Revisiting Radial Distortion Absolute Pose

Viktor Larsson[1] Torsten Sattler[2] Zuzana Kukelova[3] Marc Pollefeys[1,4]
[1]Department of Computer Science, ETH Zurich [2]Chalmers University of Technology
[3]VRG, Department of Cybernetics, Czech Technical University in Prague [4]Microsoft

Abstract

To model radial distortion there are two main approaches; either the image points are undistorted such that they correspond to pinhole projections, or the pinhole projections are distorted such that they align with the image measurements. Depending on the application, either of the two approaches can be more suitable. For example, distortion models are commonly used in Structure-from-Motion since they simplify measuring the reprojection error in images. Surprisingly, all previous minimal solvers for pose estimation with radial distortion use undistortion models. In this paper we aim to fill this gap in the literature by proposing the first minimal solvers which can jointly estimate distortion models together with camera pose. We present a general approach which can handle rational models of arbitrary degree for both distortion and undistortion.

1. Introduction

Jointly estimating the 6 degree-of-freedom (6DOF) pose of a camera and its intrinsic calibration is an important problem with applications in camera calibration [2], visual localization [37], and Structure-from-Motion (SfM) [38, 41] for uncalibrated images, e.g. from Internet photo collections such as Flickr. A common approach to this problem is to use a minimal solver [13, 20, 28, 29] inside a RANSAC [11] loop. Given a set of 2D-3D matches between pixels in an image and 3D point coordinates in the scene, the solvers provide an initial estimate of the intrinsic and extrinsic parameters. Later, the best model estimated by RANSAC is typically refined by minimizing the reprojection errors [46].

The images taken by real cameras do not perfectly follow a pinhole model. For perfectly centered lenses consisting of flawless elements, the distortion is radially symmetric around the principal point[1]. Such lens distortion is known as radial distortion. Due to manufacturing imperfections, the optical centers of various lens elements may be misaligned and the CCD sensor array may not be perfectly perpendicular to the optical axis. These misalignments can be modeled by adding a tangential component. For most modern cameras, the radial component of the lens distortion is dominant and the tangential component is often negligible. Therefore, most methods for camera calibration or joint estimation of camera pose and its intrinsic calibration consider distortion models consisting only of the radial component. The tangential component is either neglected entirely or is estimated only in a final, non-linear, refinement step.

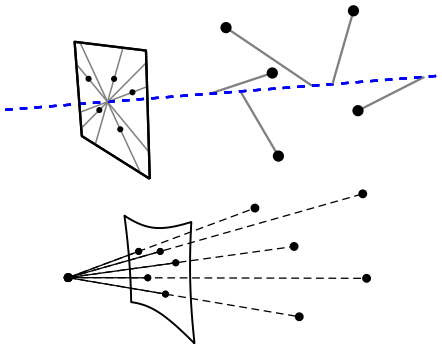

Figure 1: Two-step approach for radial distortion absolute pose. First, using radial alignment constraints we estimate the pose of the camera up to an unknown translation along the principal axis (blue line). Using the full projection equations we then solve for the remaining translational degree of freedom jointly with focal length and distortion parameters.

There are two approaches for handling radial distortion: *undistortion models* estimate a function that maps the original distorted 2D pixel measurements to undistorted measurements. Such models are useful if unprojection, i.e. mapping from image space to world space, is of particular interest. For example, undistorting an image for processing in Multi-View Stereo [39, 40]. In the context of minimal solvers, the main advantage of undistortion models is that they decouple the camera pose parameters, *i.e.*, the parameters describing the rotation and translation of the camera, from the distortion parameters. To the best of our knowledge, all minimal solvers for problems with radial distortion [6, 12, 16, 20–22, 24, 25, 27–29, 34–36] in the literature are based on undistortion models.

Distortion models determine a mapping from undistorted

[1]Which often gives a good approximation of the true distortion center.

projections of 3D points to the corresponding distorted measurements. Such models are preferable if the applications requires projecting points into image space, *e.g.*, phone-based Augmented Reality applications. Distortion models are also common in Structure-from-Motion pipelines and other 3D vision frameworks (see Table 3): it is usually preferable to minimize reprojection error in the original image, *e.g.*, for triangulation [24], and distortion models allow for easier computation of reprojection errors in image space.

