CENTER FOR
MACHINE PERCEPTION

CZECH TECHNICAL
UNIVERSITY

REPRINT

Estimation of omnidirectional camera model from epipolar geometry

Branislav Mičušík and Tomáš Pajdla

{micusb1,pajdla}@cmp.felk.cvut.cz

Branislav Mičušík and Tomáš Pajdla. Estimation of omnidirectional camera model from epipolar geometry. In Procceedings of Conference on Computer Vision and Pattern Recognition, IEEE Computer Society, Madison, Wisconsin, USA, June 16-22 2003

Available at
ftp://cmp.felk.cvut.cz/pub/cmp/articles/micusik/Micusik-CVPR2003.pdf

Center for Machine Perception, Department of Cybernetics
Faculty of Electrical Engineering, Czech Technical University
Technická 2, 166 27 Prague 6, Czech Republic
fax +420 2 2435 7385, phone +420 2 2435 7637, www: http://cmp.felk.cvut.cz

Estimation of omnidirectional camera model from epipolar geometry *

Branislav Mičušík Tomáš Pajdla
Center for Machine Perception, Dept. of Cybernetics, Faculty of Elec. Eng.
Czech Technical University in Prague, Karlovo nám. 13, 121 35 Prague, Czech Rep.
{micusb1,pajdla}@cmp.felk.cvut.cz

Abstract

We generalize the method of simultaneous linear estimation of multiple view geometry and lens distortion, introduced by Fitzgibbon at CVPR 2001 [6], to an omnidirectional (angle of view larger than $180°$*) camera. The perspective camera is replaced by a linear camera with a spherical retina and a non-linear mapping of the sphere into the image plane. Unlike the previous distortion-based models, the new camera model is capable to describe a camera with an angle of view larger than* $180°$ *at the cost of introducing only one extra parameter. A suitable linearization of the camera model and of the epipolar constraint is developed in order to arrive at a Quadratic Eigenvalue Problem for which efficient algorithms are known. The lens calibration is done from automatically established image correspondences only. Besides rigidity, no assumptions about the scene are made (e.g. presence of a calibration object). We demonstrate the method in experiments with Nikon FC–E8 fish-eye converter for* COOLPIX *digital camera. In practical situations, the proposed method allows to incorporate the new omnidirectional camera model into* RANSAC *- a robust estimation technique.*

1. Introduction

Recently, a number of high quality, but cheap and widely available lenses with an angle of view larger that $180°$ appeared, e.g. Nikon FC–E8 fish-eye converter ($200) for COOLPIX digital camera or Sigma 8mm-f4-EX fish-eye lens. Omnidirectional cameras with such an angle of view have advantages in applications like surveillance, tracking, structure from motion, and navigation. This work is motivated by the use of omnidirectional cameras for structure from motion and 3D reconstruction.

Previous work on the estimation of a camera model with lens distortion falls into two basic groups. The first one includes methods which exploit prior knowledge about the scene such as the presence of calibration patterns [3, 12] or plumb line methods for classical [18] and omnidirectional cameras [4, 15]. The method developed in [5] relies on the assumption that a lens nonlinearity introduces specific higher-order correlation in the frequency domain.

The second group covers the methods which do not use knowledge about scene. This includes calibration methods from pure rotation for standard [6], and omnidirectional cameras [9, 17], and calibration from rotation and translation for standard cameras [6, 13]. All the listed methods calibrate cameras from point correspondences only.

In [6], Fitzgibbon dealt with the problem of nonlinear lens distortion in the context of camera self-calibration and structure from motion. He introduced a one-parameter model for the radial distortion and suggested an algorithm for simultaneous estimation of multiple view geometry and lens distortion from point correspondences. His model, however, cannot be directly used for omnidirectional cameras with view angle above $180°$ because it represents images by points in which rays intersect the image plane. The problem of image representation is visualized in Figure 1.

