Radial Distortion Homography

Zuzana Kukelova[1]   Jan Heller[2]   Martin Bujnak[3]   Tomas Pajdla[2]

[1]Microsoft Research Ltd,
21 Station Road,
Cambridge CB1 2FB, UK
a-zukuke@microsoft.com

[2]Czech Technical University
in Prague, 166 27 Praha 6,
Technická 2, Czech Republic
{hellej1,pajdla}@cmp.felk.cvut.cz

[3]Capturing Reality s.r.o.,
Syslia 46, 821 05,
Bratislava, Slovakia
martin@capturingreality.com

Abstract

The importance of precise homography estimation is often underestimated even though it plays a crucial role in various vision applications such as plane or planarity detection, scene degeneracy tests, camera motion classification, image stitching, and many more. Ignoring the radial distortion component in homography estimation—even for classical perspective cameras—may lead to significant errors or totally wrong estimates. In this paper, we fill the gap among the homography estimation methods by presenting two algorithms for estimating homography between two cameras with different radial distortions. Both algorithms can handle planar scenes as well as scenes where the relative motion between the cameras is a pure rotation. The first algorithm uses the minimal number of five image point correspondences and solves a nonlinear system of polynomial equations using Gröbner basis method. The second algorithm uses a non-minimal number of six image point correspondences and leads to a simple system of two quadratic equations in two unknowns and one system of six linear equations. The proposed algorithms are fast, stable, and can be efficiently used inside a RANSAC loop.

1. Introduction

The estimation of a homography between two views is a crucial problem in computer vision with many application, *e.g.*, in image stitching, structure from motion or camera calibration. A homography exists between projections of points on a 3D plane in two views or between projections of general 3D points in two views when the transformation between the two views is a pure rotation.

A number of algorithms have been proposed for homography estimation in the past. The classical linear algorithm [8] estimates the general homography from 4 point correspondences. This algorithm is frequently used in practice, but it assumes perspective projection and thus it

doesn't provide accurate estimates for image correspondences corrupted by radial distortion.

This is a serious drawback, since virtually all real projections involve some amount of radial distortion. It was shown that ignoring the radial distortion, even for standard consumer cameras, may lead to significant errors in 3D reconstruction [7], metric measurements from images, or in camera calibration.

One way to deal with radial distortion is to model it in the final optimization step, *i.e.*, in the bundle adjustment [13]. However, this approach requires a set of correct image matches which may be difficult to find and thus in many situations—and especially for larger radial distortions—it is necessary to consider the distortion already when searching for image correspondences in a RANSAC loop [6, 4]. By not providing correct initial radial distortion estimates, the bundle adjustment is more likely to fall into a local minima without recovering correct distortion parameters. Radial distortion modelling proved to be a mathematically challenging task and algorithms for estimating homography or epipolar geometry for cameras with radial distortion have been proposed only recently [7, 1, 9, 2, 3].

In [7], Fitzgibbon proposed a one parameter division model for modeling radial distortion and, based on this model, algorithms for fundamental matrix and homography estimation. The homography estimation algorithm assumes one common radial distortion parameter and uses five image point correspondences. It leads to a quadratic eigenvalue problem yielding up to 18 real solutions.

Two algorithms for estimating homography between cameras undergoing pure rotation, *i.e.*, the panorama stitching problem, have also been proposed recently [9, 2]. Both algorithms use three image point correspondences to estimate homography as well as one radial distortion parameter and the focal length.

The first algorithm proposed in [9] formulates the panorama stitching problem as an optimization problem and employs Levenberg-Marquardt method to solve it. Be-

cause of the local optimization method used, this algorithm may have problems with convergence to the global minima. Also, it returns only one of 18 possible solutions depending on the initialization. The paper also reports poor convergence of the proposed method for larger radial distortions.

A different algorithm has been proposed in [2]. The authors formulated the panorama stitching problem for a camera with unknown radial distortion and unknown focal length as a system of two equations in two unknowns and solve it using the Gröbner basis method. The final Gröbner basis solver performs LU decomposition of a 90×132 coefficient matrix and eigenvalue computation of a 25×25 matrix. This leads to 25 solutions from which only 18 are correct. The final solver is quite slow and not suitable for real-time applications.

Since the previously mentioned algorithms estimate only one radial distortion parameter shared by both cameras, they cannot be used for two cameras with different radial distortions observing a plane.

In [3], an algorithm for estimating fundamental matrix for cameras with different radial distortions from nine image point correspondences was proposed. Unfortunately, this algorithm doesn't work for planar scenes and pure rotations. Moreover, the used Gröbner basis method leads to a quite complicated and slow solver (LU decomposition of a 330×390 matrix and eigenvalue computations of a 24×24 matrix).

An important application of homography estimation can be found in the Structure-from-Motion (SfM) problem. Homography estimation is typically used to validate a quality of a pairwise reconstruction during a seed searching phase [12]. The goal is to select a well-conditioned pair, *i.e.*, a not too planar scene with a rich 3D structure. Omitting the radial distortion estimation in the plane detection, *i.e.*, homography estimation, may lead to a wrong classification of the scene as non-planar, even if it is completely planar or a panorama. We observed a significant occurrence of this misclassification even for images with small amounts radial distortion. Thus it is important to use an algorithm that can estimate homography as well as different distortion parameters for each camera.

The main motivation of this paper is to fill the gap among the homography estimation methods by developing an algorithm that can handle both planar and pure rotation scenes as well as cameras with different radial distortions. The goal is to propose an algorithm that uses as low a number of point correspondences as possible while being sufficiently fast to be used inside a RANSAC loop [6, 4] in real-time applications or in the seed detection phase of large SfM pipelines [12].

