Are two rotational flows sufficient to calibrate a smooth non-parametric sensor?

Etienne Grossmann,
visiting Dept. IRO
U. de Montréal
etienne@isr.ist.utl.pt

Eun-Joo Lee,
Computer Science Dept.
U. of Kentucky
elee3@csr.uky.edu

Peter Hislop,
Dept. of Mathematics
U. of Kentucky
hislop@ms.uky.edu

David Nistér,
Center for Visualization
U. of Kentucky
dnister@engr.uky.edu

Henrik Stewénius
Center for Visualization
U. of Kentucky
stewe@maths.lth.se *

Abstract

We present an attempt to determine whether the shape of a generic central-projection camera, such as the eye of an insect or a log-polar camera, can be determined from two motion flows resulting from purely rotational motions with non-collinear axes. Our first contribution is to write the smooth non-parametric calibration problem as a differential equation. It is unclear at present whether this problem has unique solution, up to an orthogonal transformation. Our second contribution is a discretized version of this smooth problem, for which we give a calibration algorithm - a third contribution. Using this algorithm, we explore numerically the properties of the discrete self-calibration problem, giving some insight on the nature of the problem. We show examples of successful self-calibration, but cannot give a definite affirmative answer to the question in the title.

1. Introduction

Knowing the geometry of an imaging sensor is a critical component of many computer vision tasks. Accordingly, a large body of research has been devoted to this subject. Most research deals with the common case of a projective camera, in which pixel coordinates and Euclidean coordinates in the image plane are related by a homography [6]. A sensor is then characterized by a mapping F from sensor pixel coordinates (e.g. in $[-1,1]^2$) to optic rays, i.e. elements of the unit sphere S^2. Most work on calibration assumes a model in which F is of the form

$$F(x) \sim K \begin{bmatrix} x \\ 1 \end{bmatrix}, \qquad (1)$$

where K is an upper triangular matrix and $\sim$ denotes collinearity. Hartley [5] shows that such a projective camera can be calibrated from two finite rotations.

The model in Eq. (1) can be improved to deal with radial distortion, by considering that pixel and Euclidean coordinates are related by a polynomial or rational fraction mapping [9, 7]. The calibration of other sensors, such as catadioptric cameras [2, 3] has also been considered. All these approaches consider parametric mappings from pixel to Euclidean coordinates.

Our work differs from these approaches by considering sensors where the map from pixel coordinates to Euclidean coordinates may be any diffeomorphism F, rather than a mapping from a more restricted class. The only constraint are thus smoothness of the mapping, and the existence of a central projection point.

The increased availability of such generic sensors has motivated research on their self-calibration. Like the present work, recent results [13, 16, 11, 14, 12] concern known types of motion: pure translations or rotations, finite or infinitesimal.

In this paper, we focus on the case of two motion flows (infinitesimal motions) [8], corresponding to pure rotational motions of the camera around two non-collinear axes passing through the center of projection. Given two flows like that in Fig. 1 right, we try to determine a mapping from the image plane into itself (for example the one at the left of the figure) such that the flow, transformed into the coordinate system defined by the mapping (middle of the figure), is compatible with a pure rotational motion.

Our main contribution is the novel formulation of the problem of calibrating a generic central projection sensor from two flows, together with an algorithm to solve it. With respect to [12] we get a Euclidean (vs. projective) calibration from two (vs. three) dense (vs. sparse) flows. With re-

*The authors are thankful for the detailed reviews and for the support of NSF grant DMS-050378.

spect to [14], we our algorithm uses information from two (vs. three) motions only.

2. Notation and smooth problem formulation

We choose to represent a sensor, i.e. a mapping from the oriented[1] projective plane [15] to itself, by a diffeomorphism $F : \mathbb{R}^3 \to \mathbb{R}^3$ s.t. for all $\lambda > 0$, and for all point X represented by its homogeneous coordinates, one has $F(\lambda X) = \lambda F(X)$ and $\|F(X)\| = \|X\|$. One advantage of this sensor representation is that the Euclidean velocity induced by an angular velocity $\omega \in \mathbb{R}^3$ is simply:

$$\omega \times F(X).$$

We will plainly call a mapping F "a sensor".

The input for our problem consists in two vector fields expressed in sensor coordinates, which we will represent by two smooth functions V_1, V_2, defined $\mathbb{R}^3 \to \mathbb{R}^3$ s.t. $V_i(X)^\top X = 0$ and $V_i(\lambda X) = \lambda V_i(X)$ for all $X \in \mathbb{R}^3$ and $\lambda \in \mathbb{R}^+$. That is, the V_i are vector fields on the unit sphere. In order to represent a 2D vector field v defined on $[-1,1]^2$ by a 3D vector field V (we omit the index i when considering a single field), we use the usual homogeneous coordinate mapping T

$$T(x) = \frac{1}{\sqrt{1+x^\top x}} \begin{bmatrix} x \\ 1 \end{bmatrix} \tag{2}$$

from $\mathbb{R}^2$ to the upper half-unit sphere. By chain-differentiation, one easily sees that the fields V and v are related by

$$V(T(x)) = \mathcal{D}_x T(x)\, v(x). \tag{3}$$

Other similar relations between velocities expressed in different coordinate systems will be used below, e.g. in Eqs. (4, 5).