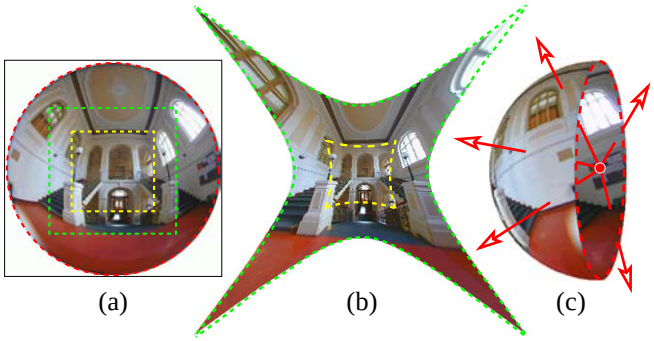
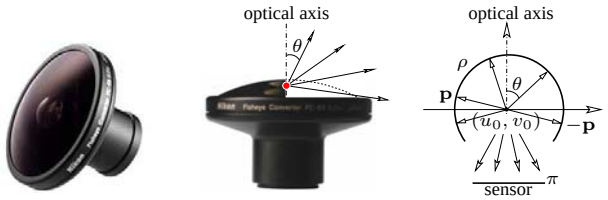


Figure 1: (a) Original image acquired by a Nikon FC–E8 fish-eye converter mounted on a COOLPIX digital camera with resolution 1600×1200 pixels. Yellow (smaller) and green (larger) dashed rectangles depict standard perspective cameras with the field of view $90°$ and $120°$ respectively. The red circle depicts the field of view $183°$. (b) The size of undistorted images using approach by Fitzgibbon [6] goes to infinity for the field of view approaching $180°$. Not all lines become straight since the model does not sufficiently capture lens nonlinearity. (c) The complete field of view can be represented on a sphere.

*This research was supported by the following projects: CTU 0209513, GAČR 102/01/0971, MSM 212300013, MŠMT Kontakt 22-2003-04, BENOGO–IST–2001–39184



(a) (b) (c)

Figure 2: (a) Nikon FC–E8 fish-eye converter. (b) The optical axis of the lens is marked by the dash dotted line and the optical center from which rays emanate is shown as the red dot. The angle between a ray and the optical axis is denoted by θ. (c) Notice, that the image taken by the lens to the planar sensor π can be represented by intersecting a spherical retina ρ with camera rays.

In this paper we generalize Fitzgibbon's method [6] to omnidirectional cameras, derive an appropriate omnidirectional model, incorporating lens nonlinearity, and find an algorithm for estimating the model from epipolar geometry. We assume that only point correspondences, information about the field of view of the lens, and its corresponding view angle are available.

The structure of the paper is the following. The full omnidirectional camera model is introduced, simplified and linearized in Section 2. An algorithm for model estimation from epipolar geometry is suggested in Sections 3 and 4. Experiments and summary are given in Sections 5 and 6.

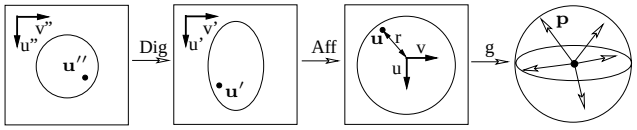
2. Omnidirectional camera model

For cameras with angle of view larger than $180°$, see Figure 2, images of all scene points $\mathbf{X}$ cannot be represented by intersections of camera rays with one image plane. Every line passing through an optical center intersects the image plane in one point, but on one such line two scene points can lie and they can be seen in an omnidirectional image at the same time, see rays $\mathbf{p}$ and $-\mathbf{p}$ in Figure 2c. For that reason, we will represent rays of the image as a set of unit vectors in $\mathbb{R}^3$ such that *one* vector corresponds just to *one* image of a scene point.

Let a scene point $\mathbf{X}$ be projected into $\mathbf{u}''$ in a sensor plane. Assume that the point $\mathbf{u}'' = (u'', v'')^\top$ in the sensor plane, see Figure 3a, and a point $\mathbf{u}' = (u', v')^\top$ in a digitized image, see Figure 3b, are related by an affine transformation, thus $\mathtt{A}\mathbf{u}' + \mathbf{t} = \mathbf{u}''$, where $\mathtt{A} \in \mathbb{R}^{2\times2}$, $\mathbf{t} \in \mathbb{R}^{2\times1}$. We propose the model of an omnidirectional camera in the form

$$\exists \alpha > 0 : \ \alpha g(\mathtt{A}\mathbf{u}' + \mathbf{t}) = \mathtt{P}\mathbf{X}, \qquad (1)$$

which assigns a point $\mathbf{X} \in \mathbb{R}^4$ of the scene to its digitized image $\mathbf{u}'$ by a perspective projection matrix $\mathtt{P} \in \mathbb{R}^{3\times4}$, a change of a coordinate system in the image $\mathtt{A}$, $\mathbf{t}$, and a nonlinear function $g : \mathbb{R}^2 \to \mathbb{R}^3$ defined in the sensor plane.