We have developed two algorithms—solvers—for estimating unknown homography and radial distortion parameters of two cameras. The first solver, H5$\lambda_1\lambda_2$, uses the

	F9$\lambda_1\lambda_2$	H4	H3λf	H5λ	H6$\lambda_1\lambda_2$	H5$\lambda_1\lambda_2$
Reference	[3]	[8]	[2]	[7]		
Plane		✓		✓	✓	✓
Pure rotation		✓	✓	✓	✓	✓
Radial distortion	✓		✓	✓	✓	✓
Different distortions	✓				✓	✓
Minimal solution	✓	✓	(✓)	(✓)		✓
Number of points	9	4	3	5	6	5
Number of solutions	24	1	18 (25)	18	2	5

Table 1: Comparison of properties of various solvers.

minimal number of five image point correspondences and solves a nonlinear system of polynomial equations using the Gröbner basis method. This solver performs Gauss-Jordan (G-J) elimination of quite a small 16×21 matrix and provides five solutions from the eigenvalues of a 5×5 matrix. The second solver, H6$\lambda_1\lambda_2$, uses six image point correspondences and solves a simple system of two quadratic equations in two unknowns and one system of six linear equations. This leads to two plausible solutions.

Table 1 presents a comparison of all algorithms related to the two new proposed solvers, H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$, w.r.t. the number of point correspondences used, the number of solutions, and other properties. The table shows that the only algorithms that work for both planar as well as pure rotation scenes and that can handle different radial distortions at the same time are the two proposed solvers. Moreover, the new solvers return less solutions than their competitors (F9$\lambda_1\lambda_2$, H3λf, H5λ) while being faster and less complicated. These properties are crucial for solvers that are to be used in a RANSAC-style loop [6, 4]. The symbol "(✓)" in "Minimal solution" row for H3λf and H5λ algorithms means that the number of correspondences used provides more constraints than needed for the respective problem, however, these problems cannot be solved from a smaller number of correspondences.

Next, we formulate the problem of homography estimation between two cameras with different radial distortions.

2. Problem Formulation

Let us assume that a planar object (a plane) is observed from two different view-points by cameras with the projection matrices $\mathtt{P} = [\mathtt{I} \,|\, \mathbf{0}]$ and $\mathtt{P}' = [\mathbf{A} \,|\, \mathbf{a}]$. The corresponding image points $\mathbf{x}_i = [x_i, y_i, 1]^\top$ in the first and $\mathbf{x}'_i = [x'_i, y'_i, 1]^\top$ in the second view are related as

$$\alpha_i \mathbf{x}'_i = \mathtt{H} \mathbf{x}_i, \qquad (1)$$

where α_i is an unknown scalar value and $\mathtt{H} \in \mathbb{R}^{3\times3}$ is a homography between the views, also known as the homography induced by a plane [8]. It can be shown that for a

plane defined by $\pi^\top \mathbf{X} = 0$ with $\pi = [\mathbf{v}, 1]^\top$, the homography H (1) induced by the plane π has the form $\mathtt{H} = \mathtt{A} - \mathbf{a}\mathbf{v}^\top$, where $\mathtt{A} \in \mathbb{R}^{3\times 3}$ and $\mathbf{a} \in \mathbb{R}^3$.

A similar homography relation can be formulated for projections of general 3D points observed in two views when the transformation between the cameras is a pure rotation. In this case the homography has the form $\mathtt{H} = \mathtt{K}'\mathtt{R}\mathtt{K}^{-1}$, where $\mathtt{K}'$ and $\mathtt{K}$ are 3×3 calibration matrices of the respective cameras and R is the rotation connecting the camera coordinate frames. The homography $\mathtt{H} = \mathtt{K}'\mathtt{R}\mathtt{K}^{-1}$ can be also viewed as a homography $\mathtt{H}_\infty$ induced by the plane at infinity.

In this paper, we consider general homographies of the form (1), where $\mathtt{H} \in \mathbb{R}^{3\times 3}$ is a general regular 3×3 matrix, making no distinction between homographies induced by a plane and homographies induced by the plane at infinity. This makes both proposed solvers oblivious to the homography type and performing equally well for either.

Note that the relation (1) holds only for points projected by an ideal perspective camera. For every real camera, some amount of radial distortion is always present and for the relation (1) to hold true, one must first "undistort" the measured image points $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}'_i$. Here, we use the one-parameter division model for radial distortion modeling [7]. This undistortion model can handle even quite pronounced radial distortions and has the following form:

$$f_u(\hat{\mathbf{x}}_i, \lambda) = \left[\hat{x}_i, \hat{y}_i, 1 + \lambda(\hat{x}_i^2 + \hat{y}_i^2)\right]^\top, \qquad (2)$$

where $\hat{\mathbf{x}}_i = [\hat{x}_i, \hat{y}_i, 1]^\top$ are the homogeneous coordinates of the measured (and radially distorted) image points and $\lambda \in \mathbb{R}$ is the distortion parameter. Here, we will assume that the centre of distortion is in the center of the image.

In this work, we consider different radial distortions for each camera. This means that the proposed solver will be able to handle images taken by one as well as by two different cameras observing a plane or undergoing pure rotation. By combining the homography relation (1) with two different undistortion models (2) we get an equation relating image correspondences distorted with different amounts of radial distortion

$$\alpha_i f_u(\hat{\mathbf{x}}'_i, \lambda') = \mathtt{H} f_u(\hat{\mathbf{x}}_i, \lambda). \qquad (3)$$

Next, we will present two algorithms for estimating unknown homography and two unknown radial distortion parameters, *i.e.*, algorithms for solving (3), with H, λ, and λ' being the unknown parameters. The first solver H5$\lambda_1\lambda_2$ uses a minimal number of five image point correspondences and solves a nonlinear system of polynomial equations using Gröbner basis method. The second solver H6$\lambda_1\lambda_2$ uses six image point correspondences and results in a simple system of two quadratic equations in two unknowns and one system of six linear equations.