Going back to our calibration problem, we know that there exist angular velocity vectors ω_1^*, $\omega_2^* \in \mathbb{R}^3$, distinct and nonzero and a sensor F^*, s.t. for all X, the following equalities hold

$$\begin{aligned} \mathcal{D}_X F^*(X) \cdot V_1(X) &= \omega_1^* \times F^*(X) \quad \text{and} \\ \mathcal{D}_X F^*(X) \cdot V_2(X) &= \omega_2^* \times F^*(X). \end{aligned}$$

These equations relate velocities expressed in Euclidean coordinates (rhs) with velocities in sensor coordinates (V_i).

Question 1: What is the set of triplets (ω_1, ω_2, F) consisting of two distinct nonzero vectors of $\mathbb{R}^3$ and a sensor, s.t., for all $X \in \mathbb{R}^3$, one has

$$\begin{aligned} \mathcal{D}_X F(X) \cdot V_1(X) &= \omega_1 \times F(X), \quad \text{and} & (4) \\ \mathcal{D}_X F(X) \cdot V_2(X) &= \omega_2 \times F(X). & (5) \end{aligned}$$

The answer to this question is given in [12]: the set consist only in

$$\mathcal{S} = \left\{ (R\omega_1^*, R\omega_2^*, R \circ F) \mid R \text{ is } 3 \times 3 \text{ and, } R^\top R = I_3 \right\}, \tag{6}$$

where I_3 is the identity matrix. That is, the solution $(\omega_1^*, \omega_2^*, F^*)$ is unique up to an orthogonal transformation. It is clear that one cannot get a smaller solution since, if ω_1, ω_2 and F solve Eqs. (4, 5), then, by virtue of the relation $(R\omega) \times (RF) = R(\omega \times F)$, one has $R \cdot \mathcal{D}_X F(X) \cdot V_i(X) = (R\omega_i) \times (RF(X))$, so that $G = R \circ F$ and $\omega_i' = R\omega_i$ also solve Eqs. (4, 5), for any orthogonal transformation (identified with an orthogonal matrix) R.

This indeterminacy allows to restrict our attention, with no loss of generality, to solutions of the form $\omega_1 = [0,\, 0,\, \theta_1]$ and $\omega_2 = [0,\, \gamma_1,\, \gamma_2]$, for some unknown θ_1, γ_1 and γ_2.

Question 1 is about calibrating from flows observed *everywhere* on the sphere. In practice, few sensors are truly omnidirectional and it is more realistic to ask whether two flows, observed only on a "little part" of the sphere, allow to calibrate a sensor.

Question 2: Same as Question 1, but assuming that the V_i and F are only defined on a set O that is a simply connected open set on the unit sphere.

In this paper, we do not attempt to answer this question by a proof, but instead:

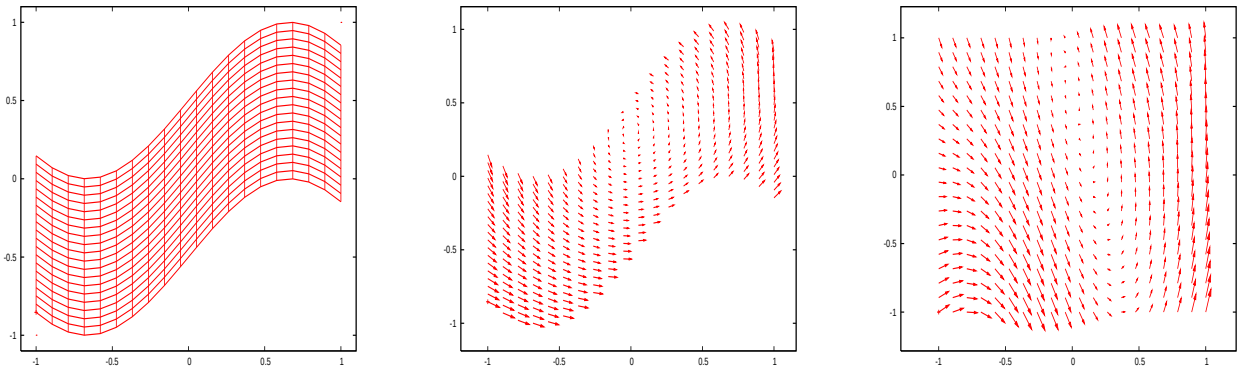
1. formulate Question 2 as a question about the solutions of a differential equation.
2. Present a discrete analogue to the smooth calibration problem.
3. Explore numerically the properties of Question 2 in the discretized model.
4. Propose an algorithm to solve the discrete problem and show its results on synthetic data.

Having defined our objectives, we may now present the differential formulation of Question 2.

3. Differential formulation

To re-state the problem in terms of differential equations, we express $\mathcal{D}_X F$ as a function of F, of the V_i and of ω_i. This is done by exploiting properties of F and the relations between F and the V_i.

[1] It is necessary to use the oriented projective plane, since a sensor may span more than one hemisphere.



Sensor shape | Flow in Euclidean coordinates | Flow in sensor coordinates

Figure 1. Left: The disposition of pixels in the Euclidean plane. A pixel is located at each vertex in the grid. Middle: the flow as it occurs in the Euclidean image plane. Right: the flow as it is observed in pixel (sensor) coordinates.