(a) (b) (c) (d)

Figure 3: Image formation and calibration of an omnidirectional camera. An image (a) on a sensor plane in a metric coordinate system is digitized to a digital image (b) related by an affine transformation, which can be estimated by transforming an ellipse to a circle (c) up to a rotation around the center of the circle and a scalar factor. From this precalibrated image, 3D vectors (d) are computed by applying a nonlinear function g.

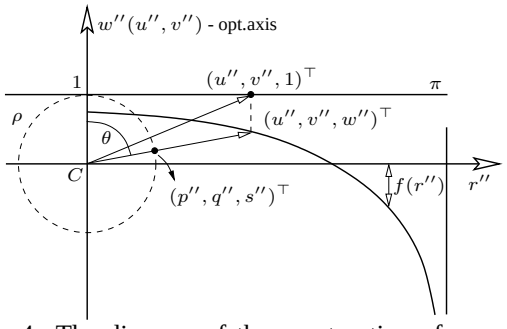


Figure 4: The diagram of the construction of mapping f from the sensor plane π to the spherical retina ρ. The point $(u'', v'', 1)^\top$ in the image plane π is transformed by $f(.)$ to $(u'', v'', w'')^\top$, then normalized to $(p'', q'', s'')^\top$ with unit length, and thus projected on the sphere ρ.

By a calibration of the omnidirectional camera we understand the determination of the matrix $\mathtt{A}$, the vector $\mathbf{t}$, and the nonlinear function g so that all vectors $g(\mathtt{A}\mathbf{u}' + \mathbf{t})$ (see Figure 3d) fulfill epipolar geometry and the projection equation (1).

Let us further assume that a nonlinear function g can be expressed as

$$g(u'', v'') = (u'', \ v'', \ f(u'', v''))^\top, \qquad (2)$$

where $f(\mathbf{u}'')$ is rotationally symmetric with respect to a point $(u_0'', v_0'')^\top$. The ray passing through $(u_0'', v_0'')^\top$ is the optical axis of the camera.

Function f can have various forms determined by lens construction [3, 10]. For instance, the model $\theta = a''r''$ holds approximately true for Nikon FC–E8 fish-eye lens, where a'' is a parameter, θ is the angle between a ray and the optical axis, $r'' = \sqrt{u''^2 + v''^2}$ is the radius of a point in the sensor plane with respect to $(u_0'', v_0'')^\top$, and it holds $f(\mathbf{u}'') = \frac{r''}{\tan\theta}$, see Figure 4. A more precise model calls for a nonlinear part. We use the division model

$$\theta = \frac{a''r''}{1 + b''r''^2}, \qquad r'' = \frac{a'' - \sqrt{a''^2 - 4b''\theta^2}}{2b''\theta}, \qquad (3)$$

where b'' is an additional parameter.

As can be seen in Figure 4, an undistorted ray represented by unit vector $\mathbf{p}''$ can be expressed as

$$\mathbf{p}'' = kg(\mathbf{u}'', a'', b'') = k \begin{pmatrix} u'' \\ v'' \\ w'' \end{pmatrix} = k \begin{pmatrix} \mathtt{A}\mathbf{u}' + \mathbf{t} \\ f(\mathbf{u}'', a'', b'') \end{pmatrix}$$

$$f(\mathbf{u}'', a'', b'') = \frac{r''}{\tan\theta} = \frac{r''}{\tan\frac{a''r''}{1+b''r''^2}}, \qquad k > 0. \quad (4)$$

The equation (4) captures the relationship between the point $\mathbf{u}'$ in the digitized image and the vector $\mathbf{p}''$ emanating from the optical center to a scene point $\mathbf{X}$.

2.1. Model simplification

In [6], it is shown that using a division model with *one parameter* leads to solving a Quadratic Eigenvalue Problem (QEP) [2, 16] for which efficient algorithms are available. The model (1) has more parameters (a, b, u_0, v_0, $\mathtt{A}$, $\mathbf{t}$). However, for omnidirectional cameras, the number of unknown parameters can be reduced to formulate the calibration as QEP. The reduction of parameters can be performed by taking into account that *i)* the view field is circular on the sensor plane and *ii)* the view angle is approximately known.