Due to a lack of distortion-based solvers, one approach is to first estimate the camera pose using an undistortion-based solver and then re-fit a distortion model on the inliers. However, there is usually no closed form solution for converting between two distortion/undistortion models. Rather, one needs to estimate the parameters using iterative (least-squares) optimization. This implies that the undistortion model needs to accurately describe the camera distortion. While this approach is likely to work in many cases, it potentially introduces additional failure modes.

In this paper, we present a general framework for minimal absolute pose solvers which can handle both distortion and undistortion models. Our approach is based on two stages: Following [7, 20], the first stage uses a 1D camera model [44] to estimate the pose of the camera up to an unobservable translation along its principal axis. In the second stage, we apply novel minimal solvers to jointly determine the radial distortion parameters of the camera and the remaining pose parameter. In detail, this paper makes the following contributions: **1)** we present a framework for estimating the absolute pose of a camera together with a distortion model. To the best of our knowledge, ours is the first work to use a distortion rather than an undistortion model. **2)** We derive novel minimal solvers for the family of rational distortion models, covering both polynomial and division models. **3)** Detailed experiments compare our solvers to state-of-the-art undistortion-based pose solvers, demonstrating similar performance.

2. Related Work

Lens distortion models. Research on camera lens distortions dates back to the beginning of the 20th century. Due to some manufacturing imperfections, the optical centers of various lens elements may be misaligned and CCD sensor arrays may not be perpendicular to the optical axis. These misalignments can be modeled by adding a tangential component to the radial component of the distortion. In 1919, Conrady [9] introduced the first decentering distortion model. Later in 1966, this model was improved by the now commonly used Brown-Conrady model [5]. It models the radial and tangential components of the distortion by placing a thin prism in front of a perfectly centered lens. The model is used in several popular calibration toolboxes [2, 3] for precise calibration of camera distortion.

As outlined above, radial distortion is the dominant distortion effect observed in modern cameras. Radial distortion is usually modeled using a rational function [4,8,32,43] that is dependent on the radius, *i.e.*, on the distance of the image point from the distortion center. Rational functions can be used to model either distortion or undistortion. Radial distortion models [8, 12, 43] differ in the degree and form of the polynomials used in numerators and denominators of these rational functions. For some special rational distortion functions, an analytical inverse formula can be derived [32]. However, in general these models lack an analytical inverse and the inversion can only be done locally using iterative schemes.

Popular sub-classes of rational distortion models are polynomial and division models [12]. For example, the Brown-Conrady model [5] uses a polynomial function to model radial distortion. Division models, popularized by Fitzgibbon [12], contain radial distortion parameters only in the denominator. They are often preferred over polynomial models in minimal solvers (see *e.g.* [6, 16, 20–22, 24, 25, 28, 35, 36]) as they usually result in simpler equations and can get sufficient accuracy even with few parameters.

In general, for rational radial distortion models the distortion is dominated especially by the first few terms. Moreover, it has been shown that using polynomials with higher-order terms may cause numerical instabilities [47]. Therefore, most of the commonly used radial distortion models contain only up to 3 parameters in numerator and / or denominator. Section 3 provides a more detailed description of the different radial distortion models used in this paper.

Absolute pose estimation. Given n 2D-3D point matches, the absolute camera pose estimation problem (PnP) aims to estimate the rotation matrix R and the translation vector t describing the absolute camera pose w.r.t. the coordinate system of the 3D points, and potentially (part of) the intrinsic calibration of the camera.

For absolute pose estimation without radial distortion many different minimal or non-minimal solutions exist, *i.e.* P3P and PnP solutions for calibrated cameras [11, 13, 17, 31], P3.5Pf and P4Pf solutions for cameras with unknown focal length [26, 28, 50, 51] or P4.5Pfuv and P5Pfuv solvers for cameras with unknown focal length and principal point [28, 45]. There are also solvers that model other non-linear distortions, e.g. from rolling-shutter [1, 18].