First, by the homogeneity properties of F, $F(\lambda X) = \lambda F(X)$, one gets the (column) vectorial equation:

$$\mathcal{D}_X F(X) \cdot X = \lim_{h \to 0} \frac{F(X + hX) - F(X)}{h} = F(X). \tag{7}$$

Then we join this to the two (column) vectorial equations (4, 5), and obtain the matrix equation: $\mathcal{D}_X F[V_1, V_2, X] = [\omega_1 \times F, \omega_2 \times F, F]$. Right-multiplying each side by the inverse of $[V_1, V_2, X]$ (wherever it may exist) yields the nonlinear equation

$$\mathcal{D}_X F = [\omega_1 \times F, \omega_2 \times F, F] \cdot [V_1, V_2, X]^{-1}, \tag{8}$$

that relates $\mathcal{D}_X F$ to F, V_i and ω_i, as desired. This equation holds in all points in O where V_1 and V_2 are linearly independent, that is, in all points X in O s.t. $F(X)$ does not lie on the plane formed by the optical center $(0, 0, 0)$ and the axes ω_1, ω_2. Question 2 can now be formulated as:

Question 2′: Given V_1 and V_2, does there exist values of ω_1, ω_2 and a sensor F that solves Eq. (8) and under what conditions is this sensor uniquely determined (up to an orthogonal transformation) by the ω_i?

This paper addresses neither this last question nor Question 2 on theoretical grounds. Neither does it attempt to integrate Eq. (8) locally numerically. Instead, we turn to practical ways of determining globally ω_1, ω_2 and F that solve equations (4, 5).

4. Discrete formulation

In this section, we show how to estimate the sensor geometry from two flows generated by purely rotational motions of the camera. The method described here relies mostly on linear algebra. We also give a glimpse of a method that attempts to solve the calibration in the least-square sense, in image coordinates.

We assume that vector motion values $v^i_{m,n}$, $i \in \{1, 2\}$, $m \in \{1, \ldots, M\}$, $n \in \{1, \ldots, N\}$ have been observed on a grid of points $x_{m,n}$ of the sensor image plane and we have mapped these values to 3D vector motions $V^i_{m,n}$, using Eq. (2) and (3). In this section, we will represent F by a $3 \times MN$ matrix, written "F" too, where each column represents the value of F at a 3D point $X_{m,n}$ on the unit sphere:

$$F = [F(X_{1,1}), F(X_{2,1}), \ldots F(X_{M,N})],$$

so that each column of F has unit norm. In addition, the solutions F that we seek has full rank (i.e. rank three), since the sensor is a diffeomorphism.

4.1. "Linear" solution

In the finite element model of F, we approximate $\mathcal{D}_X F$ at a grid point $X_{m,n}$ (not on the border) by solving the relations $\mathcal{D}_X F(X_{mn})(X_{m+1,n} - X_{m-1,n}) = F(X_{m+1,n}) - F(X_{m-1,n}) + O\left(\|X_{m+1,n} - X_{m-1,n}\|^2\right)$, $\mathcal{D}_X F(X_{mn})(X_{m,n+1} - X_{m,n-1}) = F(X_{m,n+1}) - F(X_{m,n-1}) + O(\|X_{m,n+1} - X_{m,n-1}\|)$ and $\mathcal{D}_X F(X_{mn}).X_{mn} = F(X_{mn})$, which yields the finite difference scheme

$$\begin{aligned} &\mathcal{D}_X F(X_{mn}) \simeq \\ &[F_{m+1,n} - F_{m-1,n},\ F_{m,n+1} - F_{m,n-1},\ F_{mn}] \\ &\quad [X_{m+1,n} - X_{m-1,n},\ X_{m,n+1} - X_{m,n-1},\ X_{mn}]^{-1}. \end{aligned} \tag{9}$$

At border point, the symmetric difference is replaced by forward or backward differences, which are less accurate. To isolate F in Eq. (9), we write

$$\mathcal{D}_X F(X_{m,n}) \simeq F K_{m,n},$$

where $F = [F_{1,1}, F_{2,1}, \ldots F_{M,N}]$ and $K_{m,n}$ holds the coefficients of Eq. (9). The relations (4, 5) between the observed flow and the 3D motion thus become, omitting the

superscript index i, $FK_{m,n}V_{m,n} = S_\omega F_{m,n}$. Grouping all the $MN \times 1$ vectors $K_{m,n}V_{m,n}$ into a single matrix $K = [K_{1,1}V_{1,1}, K_{2,1}V_{2,1}, \ldots, K_{M,N}V_{M,N}]$, one obtains the relation

$$FK = S_\omega F, \tag{10}$$

where S_ω, sometimes written $[\omega]_\times$, is the Rodrigues matrix of ω. Note that it would be more correct to write $FK \simeq S_\omega F$, since the finite differencing embedded in K induces some approximation error, even when the values $V_{m,n}$ verify Eq. (4) (or 5) exactly. Note also that, if F is linear, then Eq. (10) holds exactly. Since the only linear mappings that preserve the unit sphere are orthogonal transformations, Eq. (10) holds exactly holds only for sensors that are orthogonal transformations.

