_{32}$
29	$Q_4 Q'_2 Q''_4$	$R'_{22} t'_3 - R'_{32} t''_2$	$R'_{12} t'_3 - R'_{32} t''_1$
30	$Q_4 Q'_3 Q''_1$	$R'_{23} R'_{31} - R'_{21} R'_{33}$	$R'_{13} R'_{31} - R'_{11} R'_{33}$
31	$Q_4 Q'_3 Q''_2$	$R'_{23} R'_{32} - R'_{22} R'_{33}$	$R'_{13} R'_{32} - R'_{12} R'_{33}$
32	$Q_4 Q'_3 Q''_3$	$R'_{23} R'_{33} - R'_{23} R'_{33}$	$R'_{13} R'_{33} - R'_{13} R'_{33}$
33	$Q_4 Q'_3 Q''_4$	$R'_{23} t'_3 - R'_{33} t''_2$	$R'_{13} t'_3 - R'_{33} t''_1$
34	$Q_4 Q'_4 Q''_1$	$R'_{31} t'_2 - R'_{21} t'_3$	$R'_{31} t'_1 - R'_{11} t'_3$
35	$Q_4 Q'_4 Q''_2$	$R'_{32} t'_2 - R'_{22} t'_3$	$R'_{32} t'_1 - R'_{12} t'_3$
36	$Q_4 Q'_4 Q''_3$	$R'_{33} t'_2 - R'_{23} t'_3$	$R'_{33} t'_1 - R'_{13} t'_3$
37	$Q_4 Q'_4 Q''_4$	$t'_2 t''_3 - t'_3 t''_2$	$t'_1 t''_3 - t''_1 t'_3$

**Table 2.** Coefficients of two trifocal calibration tensors for a general 3D camera.

## 5 Variants of the Calibration Concept

Analogously to the case of 2D cameras, cf. §3, we developed important specializations of our calibration concept, for central cameras and planar calibration objects. We describe them very briefly; details are given in [21].

*Central cameras.* In this case, two images are sufficient. Let  $\mathbf{Z}$  be the optical center (unknown). By proceeding as in §3, we obtain 4 bifocal calibration tensors of size  $4 \times 4$  and rank 2, that are somewhat similar to fundamental matrices. One of them is shown here:

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ R'_{31} Z_4 & R'_{32} Z_4 & R'_{33} Z_4 & -Z_3 + Z_4 t'_3 \\ -R'_{21} Z_4 & -R'_{22} Z_4 & -R'_{23} Z_4 & Z_2 - Z_4 t'_2 \\ R'_{21} Z_3 - R'_{31} Z_2 & R'_{22} Z_3 - R'_{32} Z_2 & R'_{23} Z_3 - R'_{33} Z_2 & Z_3 t'_2 - Z_2 t'_3 \end{pmatrix}.$$

It is relatively straightforward to extract the motion parameters and the optical center from these tensors.

*Non-central cameras and planar calibration objects.* The algorithm for this case is rather more complicated and not shown here. Using simulations, we proved that we obtain a unique solution in general.