Since consumer photography is now dominated by mobile-phone and wide-angle cameras (*e.g.* GoPro-type cameras), images with significant radial lens distortion are increasingly common. Therefore, several solutions to the absolute camera pose estimation problem with unknown radial distortion appeared recently. To our knowledge all existing radial distortion absolute pose solvers use the division model [12] to model the undistortion function. Josephson

and Byröd [16] simultaneously estimated the absolute pose, the focal length, and a one-parameter division undistortion model from the minimal number of four 2D-3D point correspondences (P4Pfr). To reduce the size of the solver, Bujnak *et al*. [6] split this P4Pfr problem into planar and non-planar cases. Recently, Larsson *et al*. [28] significantly reduced the elimination template size and proposed a degenerate-free minimal solution to the P4Pfr problem.

Kukelova *et al*. [20] used five point correspondences to solve the absolute pose problem with unknown focal length and up to three parameters for a division undistortion model. This method uses the property that the camera pose can be estimated up to a translation along its principal axis using five point correspondences and a 1D camera model [44]. In a second step, [20] then estimated the remaining translation parameter, the unknown focal length and the undistortion parameters. Nakano [34] extended [20] to solve the PnPfr problem for $n \geq 5$ point matches in the least square sense.

Larsson *et al*. [29] used a similar two step method based on the 1D radial camera model to first estimate the camera pose up to a translation along its principal axis together with the unknown principal point from seven point correspondences. In the second step the unknown translation parameter, focal length and one-parameter division undistortion model were estimated linearly in a least-square sense.

In this paper, we propose multiple solvers for rational distortion models. Following [20], we first estimate most of the camera pose parameters in a first stage and then solve for the remaining translation parameter and the distortion parameters jointly in a second stage.

3. Background

3.1. Radial Distortion

Unlike focal length, which uniformly scales the image, radial distortion is characterized by scaling the image points differently depending on the their distance to the distortion center. Typically points further from the distortion center are more heavily distorted. Since the magnitude of radial distortion only depends on the distance to the image center it can be modelled as

$$\boldsymbol{x}' = h(\|\boldsymbol{x}\|)\boldsymbol{x}, \tag{1}$$

if the coordinate system is aligned with the distortion center.

Radial distortion models can be divided into two main categories; *distortion* models and *undistortion* models. In a distortion model, the pinhole projections are distorted such that they match the observed image points, *i.e*.

$$K^{-1}\boldsymbol{x} = D\left(\pi\left(R\boldsymbol{X} + \boldsymbol{t}\right)\right) \tag{2}$$

where π is the pinhole projection and D is some non-linear function on the form (1) which models the distortion. Conversely, in an undistortion model, the image points are undistorted such that they match the pinhole projections, *i.e*.

$$D(K^{-1}\boldsymbol{x}) = \pi\left(R\boldsymbol{X} + \boldsymbol{t}\right). \tag{3}$$

The non-linear distortion mappings D are usually approximated with a rational function,

$$h(r) = \frac{1 + \mu_1 r^2 + \mu_2 r^4 + \mu_3 r^6 + \dots}{1 + \lambda_1 r^2 + \lambda_2 r^4 + \lambda_3 r^6 + \dots}. \tag{4}$$

It is common to either only include non-linear terms in the numerator, giving rise to the polynomial models,

$$h(r) = 1 + \mu_1 r^2 + \mu_2 r^4 + \mu_3 r^6 + \dots, \tag{5}$$

or analogously the division model,

$$h(r) = \frac{1}{1 + \lambda_1 r^2 + \lambda_2 r^4 + \lambda_3 r^6 + \dots}. \tag{6}$$

As far as the authors know, all previous work on minimal solvers uses the division model for undistortion, as in (3), since it in general results in simpler equation systems.