ad i) The vector $\mathbf{t}$ and the matrix $\mathtt{A}$ are obtained by transforming the view field ellipse to a circle up to a rotation $\mathtt{R} \in \mathbb{R}^{2\times 2}$ around the center of the circle and up to scale β. A transformed point $\mathbf{u}$ is related to the corresponding point on the sensor plane $\mathbf{u}''$ by $\beta\mathbf{u} = \mathtt{R}\mathbf{u}''$. Function f is rotationally symmetric

$$f(\mathbf{u}'', a'', b'') = f(\beta\mathtt{R}^{-1}\mathbf{u}, a'', b'') = f(\beta\mathbf{u}, a'', b'') = \\ = \frac{\beta r}{\tan\frac{\beta a'' r}{1+b''\beta^2 r^2}} = \frac{\beta r}{\tan\frac{ar}{1+br^2}} = \beta f(\mathbf{u}, a, b) \quad (5)$$

$$\mathbf{p}'' = k\begin{pmatrix} \mathbf{u}'' \\ f(\mathbf{u}'', a'', b'') \end{pmatrix} = k\beta \begin{pmatrix} \mathtt{R}^{-1}\mathbf{u} \\ f(\mathbf{u}, a, b) \end{pmatrix} = k\beta \begin{pmatrix} \mathtt{R}^{-1} \\ & 1 \end{pmatrix} \mathbf{p} \quad (6)$$

Equation (5) shows that unknown scale β is absorbed into parameters of the model and does not increase the number of the parameters. The reconstructed vector $\mathbf{p}$ is related to the vector $\mathbf{p}''$ by a rotation and a nonzero scale. The vector $\mathbf{p}$ represents a ray of the camera in an orthonormal coordinate system.

ad ii) From *i)*, the view angle θ_m, the radius R of the view field circle, and (3), parameter a can be expressed as $a = \frac{(1+bR^2)\theta_m}{R}$, what would lead, after linearization, to a *one parametric* model and QEP like in [6]. When assuming that θ_m and R are not known precisely, a *two parametric* model with a priori knowledge $a_0 = \frac{\theta_m}{R}$ is obtained.

The reduction of the number of parameters to one allows to estimate the model from 9 point RANSAC as in [6] and thus detect most of outliers in this preliminary test. Then, the two parametric model and the 15 point RANSAC can be used to obtain more precise model.

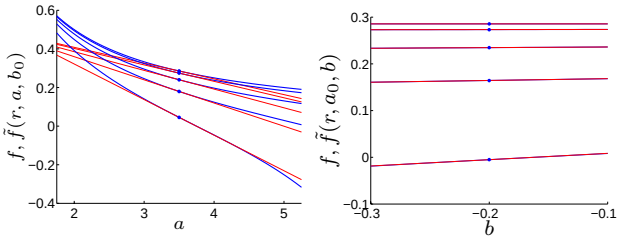


Figure 5: Comparison of the graph of original function f (thick blue curves) and its linearization $\tilde{f}$ (red lines). Left: Change of parameter a for fixed $b = -0.2$. Right: Change b for fixed $a = 3.5$. Graphs are for radii $r = 10$ (upper), $100, 200, 300, 435$ pxl (lower), where $r_{max} = 435pxl \sim 91.5°$. Blue dots mean points of the Taylor expansion: $a_0 = 3.5$, $b_0 = -0.2$. Parameters a, b were changed by $\pm 50\%$.

2.2. Linearization of the model

The model (6) does not lead directly to QEP but it can be linearized so that QEP is obtained. Function $f(r, a, b)$ is a two parametric nonlinear function, which can be expanded to Taylor series with respect to a and b at a_0 and b_0. By omitting the nonlinear part of the Taylor series we obtain function $\tilde{f}$, which is linear in a, b

$$\tilde{f}(r, a, b) = f(r, a_0, b_0) + f_a(r, a_0, b_0)(a - a_0) + \\ + f_b(r, a_0, b_0)(b - b_0), \quad (7)$$

where

$$f_a(r, a_0, b_0) = -\frac{r^2\left(1 + \left(\tan\frac{a_0 r}{1+b_0 r^2}\right)^2\right)}{\left(\tan\frac{a_0 r}{1+b_0 r^2}\right)^2 (1 + b_0 r^2)},$$

$$f_b(r, a_0, b_0) = -\frac{ar^2}{(1 + b_0 r^2)} f_a(r, a_0, b_0)$$

are partial derivatives of $f(r, a, b)$ with respect to a, resp. b.

Functions $\tilde{f}$ and f are shown for both parameters in Figure 5. As can be seen, $b_0 = 0$ can be chosen thanks to the linear character of $f(r, a_0, b)$. We assume that $a_0 = \frac{\theta_m}{R}$ is known precisely enough to use it as a point of Taylor series expansion.