3. Minimal Solver H5$\lambda_1\lambda_2$

The first of the two solvers proposed in this paper starts by eliminating the scalar values α_i from the equation (3). This is done by multiplying (3) by the skew symmetric matrix $[f_u(\hat{\mathbf{x}}'_i, \lambda')]_\times$. Since $[f_u(\hat{\mathbf{u}}_i)]_\times f_u(\hat{\mathbf{u}}_i) = 0$, this leads to the following matrix equation

$$\begin{bmatrix} 0 & -\hat{w}'_i & \hat{y}'_i \\ \hat{w}'_i & 0 & -\hat{x}'_i \\ -\hat{y}'_i & \hat{x}'_i & 0 \end{bmatrix} \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} \hat{x}_i \\ \hat{y}_i \\ \hat{w}_i \end{bmatrix} = \mathbf{0},$$
$$(4)$$

where $\hat{w}_i = 1 + \lambda(\hat{x}_i^2 + \hat{y}_i^2)$, $\hat{w}'_i = 1 + \lambda'(\hat{x}_i'^2 + \hat{y}_i'^2)$, and h_{ij} is the element from the ith row and the jth column of the homography matrix H.

The matrix equation (4) contains three polynomial equations from which only two are linearly independent, because the skew symmetric matrix $[f_u(\hat{\mathbf{u}}_i)]_\times$ has rank two. This means that we need at least 5 image point correspondences to estimate the unknown homography H as well as the unknown radial distortion parameters λ and λ'.

The explicit form of the equation corresponding to the third row of the matrix equation (4) is

$$(-\hat{y}'_i\, h_{11} + \hat{x}'_i\, h_{21})\, \hat{x}_i + (-\hat{y}'_i\, h_{12} + \hat{x}'_i\, h_{22})\, \hat{y}_i$$
$$+ (-\hat{y}'_i\, h_{13} + \hat{x}'_i\, h_{23})\, (1 + \lambda(\hat{x}_i^2 + \hat{y}_i^2)) = 0. \quad (5)$$

This is a homogeneous equation in eight monomials $\mathbf{v}_1 = [h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, \lambda\, h_{13}, \lambda\, h_{23}]^\top$. Given the minimal number of five image point correspondences, we get five equations of the form (5) which we can stack into a matrix form as

$$\mathtt{M}_1\, \mathbf{v}_1 = \mathbf{0}, \qquad (6)$$

where $\mathtt{M}_1$ is a 5×8 coefficient matrix. Assuming that $\mathtt{M}_1$ has full rank equal to 5, *i.e.*, we have 5 non-degenerate image correspondences, the dimension of $\mathrm{null}(\mathtt{M}_1)$ is 3. This means that the monomial vector $\mathbf{v}_1$ can in general be rewritten as a linear combination of three null space basis vectors $\mathbf{n}_i$ of the matrix $\mathtt{M}_1$ as

$$\mathbf{v}_1 = \sum_{i=1}^{3} \gamma_i\, \mathbf{n}_i, \qquad (7)$$

where γ_i are new unknowns. Without the loss of generality, we can set $\gamma_3 = 1$ to fix the scale of the homography and to bring down the number of unknowns. This way, we obtain a parametrization of the first two rows of the unknown homography H in two unknowns γ_1 and γ_2.

Next, let us notice that the elements of the monomial vector $\mathbf{v}_1$ are not independent. We can see that $v_7 = \lambda\, v_3$ and $v_8 = \lambda\, v_6$, where v_i is the ith element of the vector $\mathbf{v}$. These two constraints, together with the parametrization

from equation (7), give us two equations in γ_1, γ_2, and λ

$$\gamma_1 n_{17} + \gamma_2 n_{27} + n_{37}$$
$$-\lambda(\gamma_1 n_{13} + \gamma_2 n_{23} + n_{33}) = 0, \qquad (8)$$
$$\gamma_1 n_{18} + \gamma_2 n_{28} + n_{38}$$
$$-\lambda(\gamma_1 n_{16} + \gamma_2 n_{26} + n_{36}) = 0, \qquad (9)$$

where n_{ij} is the jth element of the vector $\mathbf{n}_i$.

Now, we will use five linearly independent equations from the first and the second row of the matrix equation (4). In order to increase the numerical stability of the solver, we choose the equations corresponding to the first row of (4) if $|\hat{x}_i'| < \epsilon$ and the equations corresponding to the second row if $|\hat{y}_i'| < \epsilon$, for some small predetermined threshold ϵ. Otherwise, which is the most common case, we can choose arbitrarily from these two rows, *e.g.*, the equations corresponding to the second row, which has the form

$$(\hat{w}_i' h_{11} - \hat{x}_i' h_{31})\hat{x}_i + (\hat{w}_i' h_{12} - \hat{x}_i' h_{32})\hat{y}_i$$
$$+ (\hat{w}_i' h_{13} - \hat{x}_i' h_{33})\hat{w}_i = 0. \quad (10)$$

Note that $\hat{w}_i'$ and $\hat{w}_i$ are functions of λ and λ' and that h_{11}, h_{12}, h_{13}, and λh_{13} are already reparametrized by the two new unknowns γ_1 and γ_2.

The five linearly independent equations (10), $i = 1, \ldots, 5$, together with the two equations (8) and (9) form a system of seven equations in seven unknowns $(h_{31}, h_{32}, h_{33}, \lambda, \lambda', \gamma_1, \gamma_2)$ and 13 monomials $\mathbf{v}_2 = [h_{31}, h_{32}, h_{33}, \lambda h_{33}, \lambda' \gamma_1, \lambda' \gamma_2, \lambda', \lambda \gamma_1, \lambda \gamma_2, \lambda, \gamma_1, \gamma_2, 1]^\top$. Further, by reordering the elements of the monomial vector $\mathbf{v}_2$ such that $\mathbf{v}_2 = [h_{31}, h_{32}, h_{33}, \lambda h_{33}, \lambda' \gamma_1, \lambda \gamma_1, \gamma_1, \lambda' \gamma_2, \lambda \gamma_2, \lambda', \lambda, \gamma_2, 1]^\top$ and by performing G-J elimination of the corresponding coefficient matrix $\mathtt{M}_2$, we obtain the following matrix

$$
\mathtt{M}_2 =
\begin{array}{c}
{\scriptstyle h_{31}\ h_{32}\ h_{33}\ \lambda h_{33}\ \lambda'\gamma_1\ \lambda\gamma_1\ \gamma_1\ \lambda'\gamma_2\ \lambda\gamma_2\ \lambda'\ \lambda\ \gamma_2\ 1} \\
\begin{bmatrix}
1 & & & & & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
 & 1 & & & & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
 & & 1 & & & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
 & & & 1 & & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
 & & & & 1 & & & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
 & & & & & 1 & & & \bullet & & \bullet & \bullet & \bullet \\
 & & & & & & 1 & & \bullet & & \bullet & \bullet & \bullet \\
\end{bmatrix}
\end{array}. \quad (11)
$$