Since F can only be estimated up to an orthogonal transformation, we can, without loss of generality, assume that the axis of the first rotation is of the form $\omega_1 = [0,\, 0,\, \theta_1]$ and that of the second rotation is of the form $\omega_2 = [0,\, \gamma_1,\, \gamma_2]$, for some unknown θ_1, γ_1 and γ_2. The constraints from which we wish to estimate F, θ_1, γ_1 and γ_2 are thus:

$$FK_1 = \underbrace{\begin{bmatrix} 0 & -\theta_1 & 0 \\ \theta_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{[\omega_1]_\times = S_1} F \text{ and} \tag{11}$$

$$FK_2 = \underbrace{\begin{bmatrix} 0 & -\gamma_1 & \gamma_2 \\ \gamma_1 & 0 & 0 \\ -\gamma_2 & 0 & 0 \end{bmatrix}}_{[\omega_2]_\times = S_2} F. \tag{12}$$

Our problem is thus, given the K_i, computed from the V_i, to determine the θ_1, γ_1 and γ_2 for which Eq. (11 and 12) admit a solution F, or, equivalently, the θ_1, γ_1 and γ_2 for which the nullspaces of the $3MN \times 3MN$ matrices

$$\begin{aligned} L_1 &= \left(K_1^\top \otimes I_3\right) - \left(I_{MN} \otimes S_1\right) \text{ and} \\ L_2 &= \left(K_2^\top \otimes I_3\right) - \left(I_{MN} \otimes S_2\right) \end{aligned}$$

have a non-empty intersection - preferably of dimension one. Our problem is thus different from solving a system "$AX + XB = 0$", since we do not know "A" -in our case, the Rodrigues´ matrices- entirely.

In order to solve this problem, we tailored an algorithm in two steps, plus a third refinement step. These steps are now described:

4.1.1 Determining the row span of F from the K_i

First, we may determine the third row of F: from the third row of Eq. (11), one sees that the third row of $F = [f_1,\, f_2,\, f_3]^\top$ belongs to the nullspace of $K_1^\top$. Since, in practice, $K_1^\top$ has corank one, the third row of F, $f_3^\top$, is given, up to a scale factor, by the "last" left singular vector of K_1. Then, the relation $f_3^\top K_2 = -\gamma_2 f_1^\top$ gives us f_1, up to a scale factor. Finally, $f_1^\top K_1 = -\theta_1 f_2^\top$ yields f_2, up to a scale factor. The row span of F (i.e. the span of $F^\top$) being that of these scaled versions of its rows, we are in measure to compute span F from the K_i and represent it by a $3 \times MN$ orthogonal matrix U.

Note that another estimate can be computed by exchanging the roles of K_1 and K_2 and that these estimates do not necessarily agree. The two estimates can be merged by ad-hoc means.

4.1.2 Determining θ_1, γ_1 and γ_2 from the row span of F.

If U is an orthogonal matrix that forms a basis of the row span of F, then F can be written $F = AU$, for some invertible matrix A. Eqs. (11 and 12) can then be written $AUK_i = S_i AU$, which implies $UK_iU^\top = A^{-1}S_iA$, so that $UK_iU^\top$ is similar to S_i and its eigenvalues are thus 0 and $\pm i\,\|\omega_i\|$.

Computing the eigenvalues of $UK_iU^\top$ thus yields estimates of $|\theta_1|$ and $\sqrt{\gamma_1^2 + \gamma_2^2}$. In practice, if U is inexact, the matrix $UK_iU^\top$ may have three real singular values, in which case our method cannot estimate the corresponding angle. This is a similar problem to that of [5] and [17], with the difference that, due to the nature of the matrix, this occurs much more easily and, in practice, our computations are extremely sensitive to noise.

Once we have $|\theta_1|$ and $\sqrt{\gamma_1^2 + \gamma_2^2}$, and due to the inherent ambiguity in the problem we address, we may restrict ourselves to the case $\theta_1 > 0$, $\gamma_1 = \|\omega_2\| \cos(\tau)$ and $\gamma_2 = \|\omega_2\| \sin(\tau)$, for some $\tau \in [-\pi/2, \pi/2[$. Also note that, if one knows θ_1 and the γ_i, then the relation

$$\underbrace{\begin{bmatrix} \left(UK_1^\top U^\top \otimes I_3\right) - \left(I_3 \otimes S_1\right) \\ \left(UK_2^\top U^\top \otimes I_3\right) - \left(I_3 \otimes S_2\right) \end{bmatrix}}_{B(\tau)} \text{vec}\,(A) = 0_{9\times 1}, \tag{13}$$

holds, where $\text{vec}\,(A)$ is the vector of the 9 elements of A [10]. Moreover, in practice, $B(\tau)$ is only rank-deficient for the correct value of τ, and then by one. Thus, τ can be estimated by minimizing the least singular value of $B(\tau)$, on the interval $[-\pi/2, \pi/2[$, a simple 1D minimization task. Figure 2 shows typical curves of the two smallest singular values of $B(\tau)$, for $\tau \in [-\pi, \pi]$. In addition to clearly displaying the symmetry of the curves, this figure also confirms that the two smallest singular values are distinct, so that there is a unique A (up to scale) that solves Eq. (13). Obviously, the aspect of the curve changes when different motions and sensors are used, but these properties persist.

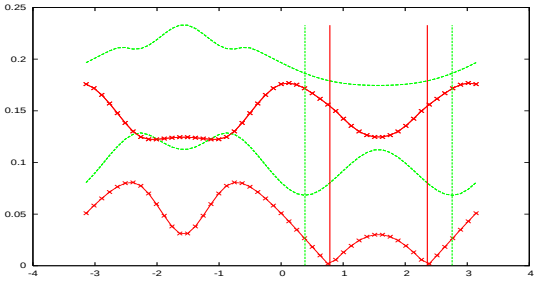


Figure 2. The two smallest singular values of $B(\tau)$ as a function of τ, computed over $[-\pi, \pi]$. The singular values clearly take the same value for τ and $\pi - \tau$. Red dotted curves: two smallest singular values of $B(\tau)$ computed for the true values of $\|\omega_i\|$ and U. Green smooth curves: singular values computed with the values of $\|\omega_i\|$ and U estimated in Sec. 4.1.2. The vertical lines indicate the positions of the minima of the smallest singular value.