3. Model estimation from epipolar geometry

The vector $\mathbf{p}$ can be written by using (4) and (7) as follows

$$\mathbf{p} = \left[\begin{pmatrix} u \\ v \\ f(.) - a_0 f_a(.) - b_0 f_b(.) + a f_a(.) + b f_b(.) \end{pmatrix}\right] \\ = \left[\begin{pmatrix} u \\ v \\ w \end{pmatrix} + a\begin{pmatrix} 0 \\ 0 \\ s \end{pmatrix} + b\begin{pmatrix} 0 \\ 0 \\ t \end{pmatrix}\right] \\ \simeq \mathbf{x} + a\mathbf{s} + b\mathbf{t}$$

where $\mathbf{x}$, $\mathbf{s}$ and $\mathbf{t}$ are known vectors computed from image coordinates. The epipolar constraint [7] for vectors $\mathbf{p}$ in the left and $\check{\mathbf{p}}$ in the right image that correspond to the same scene point reads as

$$\check{\mathbf{p}}^\top \mathtt{F}\mathbf{p} = 0$$
$$(\check{\mathbf{x}} + a\check{\mathbf{s}} + b\check{\mathbf{t}})^\top \mathtt{F}(\mathbf{x} + a\mathbf{s} + b\mathbf{t}) = 0$$

After arranging of unknown parameters into the vector $\mathbf{h}$ we obtain

$$(\mathtt{D}_1 + a\mathtt{D}_2 + a^2\mathtt{D}_3)\mathbf{h} = 0, \tag{8}$$

where matrices $\mathtt{D}_\mathtt{i}$ and and vector $\mathbf{h}$ are as follows

$$\mathtt{D}_1 = [\ u\check{u} \quad v\check{u} \quad w\check{u} \quad u\check{v} \quad v\check{v} \quad w\check{v} \quad u\check{w} \quad v\check{w} \quad w\check{w} \quad t\check{u} \quad t\check{v} \quad u\check{t} \quad v\check{t} \quad t\check{w} + w\check{t} \quad t\check{t}\ ]$$
$$\mathtt{D}_2 = [\ 0 \quad 0 \quad s\check{u} \quad 0 \quad 0 \quad s\check{v} \quad u\check{s} \quad v\check{s} \quad s\check{w} + w\check{s} \quad 0 \quad 0 \quad 0 \quad 0 \quad t\check{s} + s\check{t} \quad 0\ ]$$
$$\mathtt{D}_3 = [\ 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad s\check{s} \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0\ ]$$
$$\mathbf{h} = [\ f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \quad f_6 \quad f_7 \quad f_8 \quad f_9 \quad bf_3 \quad bf_6 \quad bf_7 \quad bf_8 \quad bf_9 \quad b^2 f_9\ ]^\top$$

Equation (8) represents a Quadratic Eigenvalue Problem (QEP) [2, 16], which can be solved by MATLAB using the function `polyeig`.

The vector $\mathbf{h}$ contains six dependent products bf_i. The parameter b can be determined from any of them. Since there is noise in data, we choose the one which best fits our model in terms of minimum error, as it will be explained further. From estimated a and b, real undistorted 3D vectors can be computed and used for determining the fundamental matrix $\mathtt{F}$ using one of standard algorithms [7]. The computation of $\mathtt{F}$ from undistorted vectors is more robust than using the first six elements of $\mathbf{h}$.

We determine the vector $\mathbf{p}$ according to (6) up to a rotation $\mathtt{R}$. The rotation $\mathtt{R}$ has no effect on the angle between rays and the optical axis. Therefore, we obtain an essential matrix (instead of a fundamental one) [7]. It allows us to determine the relative rotation and the translation direction of a pair of cameras.

Epipolar lines are represented by curves in original images. If we wanted to use the error by Hartley and Sturm [8] we would have to recompute 3D vectors of rays back to the image plane and determine distances from epipolar curves.

Since we have known parameters of model (6), and thus angles between rays and the optical axis, we have obtained a calibrated camera. We use the angle between a ray and an epipolar plane, instead of the distance of a point from its epipolar curve. Error for one pair of vectors has been given by Oliensis [11] as follows

$$\epsilon(\mathbf{p}, \check{\mathbf{p}}, \mathtt{F}) = A/2 - \sqrt{A^2/4 - B}, \tag{9}$$
$$A = \mathbf{p}^\top \mathtt{F}^\top \mathtt{F}\mathbf{p} + \check{\mathbf{p}}^\top \mathtt{F}\mathtt{F}^\top \check{\mathbf{p}}, \quad B = \left(\check{\mathbf{p}}^\top \mathtt{F}\mathbf{p}\right)^2, \tag{10}$$

where $\mathtt{F}$ is an essential matrix.