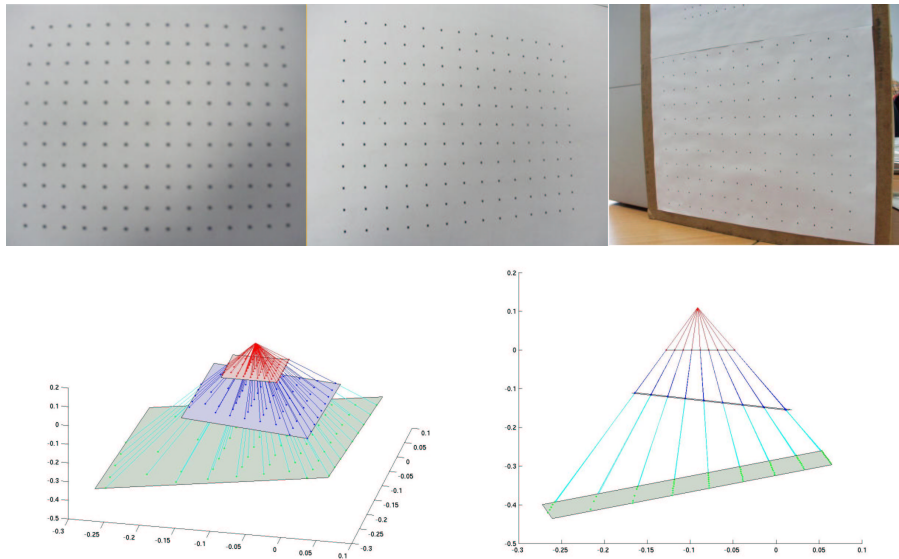


*Central cameras and planar calibration objects.* As with non-central cameras, we already obtain constraints on the motion parameters (and the optical center) from two views of the planar object. In this case however, the associated calibration tensors do not contain sufficient information in order to uniquely estimate the motion and optical center. This is not surprising: even in the very restricted case of **perspective** cameras with 5 intrinsic parameters, two views of a planar calibration object do not suffice for calibration [20, 26]. We thus developed an algorithm working with three views [21]. It is rather complicated, but was shown to provide unique solutions in general.

## 6 Experimental Evaluation

As mentioned previously, we verified each algorithm using simulated random experiments. This was first done using noiseless data. We also tested our methods using noisy data and obtained satisfying results. A detailed quantitative analysis remains yet to be carried out.

We did various experiments with real images, using a 3M-Pixel digital camera with moderate optical distortions, a camera with a fish-eye lens and “home-made” catadioptric systems consisting of a digital camera and various curved off-the-shelf mirrors. We used planar calibration objects consisting of black dots or squares on white paper. Figure 3 shows three views taken by the digital camera.

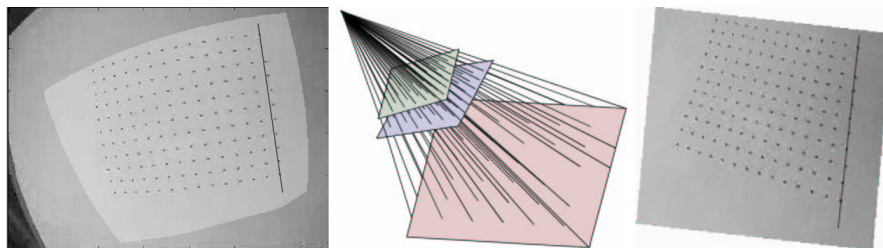


**Fig. 3.** Top: images of 3 boards of different sizes, captured by a digital camera. Bottom: two views of the calibrated camera rays and estimated pose of the calibration boards.

Dots/corners were extracted using the Harris detector. Matching of these image points to points on calibration objects was done semi-automatically. This gives calibration points for a sparse set of pixels per image, and in general there will be few, if any, pixels for which we get a calibration point in every view! We thus take into account the continuity assumption mentioned in §2. For every image, we compute the convex hull of the pixels for which calibration points were extracted. We then compute the intersection of the convex hulls over all three views, and henceforth only consider pixels inside that region. For every such pixel in the first image we estimate the calibration points for the second and third images using the following interpolation scheme: in each of these images, we determine the 4 closest extracted calibration points. We then compute the homography between these pixels and the associated calibration points on the planar object. The calibration point for the pixel of interest is then computed using that homography.

On applying the algorithm for central cameras (cf. §5), we obtained the results shown in figure 3. The bottom row shows the calibrated camera rays and the pose of the calibration objects, given by the estimated motion parameters. It is difficult to evaluate the calibration quantitatively, but we observe that for every pixel considered, the estimated motion parameters give rise to nearly perfectly collinear calibration points. Note also, cf. the bottom right figure, that radial distortion is correctly modeled: the camera rays are setwise coplanar, although the corresponding sets of pixels in the image are not perfectly collinear.

The same experiment was performed for a fish-eye lens, cf. figure 4. The result is slightly worse – aligned calibration points are not always perfectly collinear. This experiment is preliminary in that only the central image region has been calibrated (cf. figure 4), due to the difficulty of placing planar calibration objects that cover the whole field of view.