Note that this also yields A. However, we observed that the estimate $F = AU$ is very distorted, so that the final estimate of F is given by the following method:

4.1.3 Refining F

We now show how to refine the estimate of F. Fixing, in Eqs. (11, 12), F to the value obtained by the above method, yields equations that are linear in θ_1, γ_1 and γ_2. We obtain new estimates of these values by solving these equations in the least-squares sense. We then re-compute the matrices L_1 and L_2 with these new values of θ_1, γ_1 and γ_2. Finally, we re-compute F, up to a scale factor, as the nullspace of the matrix $\begin{bmatrix} L_1^\top & L_2^\top \end{bmatrix}^\top$, estimated by the SVD of this matrix. We found that, due to the conditioning of the L_i, this is preferable to estimating the nullspace of the smaller matrices $L_1^\top L_1 + L_2^\top L_2$.

Note that the methods of this and the previous section do not guarantee that the columns of F will all have the same norm. In this paper, we normalize each column as a final computation step.

Another limitation is that the sensor F may "cross itself," i.e. not correspond to an injective mapping, as will be illustrated in the next section.

4.2. Numerical experiments

In this section, we consider four types of 20×20 sensors. First, the *identity sensor* $F(X) = X$, displayed in Fig. 3, top, left; we will also use its variant, the *orthogonal sensor*, $F(X) = RX$, for some random orthogonal matrix R. Second, in Fig. 3, top, right, the *homography sensor* defined by $F(X) \sim AX$, where A is a random 3×3 matrix. Third, the *sine sensor* shown in Fig. 3, bottom, left, with pixels disposed at positions

$$F_s(x, y, 1) \sim \left(x\,,\ \frac{1}{2} y + \frac{1}{2} \sin\left(\frac{3\pi x}{4} \right)\,,\ 1 \right),$$

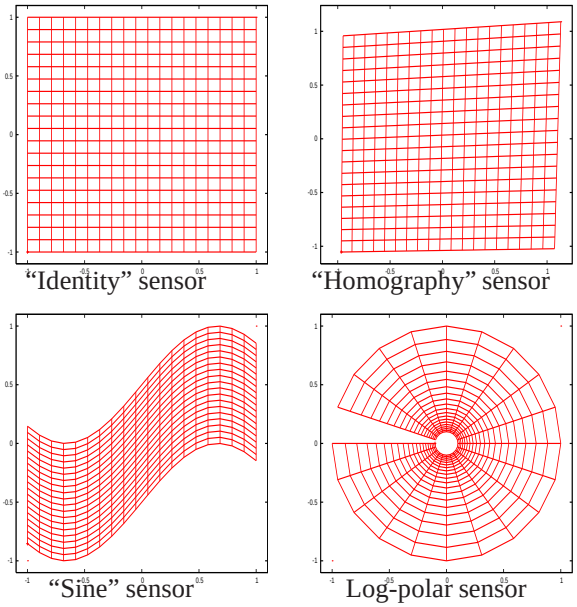


Figure 3. The two types of sensors that we consider.

where x and y take regularly-spaced values in the interval $[-1, 1]$ (again, $\sim$ means equality up to a positive scale factor). Finally, the *log-polar sensor* shown in Fig. 3, bottom, right, which is defined by the mapping

$$F_l(\rho, \phi, 1) \sim \left(10^{\frac{(\rho-1)}{2}} \cos(\pi\phi)\,,\ 10^{\frac{(\rho-1)}{2}} \sin(\pi\phi)\,,\ 1 \right).$$

where the values of ρ and ϕ are regularly spaced in the interval $[-1, 1]$.

All the pairs of flows used here correspond to randomly chosen rotation axes ω_i forming an angle of $2\pi/3$ (other values give similar results) with norm drawn randomly and uniformly in $[0.15,\ 0.30]$.

4.2.1 Unicity of the solution to Eq. (11, 12)

In order to determine whether the least-squares solution F, θ_1, γ_1, γ_2 to Eqs. (11, 12) is unique, we study the two smallest singular values of $\begin{bmatrix} L_1^\top, L_2^\top \end{bmatrix}^\top$, as functions of $(\theta_1, \gamma_1, \gamma_2)$, in the neighborhood of the true values $(\theta_1^*, \gamma_1^*, \gamma_2^*)$. The solution will be unique if 1) the smallest singular value σ_1 is distinct from the second smallest σ_2 and 2) the minimum of the smallest singular value σ_1 is isolated. We inspect these three-dimensional functions $\sigma_1(\theta_1, \gamma_1, \gamma_2)$ and $\sigma_2(\theta_1, \gamma_1, \gamma_2)$ by looking at three of their one-dimensional slices: $s_{i,1}(\varepsilon) = \sigma_i((1+\varepsilon)\theta_1, \gamma_1, \gamma_2)$, $s_{i,2}(\varepsilon) = \sigma_i(\theta_1, (1+\varepsilon)\gamma_1, (1+\varepsilon)\gamma_2)$ and $s_{i,3}(\varepsilon) = \sigma_i(\theta_1, \gamma_1 - \varepsilon\gamma_2, \gamma_2 + \varepsilon\gamma_1)$. In geometric terms, $s_{i,1}(\varepsilon)$ shows the effect of a perturbation of the angular velocity ω_1 parallel to itself, $s_{2,1}$ shows the effect of a perturbation of ω_2 parallel to itself, and $s_{i,3}(\varepsilon)$ shows the effect of a