4. Algorithm

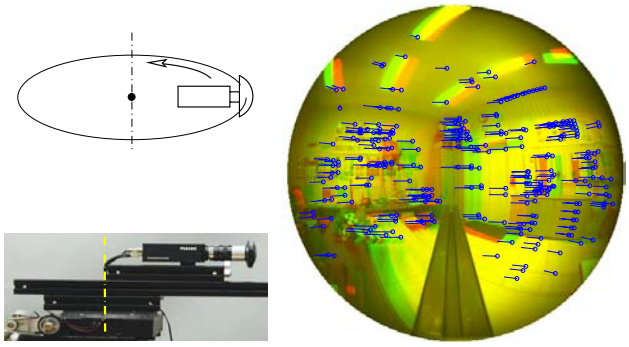
Algorithm for computing 3D rays and an essential matrix follows

1. Find an ellipse corresponding to the view field of the lens. Transform the image so that the ellipse becomes a circle, determine $\mathtt{A}$ and $\mathbf{t}$. Find correspondences $\{\mathbf{u} \leftrightarrow \check{\mathbf{u}}\}$ between two images.
2. Scale image points $\mathbf{u} := \mathbf{u}/1000$ to obtain better numerical stability. Choose first estimates $a_0 = \frac{R}{\theta_m}$ and $b_0 = 0$.
3. Create matrices $\mathtt{D}_1, \mathtt{D}_2, \mathtt{D}_3 \in \mathbb{R}^{N\times 15}$, N is the number of correspondences. Solve equation (8) with inverted QEP due to singularity of $\mathtt{D}_3$ [2]. Use MATLAB: $[\mathtt{H}\ \mathbf{a}] = \mathtt{polyeig}(\mathtt{D}_1^\top \mathtt{D}_3, \mathtt{D}_1^\top \mathtt{D}_2, \mathtt{D}_1^\top \mathtt{D}_1)$, $\mathtt{H}$ is a 15×30 matrix with columns $\mathbf{h}$, $\mathbf{a}$ is a 30×1 vector with elements $1/a$. Six possible solutions of b from last six elements of $\mathbf{h}$ appear.
4. Choose $a \neq 0$ and $a < 10$ (other solutions seem never be correct), 1–4 solutions remain. For every a there are 6 solutions of b. Create 3D rays from a and b and compute $\mathtt{F}$ using a standard method. The set of possible solutions $\{a_i \leftrightarrow b_{i,1\ldots6} \leftrightarrow \mathtt{F}_{i,1\ldots6}\}$ arises.
5. Compute the error (9) for all triplets $\{a \leftrightarrow b \leftrightarrow \mathtt{F}\}$ as a sum of errors for all correspondences. The triplet with the minimum error is the solution and a, b, and the essential matrix $\mathtt{F}$ are obtained .

5. Experiments

In this section, we apply the method to real data. Correspondences were obtained by the commercial program boujou [1]. Parameters of camera models and cameras trajectories (up to magnitudes of translation vectors), were estimated. Relative camera rotations and directions of translations used for trajectories estimations were computed from essential matrices [7]. For obtaining the magnitudes we would need to reconstruct the observed scene. It was not the task of this paper. Instead, we assumed unit length of translation vectors.

The first experiment shows a rotating omnidirectional camera, see Figure 6. The camera was mounted on a turntable such that the final trajectory of its optical center was circular. Images were acquired every 10°, 36 images in total. Three approaches to estimating the parameters a, b, and essential matrices $\mathtt{F}$ were used. The first approach used all correspondences and the essential matrix F was computed for every pair independently from a, b estimated for the given pair, see Figure 8a. The second approach estimates one $\hat{a}$ and one $\hat{b}$ as the median of all a, b's computed for every consecutive pair of images in the whole sequence,



(a) (b)

Figure 6: The optical center is rotated on a circle. (a) Nikon FC–E8 fish-eye converter is mounted on the PULNIX TM1001 digital camera with resolution 1017×1008 pixels. (Courtesy of J.Šivic.) (b) Correspondences between two images. Circles mark points in one image, lines join them to the matches in the next image. The images are superimposed in red and green channel.