Note that the left 7×6 submatrix of the eliminated matrix $\mathtt{M}_2$ contains some zero elements. Also, the elements of this submatrix are not completely independent. Let $m_{i,j}$ denote the element of the ith row and the jth column of $\mathtt{M}_2$ (11). Then it holds that

$$\frac{m_{i,9}}{m_{i,11}} = \frac{m_{k,9}}{m_{k,11}}, \qquad (12)$$

for all $i, k \in \{1, 2, 3, 4, 5, 7\}$, $i \neq k$. These dependencies will be important later for generating the correct Gröbner basis solver.

Meanwhile, the eliminated system $\mathtt{M}_2 \mathbf{v}_2 = \mathbf{0}$ gives us seven equations

$$
\begin{aligned}
h_{31} + f_1\left(\lambda', \lambda, \gamma_2\right) &= 0, & (13) \\
h_{32} + f_2\left(\lambda', \lambda, \gamma_2\right) &= 0, & (14) \\
h_{33} + f_3\left(\lambda', \lambda, \gamma_2\right) &= 0, & (15) \\
\lambda h_{33} + f_4\left(\lambda', \lambda, \gamma_2\right) &= 0, & (16) \\
\lambda' \gamma_1 + f_5\left(\lambda', \lambda, \gamma_2\right) &= 0, & (17) \\
\lambda \gamma_1 + f_6\left(\lambda', \lambda, \gamma_2\right) &= 0, & (18) \\
\gamma_1 + f_7\left(\lambda', \lambda, \gamma_2\right) &= 0, & (19)
\end{aligned}
$$

where $f_1\left(\lambda', \lambda, \gamma_2\right), \ldots, f_7\left(\lambda', \lambda, \gamma_2\right)$ are polynomials in three variables λ', λ and γ_2. By exploiting dependencies between these seven equations we get a new system of three equations in three unknowns of degree three

$$
\begin{aligned}
\lambda f_3\left(\lambda', \lambda, \gamma_2\right) - f_4\left(\lambda', \lambda, \gamma_2\right) &= 0, \\
\lambda' f_7\left(\lambda', \lambda, \gamma_2\right) - f_5\left(\lambda', \lambda, \gamma_2\right) &= 0, \quad (20) \\
\lambda f_7\left(\lambda', \lambda, \gamma_2\right) - f_6\left(\lambda', \lambda, \gamma_2\right) &= 0.
\end{aligned}
$$

To solve the system of polynomial equations (20), we use the Gröbner basis method [5]. The Gröbner basis method has been recently used to create very fast, efficient, and numerically stable solvers for many difficult problems in computer vision. The method is based on polynomial ideal theory and special bases of ideals called Gröbner bases [5]. Gröbner bases can be used to create special multiplication matrices [5] whose eigenvalues and eigenvectors give solutions to the initial systems of polynomial equations.

In order to implement an efficient Gröbner basis solver for the system (20), we used the automatic generator of Gröbner basis solvers proposed in [11]. However, in the case of equations (20), the coefficients are not fully independent. This means that using the default settings for the automatic generator [11] that initialize the coefficients of equations (20) by random values from $\mathbb{Z}_p$ does not lead to a correct solver. To obtain a working Gröbner basis solver, one has to create a correct problem instance with values from $\mathbb{Z}_p$ for the automatic generator [11] initialization. In case of system (20), we need to force the elements of the matrix $\mathtt{M}_2$ (11) to fulfill dependencies (12) and to force the elements $m_{6,8}, m_{6,10}, m_{7,8}$, and $m_{7,10}$ to vanish.

By enforcing these dependencies among the coefficients of the input system (20), we have obtained a Gröbner basis solver that performs one G-J elimination of a 16×21 matrix. This matrix contains coefficients which arise from the five specific 2D-to-2D point correspondences. Finally, the solutions to γ_2 and the radial distortion parameters λ and λ' are extracted from the eigenvectors of a 5×5 multiplication matrix created from the rows of the 16×21 template matrix after the G-J elimination. This gives us up to 5 real solutions. By substituting these solutions to the equations (10), the solutions to the homography $\mathtt{H}$ can be obtained.

4. Non-minimal Solver H6$\lambda_1\lambda_2$

In this section, we will describe the second proposed solver, a non-minimal 6-point solver H6$\lambda_1\lambda_2$. This solver is faster and simpler than the 5-point minimal H5$\lambda_1\lambda_2$ solver presented in section 3, however, it is slightly more sensitive to noise.

This solver starts, analogously to H5$\lambda_1\lambda_2$ solver, with equations (5), corresponding to the third row of the matrix equation (4). This time, however, we have six of these equations: $\bar{\mathtt{M}}_1 \mathbf{v}_1 = \mathbf{0}$, where $\bar{\mathtt{M}}_1$ is a 6×8 coefficient matrix. Again, assuming that $\bar{\mathtt{M}}_1$ has full rank of 6, *i.e.*, we have 6 non-degenerate image correspondences, the dimension of $\mathrm{null}(\mathtt{M}_1)$ is 2. Therefore, the monomial vector $\mathbf{v}_1$ can be rewritten as a linear combination of two null space basis vectors $\mathbf{v}_1 = \bar{\gamma}_1 \bar{\mathbf{n}}_1 + \bar{\mathbf{n}}_2$, where $\bar{\gamma}_1$ is the new unknown.