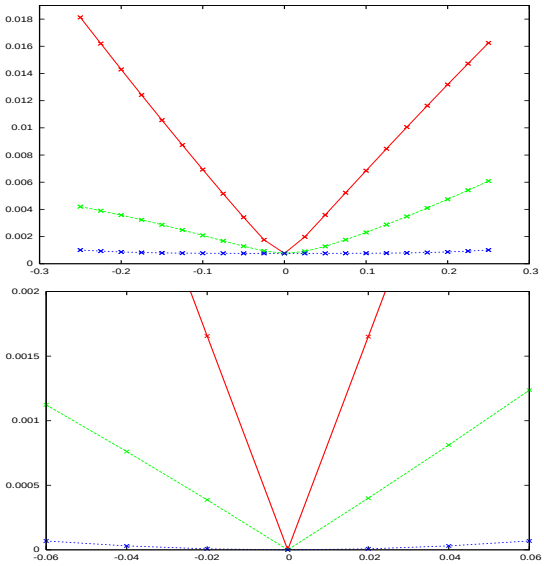


Figure 4. Smallest singular value of $\left[L_1^\top,\, L_2^\top\right]$, when θ_1, γ_1 and γ_2 are perturbed in the three directions described in the text. Top: with a sine sensor. The steepest curve is $s_{1,j}(\varepsilon)$, the second steepest is $s_{2,j}(\varepsilon)$, while $s_{3,j}(\varepsilon)$ has a wide, slightly concave, floor around the true value. Bottom: with an orthogonal sensor. $s_{3,j}(\varepsilon)$ reaches a (shallow) minimum in 0.

perturbation of ω_2 orthogonal to itself, while staying in the (ω_1, ω_2) plane.

We do not need to plot the $s_{2,j}(\varepsilon)$ here, as their values are much greater than that of the $s_{1,j}(\varepsilon)$. This indicates that, for fixed ω_i, the smallest singular value is isolated, so there is a unique (up to scale) F that solves Eqs. (11, 12) in the least-squares sense.

Figure 4, top, shows the curves of $s_{1,j}(\varepsilon)$, $1 \leq j \leq 3$, for $\varepsilon \in [-1/4, 1/4]$, computed for a "sine" sensor, while the bottom curves correspond to an orthogonal sensor, for $\varepsilon \in [-0.06, 0.06]$. The results of this experiment do no vary greatly with different sensors and motions. One important fact showed by these curves is that the minima of $s_{11}(\varepsilon)$ and $s_{12}(\varepsilon)$ are sharply defined, indicating that the amplitude of the angular velocities $\|\omega_i\|$ are well defined by the matrices L_1 and L_2, at least near the true values. This conclusion can also be reached analytically.

The most important fact in this figure is that $s_{1,3}(\varepsilon)$ has a wide floor around 0. For the sine sensor, this function is actually concave in 0, while it is convex for the orthogonal sensor. In both cases, the extreme flatness indicates that the angle between ω_1 and ω_2 is, at best, poorly determined by Eqs. (11, 12).

The extent to which this indeterminacy is caused, on the one hand, by numerical approximations in the matrices K_i and their ill-posedness and, on the other hand, by a genuine ambiguity is unclear.

To summarize the results experiment, it shows that the angular velocities $\|\omega_i\|$ are well defined by the matrices L_1 and L_2, while the angle between the axes is not, due to the flat region around the minimum of $s_{1,3}(\varepsilon)$. Since this flat region directly challenges our ability to estimate the γ_i accurately, determining the factors that influence the extent of this floor appears as important subject for future investigation.

Having presented an important limitation to the estimation of the angular velocities, we now show how the algorithm presented in the previous section behaves, in spite of this limitation.

4.2.2 Calibration experiments

We now present results of the steps described in Secs. 4.1.1-4.1.3. The first row of Figure 5 shows the flow as it appears in the Euclidean image plane. The second row shows the 2D projection of the result of applying the steps in Secs. 4.1.1 and 4.1.2, superposed with the true sensor. Since these results are obtained up to an orthogonal transformation, we align the true and estimated sensor by 3D procrustes [1]. The third row shows the result of the method of Sec. 4.1.3.

The first column of Figure 5 shows that the methods of Secs. 4.1.2 and 4.1.3 both yield the exact sensor in the case of the identity sensor, for which differentiation by finite differences is exact. The same can be observed with any orthogonal sensor. The success of our method in these cases strongly points towards the unicity of the solution of Eqs. (11) and (12).

The second column shows that the reconstruction becomes inexact when the sensor is not an orthogonal transformation, but just "collinear" to a general homography. The quality of the reconstruction displayed here is typical of what is obtained when reconstructing a "homography" sensor.

The third column shows that the reconstruction becomes less precise in the case of the "sine" sensor - the reconstruction displayed here is typical of what is obtained when reconstructing a "sine" sensor. It is not uncommon that the estimation of the sine sensor fails.

The last column shows that the reconstruction becomes less precise still in the case of the "log-polar" sensor - the reconstruction displayed here is a particularly good-looking one. It is common that the estimation of the log-polar sensor fails.