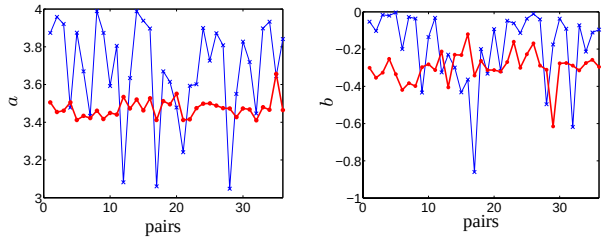
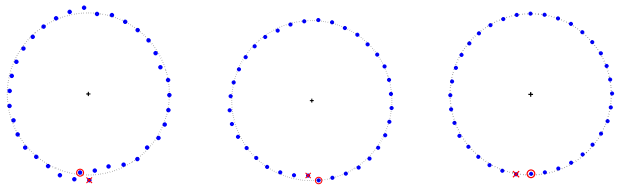


Figure 7: Simultaneous estimation of a and b for a circle sequence. Blue crosses depict the approach when all image correspondences are used, $\hat{a} = 3.80$, $\hat{b} = -0.16$. Red dots when RANSAC is used, $\hat{a} = 3.47$, $\hat{b} = -0.30$.

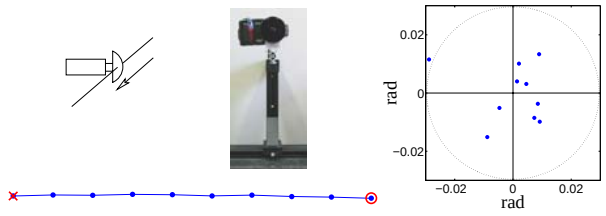
see Figure 7. Matrices F were then computed for each pair using the same $\hat{a}$, $\hat{b}$, see Figure 8b. The third approach differs from the second one such that a 9 point RANSAC as a pre-test to detect most of outliers and then a 15 point RANSAC were performed to compute the parameters a, b for every pair, see Figure 7 and estimated trajectory in Figure 8c. RANSAC leads to the best solution. A small error between the end and the start point appears. It is a good initial estimate for a bundle adjustment enforcing metrical constraints on the reconstruction and including lens nonlinearity.

The next experiment calibrates the omnidirectional camera from its translation in the direction perpendicular to the optical axis, see Figure 9, and in the direction along the optical axis, see Figure 10. Estimated trajectories are shown in Figures 9a and 10a. The angular differences between estimated and true motion directions for every pair are depicted in Figures 9b and 10b. In fact, the errors are approximately the same. The difference is under experimental error and therefore, in this case, the side and the forward motion perform similarly.



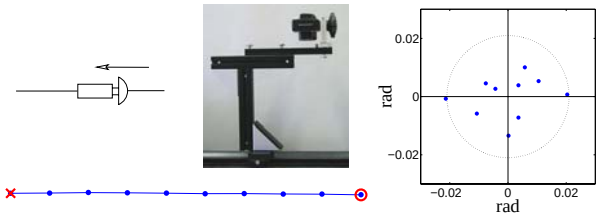
(a) (b) (c)

Figure 8: Motion estimation for a circle sequence. Red ○ depicts the start position, × depicts the end position. (a) Essential matrix F is computed from actual estimates of a and b. (b) F is computed from a, b determined from whole sequence. (c) F is computed from a, b determined from whole sequence using RANSAC.



(a) (b)

Figure 9: Side motion. Nikon FC–E8 fish-eye converter with COOLPIX digital camera with resolution 1600×1200 pixels was used.(a) On the left hand side, a diagram of the camera motion is depicted and on the right hand side a picture of the real setup is shown. The estimated trajectory is shown below the diagram. (b) Angular error between the direction of motion and the optical axis for each pair, and circle 3σ is plotted.

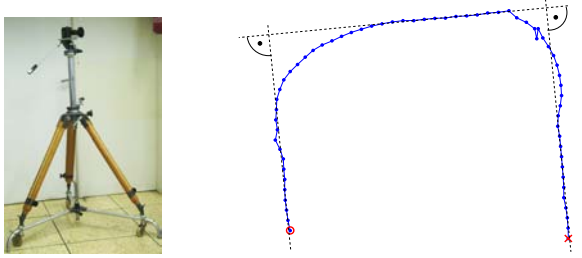


(a) (b)

Figure 10: Forward motion of the fish-eye lens. See Figure 9 for the explanation.