Next, we use the dependencies of the elements of the monomial vector $\mathbf{v}_1$ (7), $v_7 = \lambda v_3$ and $v_8 = \lambda v_6$. These, combined with the reparameterization of $\mathbf{v}_1$, lead to two quadratic equations in two unknowns $\bar{\gamma}_1$ and λ

$$\bar{\gamma}_1 \bar{n}_{17} + \bar{n}_{27} - \lambda(\bar{\gamma}_1 \bar{n}_{13} + \bar{n}_{23}) = 0, \qquad (21)$$

$$\bar{\gamma}_1 \bar{n}_{18} + \bar{n}_{28} - \lambda(\bar{\gamma}_1 \bar{n}_{16} + \bar{n}_{26}) = 0, \qquad (22)$$

where $\bar{n}_{ij}$ is the jth element of the vector $\bar{\mathbf{n}}_i$. This system can easily be solved, for example, using the hidden variable resultant method [5]. Using this method, we can rewrite the two equations as

$$\begin{bmatrix} -\bar{\gamma}_1 \bar{n}_{13} - \bar{n}_{23} & \bar{\gamma}_1 \bar{n}_{17} + \bar{n}_{27} \\ -\bar{\gamma}_1 \bar{n}_{16} - \bar{n}_{26} & \bar{\gamma}_1 \bar{n}_{18} + \bar{n}_{28} \end{bmatrix} \begin{bmatrix} \lambda \\ 1 \end{bmatrix} = \mathbf{0}. \quad (23)$$

The matrix equation (23) has a solution if and only if the determinant of its matrix is equal to zero which gives a quadratic equation in $\bar{\gamma}_1$. By solving this quadratic equation and by substituting the solution to (23), we obtain two solutions for $\bar{\gamma}_1$ and λ.

Next, from the equation $\mathbf{v}_1 = \bar{\gamma}_1 \bar{\mathbf{n}}_1 + \bar{\mathbf{n}}_2$ we obtain two solutions to the first two rows of the homography matrix $\mathtt{H}$, *e.g.*, $h_{11}, h_{12}, h_{13}, h_{21}, h_{22}$, and h_{23}.

Finally, the equations (10) corresponding to the second row of the matrix equation (4) for the six point correspondences lead to six linear equations in the last four unknowns h_{31}, h_{32}, h_{33} and λ'. We can simply solve this overdetermined system using SVD.

5. Experiments

To precisely gauge the stability, precision, and speed of H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$, we tested the solvers on synthetic as well as real world data with various amounts of radial distortions, noise levels, and scene configurations. We compared the proposed solvers with a comprehensive set of related algorithms—F9$\lambda_1\lambda_2$ [3], H4 [8], H3λf [2], and H5λ [7].
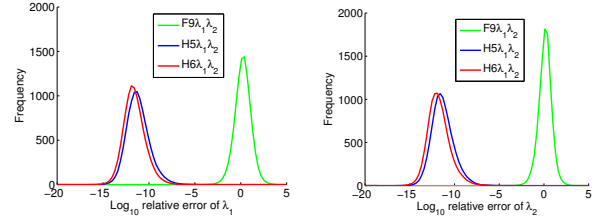

Figure 1: Comparison of H5$\lambda_1\lambda_2$ (Blue) and H6$\lambda_1\lambda_2$ (Red) with the F9$\lambda_1\lambda_2$ solver (Green) for two cameras with different radial distortions observing a planar scene. Histograms of log_{10} relative errors of estimated λ_1 (Left) and λ_2 (Right).

5.1. Synthetic data

First, we studied the performance of the new solvers on synthetically generated 3D scenes with known ground-truth parameters. These scenes consist of 3D points randomly distributed on a plane or in a 3D cube, depending on the testing configuration. Each 3D point was projected by two cameras with random, yet still realistic, focal lengths. The orientations and positions of the cameras were selected at random as to look at the scene (in the case of planar scenes) or so that the transformation between the two cameras was a pure random rotation (in the case of general 3D scenes). Next, the image points in both cameras were corrupted by different or equal amounts of radial distortion following the one-parameter division model [7]. Finally, the radially distorted image points were corrupted by Gaussian noise with standard deviation σ, assuming a 1000×1000 pixel image.

5.1.1 Numerical stability

In the numerical stability experiment, we studied the behavior of H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$ on noise-free data. We compared the results with the numerical stability of the most relevant solvers (F9$\lambda_1\lambda_2$ [3], H3λf [2], H5λ [7]), depending on the testing configuration.

In the experiments, we generated 10000 scenes with 3D points distributed at random on a plane or in a cube and cameras with random feasible poses or undergoing pure rotation, depending on the testing configuration. The radial distortion parameters λ_1 and λ_2 were drawn at random from the interval $[-0.7, 0]$ to show the behaviour of the algorithms for large as well as small amounts of radial distortion. For comparison with solvers H3λf [2] and H5λ [7] estimating one shared radial distortion parameter, the radial distortion parameters were set to be equal, $\lambda_1 = \lambda_2$.

Figure 1 shows the results of the solvers H5$\lambda_1\lambda_2$ (Blue) and H6$\lambda_1\lambda_2$ (Red) compared the F9$\lambda_1\lambda_2$ solver (Green) for scene configurations with two cameras with different radial distortions observing a planar scene. The log_{10} of the rela-
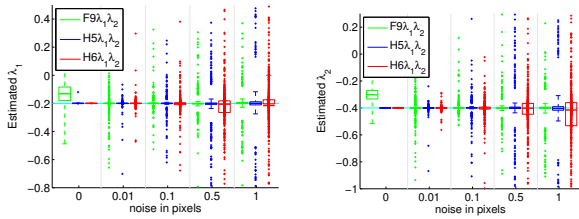
Figure 2: Comparison of H5$\lambda_1\lambda_2$ (Blue) and H6$\lambda_1\lambda_2$ (Red) with the F9$\lambda_1\lambda_2$ solver (Green) for two cameras with different radial distortions $\lambda_{1_{gt}} = -0.2$ and $\lambda_{2_{gt}} = -0.4$ observing a planar scene. Boxplots of estimated λ_1 (Left) and λ_2 (Right) for different noise levels.
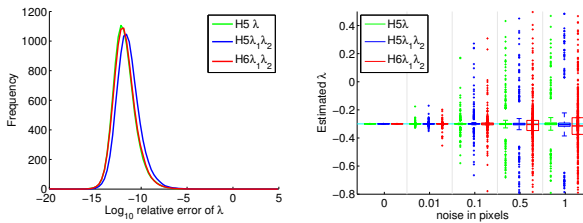