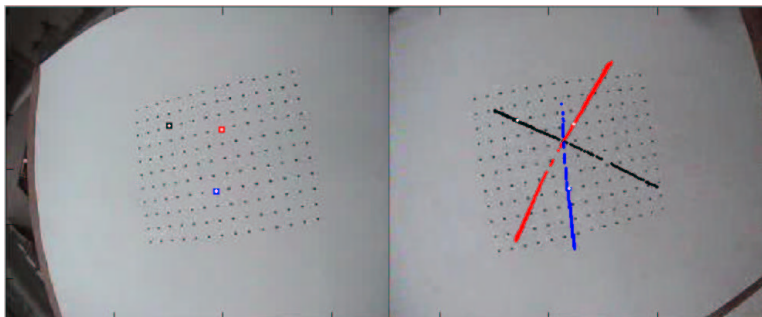


**Fig. 4.** Left: one of 3 images taken by the fish-eye lens (in white the area that was calibrated). Middle: calibrated camera rays and estimated pose of calibration objects. Right: image from the left after distortion correction, see text.

Using the calibration information, we carried out two sample applications, as described in the following. The first one consists in correcting non-perspective distortions: calibration of the central camera model gives us a bunch of rays passing through a single point. We may cut these rays by a plane; at each intersection with a camera ray, we “paint” the plane with the “color” observed by the pixel

associated with the ray in some input image. Using the same homography-based interpolation scheme as above, we can thus create a “densely” colored plane, which is nothing else than the image plane of a distortion-corrected perspective image. See figure 4 for an example. This model-free distortion correction scheme is somewhat similar to the method proposed in [5].

Another application concerns (ego-) motion and epipolar geometry estimation. Given calibration information, we can estimate relative camera pose (or motion), and thus epipolar geometry, from two or more views of an unknown object. We developed a motion estimation method similar to [15] and applied it to two views taken by the fish-eye lens. The epipolar geometry of the two views can be computed and visualized as follows: for a pixel in the first view, we consider its camera ray and determine all pixels of the second view whose rays (approximately) intersect the first ray. These pixels form the “epipolar curve” associated with the original pixel. An example is shown in figure 5. The estimated calibration and motion also allow of course to reconstruct objects in 3D (see [21] for examples).



**Fig. 5.** Epipolar curves for three points. These are not straight lines, but intersect in a single point, since we here use the central camera model.

## 7 Discussion

The algorithm for central cameras seems to work fine, even with the minimum input of 3 views and a planar calibration object. Experiments with non-central catadioptric cameras however did so far not give satisfying results. One reason for poor stability of the non-central method is the way we currently obtain our input (homography-based interpolation of calibration points). We also think that the general algorithm, which is essentially based on solving linear equations, can only give stable results with minimum input (3 views) if the considered camera is clearly non-central. By this, we mean that there is not any point that is “close” to all camera rays; the general algorithm does not work for perspective cameras, but for multi-stereo systems consisting of sufficiently many cameras<sup>5</sup>.

<sup>5</sup> Refer to the appendix of [21] on the feasibility of the general calibration method for stereo systems consisting of three or more central cameras.

We propose several ideas for overcoming these problems. Most importantly, we probably need to use several to many images for a stable calibration. We have developed bundle adjustment formulations for our calibration problem, which is not straightforward: the camera model is of discrete nature and does not directly allow to handle sub-pixel image coordinates, which are for example needed in derivatives of a reprojection error based cost function. For initialization of the non-central bundle adjustment, we may use the (stabler) calibration results for the central model. Model selection may be applied to determine if the central or non-central model is more appropriate for a given camera. Another way of stabilizing the calibration might be the possible inclusion of constraints on the set of camera rays, such as rotational or planar symmetry, if appropriate.

Although we have a single algorithm that works for nearly all existing camera types, different cameras will likely require different designs of calibration objects, e.g. panoramic cameras vs. ones with narrow field of view. We stress that a single calibration can use images of different calibration objects; in our experiments, we actually use planar calibration objects of different sizes for the different views, imaged from different distances, cf. figure 3. This way, we can place them such that they do not “intersect” in space, which would give less stable results, especially for camera rays passing close to the intersection region. We also plan to use different calibration objects for initialization and bundle adjustment: initialization, at least for the central model, can be performed using the type of calibration object used in this work. As for bundle adjustment, we might then switch to objects with a much denser “pattern” e.g. with a coating consisting of randomly distributed colored speckles. Another possibility is to use a flat screen to produce a dense set of calibration points [8].

One comment on the difference between calibration and motion estimation: here, with 3 views of a *known* scene, we solve simultaneously for motion and calibration (motion is determined explicitly, calibration implicitly). Whereas once a (general) camera is calibrated, (ego-)motion can already be estimated from 2 views of an *unknown* scene [15]. Hence, although our method estimates motion directly, we consider it a calibration method.