Comparing the second and third rows plainly shows that the method of Sec. 4.1.3 yields better results than that of Sec. 4.1.2.

In particular, the estimated sine and log-polar sensors shown in the second row do not correspond to diffeomorphisms, as they have overlapping sections.

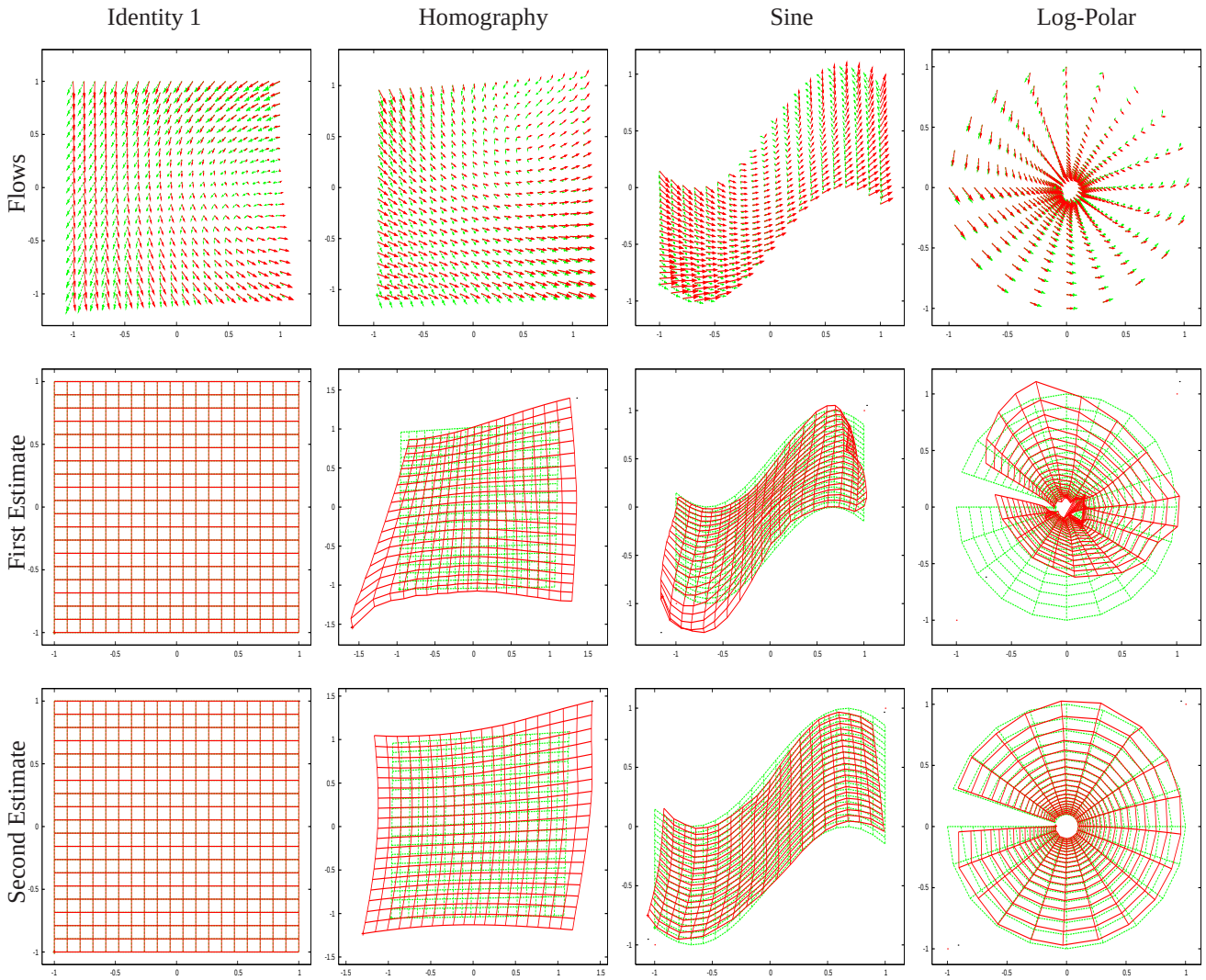


Figure 5. Top row: the flows from which the sensor geometry will be computed, displayed in the Euclidean image plane. The second and third rows show, respectively, the result obtained by the methods in Sec. 4.1.2 and in Sec. 4.1.3. The true sensor shape is superposed, for comparison. Each column shows the data for one of the four tested sensor types.

4.2.3 (Hyper) Sensitivity to noise in the observations

When one adds even tiny amounts of noise to the observations, our method ceases to produce meaningful results. It will occasionally produce recognizable results with 50DB of noise, i.e. when the error in the observations has a standard deviation of approximately 0.3% of the flow amplitude, a level much smaller than real-world levels. Figure 6 shows the output of the method of Sec. 4.1.2 (left), of that of Sec. 4.1.3 (middle) and of a method which minimizes the sum of squared residues in image space over all possible sensor geometries and angular velocity vectors ω_i (left), and gives a clearly superior results.

We do not describe this last method here, because it is based on totally different principles and for lack of space.

5. Conclusions

Without answering Questions 2 and 2' by mathematical proof, we have presented numerical results that provide some insight on the self-calibration problem. These results could be coherent with either a unique solution (up to an orthogonal transformation), or a continuum of solutions. Although answering these questions in a definitive manner still appears as a major challenge, the fact that orthogonal sensors are perfectly reconstructed by our method draws a case for the unicity of the solution, at least for that type of sensor and for the conditions of our experiments.