Figure 3: Comparison of H5$\lambda_1\lambda_2$ (Blue) and H6$\lambda_1\lambda_2$ (Red) with the H5λ solver (Green) for cameras with one shared radial distortion and a planar scene. (Left) Histograms of log_{10} relative errors of estimated λ's. (Right) Boxplots of estimated λ's for different noise levels and $\lambda_{gt} = -0.3$.

tive error of the radial distortion parameter λ_1 obtained by selecting the real root closest to the ground truth value is on the left and the log_{10} of the relative error of the radial distortion parameter λ_2 on the right. We can see that the numerical stability of both H5$\lambda_1\lambda_2$ (Blue) and H6$\lambda_1\lambda_2$ (Red) is virtually equivalent. The F9$\lambda_1\lambda_2$ solver for estimating fundamental matrix and two different radial distortion parameters fails for perfectly planar scenes. However, as will be seen from noise and real data experiments in the next, F9$\lambda_1\lambda_2$ gives quite precise estimates of radial distortion parameters in the presence of noise.

We performed an analogous numerical stability experiment for two cameras with one common radial distortion observing a plane. In time, we have compared the H5$\lambda_1\lambda_2$ (Blue) and H6$\lambda_1\lambda_2$ (Red) to the H5λ solver (Green). Figure 3 (Left) shows the log_{10} of the relative error of the estimated shared radial distortion parameter $\lambda = \lambda_1 = \lambda_2$. All three tested solvers have comparable numerical stability.

Finally, we tested the numerical stability of the proposed solvers in situations where the relative motion between cameras was pure rotation. The synthetic scenes contained random non-planar 3D points and radial distortion parameters were fixed to be equal for both views. In this experiment, we compared H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$ to the H3λf [2] solver. As we can see in Figure 4 (Left), H5$\lambda_1\lambda_2$
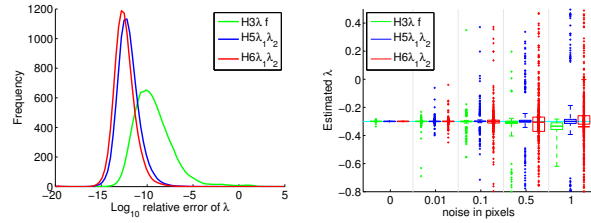

Figure 4: Comparison of H6$\lambda_1\lambda_2$ (Red) and H5$\lambda_1\lambda_2$ (Blue) with the H3λf solver (Green) in the case when the relative motion between cameras is a pure rotation and both cameras share the same radial distortion. (Left) Histograms of log_{10} relative errors of estimated λ's. (Right) Boxplots of estimated λ's for different noise levels and $\lambda_{gt} = -0.3$.

(Blue) and H6$\lambda_1\lambda_2$ (Red) are numerically more stable than the H3λf (Green). This stability issue of H3λf stems from the fact that quite a large G-J elimination needs to be performed by this solver.

5.1.2 Noise experiment

In the next section, we study the performance of the proposed solvers on the noised image correspondences.

Figure 2 shows the estimated radial distortion parameters for cameras with different radial distortions. In this case the ground truth radial distortion parameters were set to $k_{1_{gt}} = -0.2$ and $k_{2_{gt}} = -0.4$ and we compared the proposed solvers to the F9$\lambda_1\lambda_2$ solver (Green). Results in Figure 2 are depicted using MATLAB function `boxplot` which shows 25% to 75% quantile values as boxes with a horizontal line for median. The crosses show data beyond 1.5 times the interquartile range. In the presence of noise, the F9$\lambda_1\lambda_2$ solver provides quite precise estimates of radial distortion parameters. However, the estimated epipolar geometries were invalid for these noisy planar scenes.

We have performed an analogous noise experiments for cameras with one shared radial distortion parameter $\lambda_{gt} = -0.3$ observing a plane (Figure 3 (Right)) and cameras undergoing pure rotation (Figure 4 (Right)). The performance of the minimal H5$\lambda_1\lambda_2$ solver is very similar to the performance of the competing solvers in the presence of noise. However, the proposed H5$\lambda_1\lambda_2$ solver is more general than both H3λf and H5λ, *i.e.*, it could handle all tested situations (different distortions, planar scenes, pure rotation). Further, the new H5$\lambda_1\lambda_2$ solver has much lower computational complexity than both F9$\lambda_1\lambda_2$ and H3λf. The new non-minimal H6$\lambda_1\lambda_2$ solver is slightly more sensitive to noise than H5$\lambda_1\lambda_2$. This is because H6$\lambda_1\lambda_2$ first uses all six correspondences to estimate the first two rows of H to exactly satisfy these constraints on all six points. Only then the third row is estimated as a least square fit to these six correspondences.
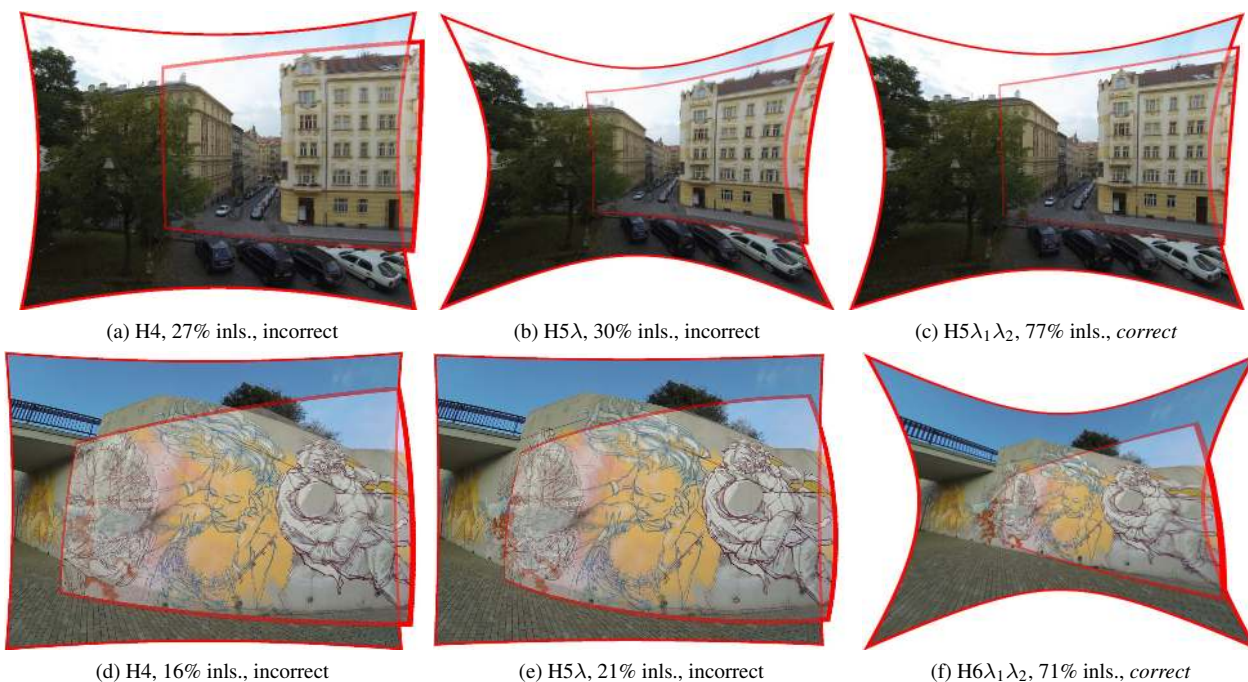
(a) H4, 27% inls., incorrect   (b) H5λ, 30% inls., incorrect   (c) H5$\lambda_1\lambda_2$, 77% inls., *correct*