## 8 Conclusions

We have proposed a theory and algorithms for a highly general calibration concept. As for now, we consider this mainly as a conceptual contribution: we have shown how to calibrate nearly any camera, using one and the same algorithm.

We already propose specializations that may be important in practice: an algorithm for central, though otherwise unconstrained cameras, is presented, as well as an algorithm for the use of planar calibration objects. Results of preliminary experiments demonstrate that the approach allows to calibrate central cameras without using any parametric distortion model.

We believe in our concept’s potential for calibrating cameras with “exotic” distortions – such as fish-eye lenses with hemispheric field of view or catadioptric cameras, especially non-central ones. We are working towards that goal, by developing bundle adjustment procedures to calibrate from multiple images,

and by designing better calibration objects. These issues could bring about the necessary stability to really calibrate cameras without any parametric model in practice. Other ongoing work concerns the extension of classical structure-from-motion tasks such as motion and pose estimation and triangulation, from the perspective to the general imaging model.

## References

1. E.H. Adelson, J.R. Bergen. The Plenoptic Function and the Elements of Early Vision. *Computational Models of Visual Processing*, MIT Press, 1991.
2. D.G. Aliaga. Accurate Catadioptric Calibration for Real-time Pose Estimation in Room-size Environments. *ICCV*, 127-134, 2001.
3. S. Baker, S. Nayar. A Theory of Catadioptric Image Formation. *ICCV*, 1998.
4. H. Bakstein, T. Pajdla. An overview of non-central cameras. *Proceedings of Computer Vision Winter Workshop*, Ljubljana, Slovenia, 2001.
5. P. Brand. Reconstruction tridimensionnelle d'une scène à partir d'une caméra en mouvement. PhD Thesis, Université Claude Bernard, Lyon, October 1995.
6. C. Geyer, K. Daniilidis. Paracatadioptric Camera Calibration. *PAMI*, 2002.
7. S.J. Gortler et al. The Lumigraph. *SIGGRAPH*, 1996.
8. M.D. Grossberg, S.K. Nayar. A general imaging model and a method for finding its parameters. *ICCV*, 2001.
9. R.I. Hartley, A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
10. R.A. Hicks, R. Bajcsy. Catadioptric Sensors that Approximate Wide-angle Perspective Projections. *CVPR*, pp. 545-551, 2000.
11. M. Levoy, P. Hanrahan. Light field rendering. *SIGGRAPH*, 1996.
12. J. Neumann, C. Fermüller, Y. Aloimonos. Polydioptric Camera Design and 3D Motion Estimation. *CVPR*, 2003.
13. T. Pajdla. Stereo with oblique cameras. *IJCV*, 47(1), 2002.
14. S. Peleg, M. Ben-Ezra, Y. Pritch. OmniStereo: Panoramic Stereo Imaging. *PAMI*, pp. 279-290, March 2001.
15. R. Pless. Using Many Cameras as One. *CVPR*, 2003.
16. S. Seitz. The space of all stereo images. *ICCV*, 2001.
17. C.C. Slama (editor). *Manual of Photogrammetry*. Fourth Edition, ASPRS, 1980.
18. A. Shashua, L. Wolf. Homography Tensors: On Algebraic Entities That Represent Three Views of Static or Moving Planar Points. *ECCV*, 2000.
19. H.-Y. Shum, A. Kalai, S.M. Seitz. Omnivergent Stereo. *ICCV*, 1999.
20. P. Sturm, S. Maybank. On Plane-Based Camera Calibration. *CVPR*, 1999.
21. P. Sturm, S. Ramalingam. A Generic Calibration Concept: Theory and Algorithms. Research Report 5058, INRIA, France, 2003.
22. R. Swaminathan, M.D. Grossberg, and S.K. Nayar. Caustics of Catadioptric Cameras. *ICCV*, 2001.
23. R. Swaminathan, M.D. Grossberg, S.K. Nayar. A perspective on distortions. *CVPR*, 2003.
24. Y. Wexler, A.W. Fitzgibbon, A. Zisserman. Learning epipolar geometry from image sequences. *CVPR*, 2003.
25. D. Wood et al. Multiperspective panoramas for cell animation. *SIGGRAPH*, 1997.
26. Z. Zhang. A flexible new technique for camera calibration. *PAMI*, 22(11), 2000.