3.2. Pose under Radial Projections

Determining the 6 DOF pose of a camera with unknown radial distortion is a difficult task due to the extra non-linearities in the projection equations. One approach, originally introduced by Tsai [47], is to only consider the projections modulo a radial scaling. In the context of camera resectioning, this means that we only require that the projections lie on the radial lines passing though the image points. See Figure 1 for an illustration. While these constraint are weaker, they are also invariant to changes in focal length and radial distortion, since these will just move projections along these lines. Unfortunately, these constraints are also invariant to translation along the principal axis, thus it is only possible to recover the camera pose up to this unobservable forward translation using these constraints. This projection model is sometimes called the *1D radial camera* model, see [44] and the supplementary material for more details. The resection problem for 1D radial cameras can be linearly solved using 7 point correspondences (see [7]). The minimal problem (with 5 points) was solved in [20] as part of their two-step distortion pose solver. This was recently extended to also estimate the principal point (from 7 points) by Larsson et al. [29].

4. Radial Distortion Absolute Pose

Our approach for radial distortion absolute pose follows the two-step approach used in previous work [7,20], by first estimating a 1D radial camera and then estimating the focal length/distortion parameters jointly with the remaining degree of freedom in the translation. Compared to previous

work we consider more general distortion models, for both distortion and undistortion.

In the following sections we assume that we have estimated a 1D radial camera. Using this we can then transform the 3D points into the camera's local coordinate system,

$$\boldsymbol{X}_c = \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \boldsymbol{X}_g + \begin{pmatrix} t_1 \\ t_2 \\ 0 \end{pmatrix}. \quad (7)$$

Note that these points are only given up to an unknown translation in the z-direction and possibly a flipped sign for the z-axis (corresponding to a negative focal length).

4.1. Rational Undistortion Model

We begin by considering the case of undistortion, *i.e.* the rational model is used to undistort the observed image points. This case is slightly simpler compared to the distortion case, but we will show that a similar approach can be applied to both problems.

The translation parameter t_3 will move the points (7) along the z-axis. The goal is now to determine t_3 such that the projection equations for the undistortion model (3), i.e.

$$h(\|K^{-1}\boldsymbol{x}\|)K^{-1}\boldsymbol{x} = \pi(\boldsymbol{X}_c + \boldsymbol{e}_3 t_3) \quad (8)$$

are satisfied.[2] For a rational model (4), this becomes,

$$\frac{1 + \sum_{i=1}^{n_p} \mu_i (\frac{r}{f})^{2i}}{1 + \sum_{i=1}^{n_d} \lambda_i (\frac{r}{f})^{2i}} \cdot \frac{1}{f}\boldsymbol{x} = \frac{1}{z + t_3} \begin{pmatrix} x \\ y \end{pmatrix}, \quad (9)$$

where $\boldsymbol{x}$ is the observed (distorted) image point, $r = \|\boldsymbol{x}\|$ and $\left(x, y, z + t_3\right)^T$ is the 3D point in the local coordinate system of the camera (see previous section). The change of variables, $\mu_i \to \mu_i f^{2i}$ and $\lambda_i \to \lambda_i f^{2i-1}$, yields

$$\frac{1 + \sum_{i=1}^{n_p} \mu_i r^{2i}}{f + \sum_{i=1}^{n_d} \lambda_i r^{2i}} \boldsymbol{x} = \frac{1}{z + t_3} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (10)$$

While it may seem that (10) gives two constraints (since there are two equations), recall that the unknowns $(t_3, f, \boldsymbol{\lambda}, \boldsymbol{\mu})$ only move the projections along the radial lines. Thus only the radial component of (10) gives us a constraint. Taking the dot product of (10) with $\boldsymbol{x}$ and multiplying with the denominators, we get

$$(z+t_3)\left(1 + \sum_{i=1}^{n_p} \mu_i r^{2i}\right) - \alpha\left(f + \sum_{i=1}^{n_d} \lambda_i r^{2i}\right) = 0, \quad (11)$$

where $\alpha = \boldsymbol{x}^T(x, y)^T / \|\boldsymbol{x}\|^2$. This is a polynomial equation of degree $\min(2, n_p + 1)$ in the unknowns $(t_3, f, \boldsymbol{\lambda}, \boldsymbol{\mu})$. Since each 2D-3D correspondence yields one such equation, we need $2 + n_p + n_d$ points to get a minimal problem.