(d) H4, 16% inls., incorrect   (e) H5λ, 21% inls., incorrect   (f) H6$\lambda_1\lambda_2$, 71% inls., *correct*

Figure 5: *Image stitching.* Stitching of (a–c) images connected by rotation induced homography initialized by H4, H5λ, and H5$\lambda_1\lambda_2$, respectively, (d–f) images connected by plane induced homography initialized by H4, H5λ, and H6$\lambda_1\lambda_2$, respectively. The percentages show the ratio of the number of inliers found by LO-RANSAC for the respective solver to the number of tentative matches.

5.2. Real data

In this section, we present several homography estimation applications to show the superior performance of the proposed solvers in real world situations where radial distortion is present.

5.2.1 Image Stitching Experiment

Stitching of images connected by a homography transformation induced either by a plane or camera rotation is a classical problem of computer vision. Typically, tentative image correspondences are validated in a RANSAC-style loop, followed by a local optimization step that can optionally include more parameters.

Figures 5(a–c) show an image pair connected by camera rotation induced homography with different radial distortions λ_1, λ_2 stitched together using different solvers. The images were taken by GoPro Hero3 and PowerShot SX260 cameras. In all cases we used LO-RANSAC loop [4] fixed to 100 iterations, followed by local optimization step where both λ_1, λ_2 were optimized. Figures 5(a–c) show the result of image stitching initialized by H4, H5λ, and H5$\lambda_1\lambda_2$, respectively. We can clearly see that in this case H4 and H5λ solvers did not provide sufficiently close model for local optimization step to find a satisfactory result. In the case of H5$\lambda_1\lambda_2$, both λ_1 and λ_2 were correctly recovered.

Figures 5(d–f) show an image pair connected by a plane induced homography, again, with different radial distortions λ_1, λ_2 stitched together using different solvers. The images were taken by a GoPro Hero3 camera and and a HTC Desire 500 mobile phone. This time, we used H4, H5λ, and H6$\lambda_1\lambda_2$, respectively. Similarly to the previous case, initializations by neither H4 nor H5λ were good enough to help the final local optimization to recover the correct solution.

Figure 6 shows a number of gained inliers as a function of LO-RANSAC cycles for an image pair connected by a plane induced homography. The algorithm F9$\lambda_1\lambda_2$ recovers the maximal number of inliers first. This is not surprising, since the fundamental matrix F models planar scenes too. We have observed that if seven out of the nine points lie on a plane, then the fundamental matrix returned by F9$\lambda_1\lambda_2$ covers the plane. Interestingly enough, F9$\lambda_1\lambda_2$ recovers correct radial distortions even if seven points are on a plane and the remaining two are mismatches (a rigorous study of this behavior is out of scope of this paper). Since F9$\lambda_1\lambda_2$ will classify these mismatches as correct ones, the inlier set is bigger and the geometry recovered by F9$\lambda_1\lambda_2$ is invalid. Hence additional filtration of these mismatches is necessary. Moreover, F9$\lambda_1\lambda_2$ is very slow and therefore not useful in real RANSAC applications. The only two solvers that recover all inliers as well as the correct geometry are H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$.
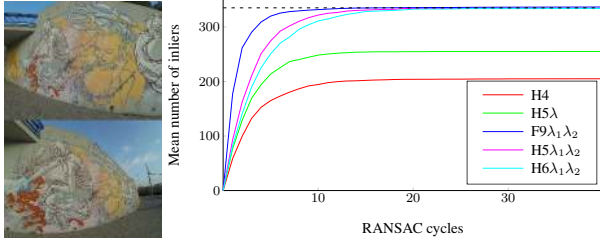
Figure 6: *No. of inliers as a function of LO-RANSAC cycles.*

5.2.2 SfM Seed Score Experiment

Homography estimation also plays an important role in SfM pipelines [12], where it is used to select a correct image pair to seed an incremental SfM reconstruction. This is usually done by finding the epipolar geometry inliers, *e.g.*, using the F7 algorithm [8] with a loose RANSAC threshold and by estimating homography inliers among the inliers selected by F7. The ratio of the homography and the epipolar geometry inliers is sometimes called *seed score*. If the score is close to 1, the image pair is unsuitable as a SfM seed, since it lacks the needed 3D structure. If radial distortion is not modeled, then a point on the plane may by wrongly classified as an off-the-plane point due to the distortion. As a consequence, a completely planar scene can be classified as a good seed. This is shown in the next table, which shows seed scores for an image pair similar to the first image pair from section (5.2.1) connected by a rotation induced homography. The rows show results for different solvers; the columns show results for different threshold combinations (H* thr./F7 thr.). We used LO-RANSAC to recover the respective inlier sets.