The next experiment shows the calibration from a general planar motion. Figure 11a shows a mobile tripod with an omnidirectional camera. Figure 11b shows an estimated trajectory. We made a U-shaped trajectory with right angles. Discontinuities of the trajectory were caused by hand driven motion of the mobile tripod. Naturally, they have no effect on the final estimate and the final trajectory has really right angles.

The last experiment, see Figure 12, applied our model and model introduced in [6] to an omnidirectional image. It can be seen that the model [6] does not sufficiently capture lens nonlinearity.



(a) (b)

Figure 11: General motion of Nikon FC–E8 fish-eye converter with COOLPIX digital camera. (a) Setup of the experiment. A mobile tripod with the camera. (b) The estimated trajectory.

Figure 12: Image with 120° angle of view reprojected to a plane. Left: approach by [6]. Right: our method. Notice that with our method all lines are straight.

6. Summary and Conclusions

The paper extended the non-linear lens model estimation from epipolar geometry to omnidirectional cameras with the angle of view larger than 180°. All steps leading to the full calibration of an omnidirectional camera were formulated. The model of the omnidirectional camera, the method for reduction of the number of parameters, and an algorithm for estimation of the parameters from the epipolar constraint was proposed. Experiments demonstrated that the method is useful for estimating parameters of a camera model and structure from motion with sufficient accuracy to be used as a starting point for a bundle adjustment enforcing metrical constraints on the reconstruction and including lens nonlinearity.

It is important to realize that it is the number of parameters, rather than the actual form of the non-linear function in the camera model, allowing to formulate the calibration process as QEP. We presented tests on one type of fish-eye converter but the method is applicable to a large class of omnidirectional cameras, e.g. fish-eye lenses or mirrors [14] possessing the central projection and providing view angle above 180°.

The important conclusion is that the suggested method allows us to incorporate our omnidirectional camera model into a robust estimation technique based on the 9 point followed by the 15 point RANSAC for outliers detection and essential matrix computation.

References

[1] 2d3 Ltd. *Boujou*. 2000. `http://www.2d3.com`.

[2] Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst. *Templates for the Solution of Algebraic Eigenvalue Problems : A Practical Guide*. SIAM, Philadelphia, 2000.

[3] H. Bakstein and T. Pajdla. Panoramic mosaicing with a 180° field of view lens. In *Proc. of the IEEE Workshop on Omnidirectional Vision*, pages 60–67, 2002.

[4] C. Bräuer-Burchardt and K. Voss. A new algorithm to correct fish-eye- and strong wide-angle-lens-distortion from single images. In *Proc. ICIP*, pages 225–228, 2001.

[5] H. Farid and A. C. Popescu. Blind removal of image nonlinearities. In *Proc. ICCV*, volume 1, pages 76–81, 2001.

[6] A. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proc. CVPR*, volume 1, pages 125–132, 2001.

[7] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK, 2000.

[8] R. I. Hartley and P. Sturm. Triangulation. *Computer Vision and Image Understanding: CVIU*, 68(2):146–157, 1997.

[9] S. Kang. Catadioptric self-calibration. In *Proc. CVPR*, volume 1, pages 201–207, June 2000.

[10] J. Kumler and M. Bauer. *Fisheye lens designs and their relative performance*. `http://www.coastalopt.com/fisheyep.pdf`.

[11] J. Oliensis. Exact two–image structure from motion. *PAMI*, 24(12):1618–1633, 2002.

[12] S. Shah and J. K. Aggarwal. Intrinsic parameter calibration procedure for a (high distortion) fish-eye lens camera with distortion model and accuracy estimation. *Pattern Recognition*, 29(11):1775–1788, November 1996.

[13] G. P. Stein. Lens distortion calibrating using point correspondences. In *Proc. CVPR*, pages 602–609, 1997.

[14] T. Svoboda and T. Pajdla. Epipolar geometry for central catadioptric cameras. *IJCV*, 49(1):23–37, 2002.

[15] R. Swaminathan and S. K. Nayar. Nonmetric calibration of wide-angle lenses and polycameras. *PAMI*, 22(10):1172–1178, 2000.

[16] F. Tisseur and K. Meerbergen. The quadratic eigenvalue problem. *SIAM Review*, 43(2):235–286, 2001.

[17] Y. Xiong and K. Turkowski. Creating image-based VR using a self-calibrating fisheye lens. In *Proc. CVPR*, pages 237–243, 1997.

[18] Z. Zhang. On the epipolar geometry between two images with lens distortion. In *Proc. ICPR*, pages 407–411, 1996.