We have presented what is, to our knowledge, the first algorithm to densely calibrate a non-parametric sensor from two motion flows generated by purely rotational motions.

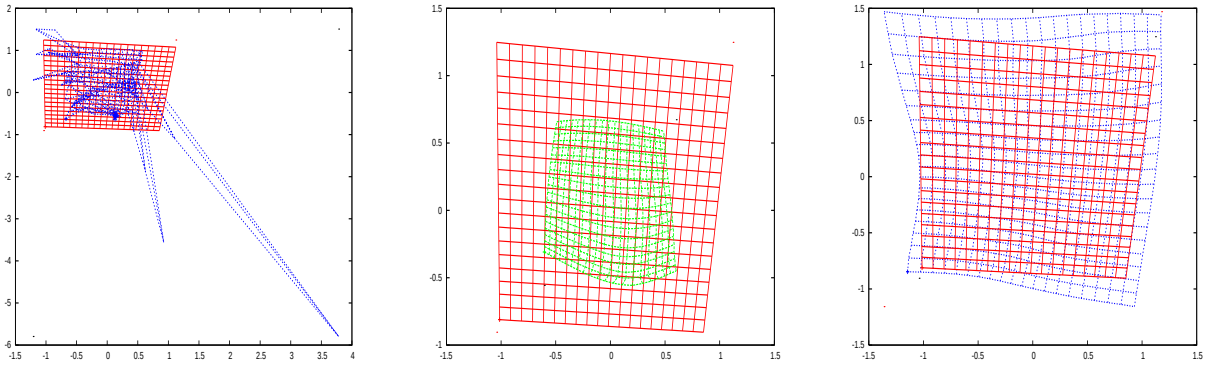


Figure 6. A successful reconstruction in the presence of noise at a level 50DB. From left to right, the results of the method of Sec. 4.1.2, Sec. 4.1.3 and of a method that minimizes the squared residues in image space.

We have presented its results together with its limitations, which may be inherent to the problem at hand.

In the process, we found some numerical evidence that the ambiguity in self-calibrating from two motion flows induced by pure rotations could be greater than just an orthogonal transformation. However, it remains unclear whether this ambiguity is due to numerical approximations in the matrices K_i, or whether it is a genuine ambiguity. Because, in the absence of approximations in the matrices K_i, i.e. with the identity or other orthogonal sensors, the reconstruction is perfect, we believe the difficulty met by our algorithm is due to the approximation in the finite difference method and to ill-conditioning of the problem.

It would thus be interesting know whether preconditioning the matrices K_i could improve the stability of the algorithm. Certainly, the computation cost could be reduced by using fast methods to compute the least singular values and vectors of large sparse matrices [4]. Possible extensions include using discrete elements, rather than a finite element mesh. Also, adapting matrices K_i to other sensor topologies seems feasible. This could e.g. accommodate the cylindrical topology of the logpolar sensor. An important improvement would obviously be the possibility of using more than two images.

References

[1] M. D. Akca. Generalized procrustes analysis and its applications in photogrammetry. Technical report, ETH, Swiss Federal Institute of Technology Zurich, Institute of Geodesy and Photogrammetry, 2003. 6

[2] S. Baker and S. K. Nayar. A theory of catadioptric image formation. In *proc. ICCV*, pages 35–42, 1997. 1

[3] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical applications. In *proc. ECCV*, volume II, pages 445–461. Springer-Verlag, 2000. 1

[4] H. Guo. An algorithm for computing the smallest singular value of large-scale matrices. *International J. of Computer Math*, pages 89–104, 2001. 8

[5] R. Hartley. Self-calibration from multiple views with a rotating camera. In *proc. ECCV*, pages 471–478, 1994. 1, 4

[6] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000. 1

[7] J. Heikkilä and Olli Silvén. A four–step camera calibration procedure with implicit image correction. In *proc. CVPR*, Puerto Rico, June 1997. IEEE Computer Society Press. 1

[8] B.K. Horn. *Robot vision*. MIT Press, 1986. 1

[9] R.K. Lenz and R.Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3-D machine vision metrology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10:713–720, 1988. 1

[10] J. R. Magnus and H. Neudecker. *Matrix differential calculus with applications in statistics and econometrics*. John Wiley & Sons, 1999. 4

[11] D. Nistér and F. Schaffalitzky. What do four points in two calibrated images tell us about the epipoles? In *proc ECCV*, 2004. 1

[12] D. Nistér, H. Stewenius, and E. Grossmann. Non-parametric self-calibration. In *proc. ICCV*, 2005. 1, 2

[13] R. Pless. Discrete and differential two-view constraints for general imaging systems. In *proc. Workshop on Omnidirectional Vision (OMNIVIS)*, 2002. 1

[14] S. Ramalingam, P. Sturm, and S. Lodha. Towards generic self-calibration of central cameras. In *proc. ICCV workshop on Omnidirectional Vision, Camera Networks and Non-classical cameras (OMNIVIS)*, pages 20–27, 2005. 1, 2

[15] J. Stolfi. *Oriented projective geometry*. Academic Press, 1991. 2

[16] P. Sturm and S. Ramalingam. A generic concept for camera calibration. In *proc ECCV*, 2004. 1

[17] C. Tomasi and T. Kanade. Shape and motion from image streams : a factorization method. Technical Report TR 92-1270 and CMU-CS-91-104, Cornell University and Carnegie Mellon University, 1992. 4