	10/3	20/3	10/5	20/5
H4 / F7	0.0985	0.2304	0.2868	0.2845
H5λ / F7	0.3815	0.3524	0.4628	0.4477
H5$\lambda_1\lambda_2$ / F7	0.8620	0.8513	0.9373	0.9299

The table shows that by using H5$\lambda_1\lambda_2$ we can clearly distinguish the given image pair as unsuitable for SfM seeding for wide range of RANSAC thresholds.

The next table shows analogous statistics for an image pair connected by a plane induced homography. The pair was captured by a GoPro Hero3 camera with different focal length settings.

	10/3	20/3	10/5	20/5
H4 / F7	0.1674	0.1593	0.2942	0.2210
H5λ / F7	0.4634	0.4181	0.6528	0.5399
H5$\lambda_1\lambda_2$ / F7	0.9268	0.8884	0.9711	0.9513

Again, H5$\lambda_1\lambda_2$ correctly classifies as unsuitable seed pair.

5.3. Computational complexity

The significant improvement of the proposed solvers over the state-of-the-art solvers F9$\lambda_1\lambda_2$ [3], H3λf [2], and H5λ [7] is in the speedup. The C++ implementations of the H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$ solvers run in about 2.6 μs and 1.5 μs on an Intel i7-4700MQ 2.4Ghz based laptop, respectively. This is comparable to the speed of the classical perspective linear H4 algorithm [8] that computes the null space of a 8×9 matrix and runs about 1.8 μs on the same machine.

Unfortunately, we only have MATLAB implementations of the other three related solvers (F9$\lambda_1\lambda_2$ (9.2 ms), H3λf (3.3 ms), H5λ (1.2 ms)). It would not be fair to compare these to the C++ implementation of the proposed solvers. However, we can at least perform theoretical computational complexity analysis based on the mathematical operations performed by the solvers.

The H5$\lambda_1\lambda_2$ solver needs to compute the null space of a 5×8 matrix, to perform G-J elimination of a 16×21 matrix and to find eigenvalues of a 5×5 matrix. The H6$\lambda_1\lambda_2$ solver needs to compute the null space of a 8×8 matrix, the inverse of a 4×4 matrix, and to solve one quadratic equation.

For comparison, F9$\lambda_1\lambda_2$ performs LU decomposition of a 393×390 matrix and the eigenvalue computations of a 24×24 matrix. The panorama stitching H3λf solver performs LU decomposition of a 90×132 matrix and the eigenvalue computations of a 25×25 matrix. Finally, the polynomial eigenvalue solver H5λ needs to compute eigenvalues and eigenvectors of a 18×18 matrix. These operations make both F9$\lambda_1\lambda_2$ and H3λf significantly slower than the proposed solvers.

The G-J (LU) parts of the Gröbner basis solvers F9$\lambda_1\lambda_2$ [3] and H3λf [2] can be potentially speeded up using the recently published SBBD method [10]. However, the expected speed up of these G-J parts is only 3–5$\times$. Therefore, the final F9$\lambda_1\lambda_2$ and H3λf solvers will still be quite slow compared to H5$\lambda_1\lambda_2$ and H6$\lambda_1\lambda_2$.

6. Conclusion

The estimation of a homography connecting two views is a crucial problem in computer vision with many applications. Ignoring the radial distortion component in the homography estimation may lead to significant errors or totally invalid estimates. In this paper, we presented two novel algorithms for estimating homography between two cameras with different radial distortions. Both algorithms can handle planar scenes as well as scenes where the relative motion between the cameras is a pure rotation. The proposed solvers are more general than the related state-of-the-art H3λf [2] and H5λ [7] solvers and have much lower computational complexity than both F9$\lambda_1\lambda_2$ [3] and H3λf [2]. The experiments show that the proposed algorithms are stable, very fast ($\approx$ 2 μs), and can be efficiently used inside a RANSAC loop.

Acknowledgment

References

[1] J. P. Barreto and K. Daniilidis. Fundamental matrix for cameras with radial distortion. In *Proc. ICCV*, pages 625–632, 2005. 1

[2] M. Byröd, M. Brown, and K. Åström. Minimal solutions for panoramic stitching with radial distortion. In *BMVC'09*, 2009. 1, 2, 5, 6, 8

[3] M. Byröd, Z. Kukelova, K. Josephson, T. Pajdla, and K. Åström. Fast and robust numerical solutions to minimal problems for cameras with radial distortion. In *CVPR'08*, 2008. 1, 2, 5, 8

[4] O. Chum, J. Matas, and J. Kittler. Locally optimized ransac. In *Pattern Recognition*, pages 236–243. Springer Berlin Heidelberg, 2003. 1, 2, 7

[5] D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Springer, 2nd edition, 2005. 4, 5

[6] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. 1, 2

[7] A. W. Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *CVPR'01*, volume 1, page 125, Los Alamitos, CA, USA, 2001. IEEE Computer Society. 1, 2, 3, 5, 8

[8] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 1, 2, 5, 8

[9] H. Jin. A three-point minimal solution for panoramic stitching with lens distortion. In *CVPR'08*, 2008. 1

[10] Z. Kukelova, M. Bujnak, J.Heller, and T. Pajdla. Singly-bordered block-diagonal form for minimal problem solvers. In *ACCV'14*, 2014. 8

[11] Z. Kukelova, M. Bujnak, and T. Pajdla. Automatic generator of minimal problem solvers. In *ECCV'08, Part III*, volume 5304 of *Lecture Notes in Computer Science*, 2008. 4

[12] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: exploring photo collections in 3D. In *ACM SIGGRAPH'06*, 2006. 2, 8

[13] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, ICCV '99, pages 298–372. Springer-Verlag, 2000. 1