[2] We assume that the camera has zero skew, unit aspect ratio and centered principal point, i.e. $K = \mathrm{diag}(f, f, 1)$

The unknowns $(f, \boldsymbol{\lambda})$ appears linearly in (11) and can be easily eliminated. For a pure division model (*i.e.* $n_p = 0$, which was considered in [20]) all of the equations are in fact linear and the problem is reduced to solving a simple linear system (as was also done in [20]). However, in general ($n_p \neq 0$), the equations will be non-linear polynomials. To tackle these we use the hidden variable trick [10]. Hiding t_3 after linearly eliminating f and $\boldsymbol{\lambda}$ yields

$$\begin{bmatrix} \\ A(t_3) \\ \\ \end{bmatrix} \begin{pmatrix} \boldsymbol{\mu} \\ 1 \end{pmatrix} = 0, \quad (12)$$

where $A(t_3) = \sum_{k=0}^{n} t_3^k A_k$ and each A_k and A_k are constant matrices.

4.1.1 Reduction to Univariate Polynomial

Since the matrix $A(t_3)$ is rank-deficient we must have that

$$p(t_3) = \det(A(t_3)) = 0. \quad (13)$$

This gives us a single univariate polynomial in t_3, whose roots can be efficiently found using root-bracketing methods such as Sturm sequences [15].

4.1.2 Polynomial Eigenvalue Formulation

Another approach is to formulate equation (12) as a polynomial eigenvalue problem (PEP), see *e.g.* [19] for more details. Equation (12) can be rewritten as

$$\left(A_0 + A_1 t_3 + A_2 t_3^2 + \dots\right) \begin{pmatrix} \boldsymbol{\mu} \\ 1 \end{pmatrix} = 0, \quad (14)$$

and can be further transformed to a generalized eigenvalue problem (GEP) [10] and solved using standard efficient eigenvalue solvers. Note that the GEP formulation can introduce additional spurious solutions. As described in [19] we can reduce the size of the eigenvalue problem by removing columns and rows which only contain zeroes. These correspond to parasitic solutions that were introduced by transformation of PEP to GEP. In our solvers, we were able to remove all spurious solutions in this way.

4.2. Rational Distortion Model

We now turn our attention to the case of the rational distortion model. In this case, we have that the projection of the 3D point, *i.e.* $(x, y)/(z + t_3)$, undergoes rational distortion such that it matches the image point (as in (2)). This case is more difficult since the radius (used in the distortion) depends on the unknown t_3, which makes the equations highly non-linear. The reprojection equations are

$$\boldsymbol{x} = \frac{f + \sum_{i=1}^{n_p} \mu_i (\frac{r}{d})^{2i}}{1 + \sum_{i=1}^{n_d} \lambda_i (\frac{r}{d})^{2i}} \begin{pmatrix} x/d \\ y/d \end{pmatrix}, \quad (15)$$

where $r = \sqrt{x^2 + y^2}$ and $d = z + t_3$. Here we have made the change of variables, $\mu_i \to \mu_i / f$. Note that $d = z + t_3$ will be different for each point, thus we cannot remove the extra non-linearity with a simple change of variables as was the case for the undistortion model. Taking the dot product with $\boldsymbol{x}$ and multiplying with the denominators we get

$$d^n + \sum_{i=1}^{n_d} \lambda_i r^{2i} d^{n-2i} - \alpha f d^{n-1} + \alpha \sum_{i=1}^{n_p} \mu_i r^{2i} d^{n-2i-1} = 0, \tag{16}$$

where $n = \max(2n_p + 1, 2n_d)$ and α is defined as before. This is a polynomial equation of degree n. In contrast to the undistortion case, only one of the unknowns appear linearly in these equations (either μ_{n_p} or λ_{n_d}).

Note that while the polynomial system is of degree n, only the translation t_3 appears in higher order degrees. Again, this makes the hidden variable technique [10] especially suitable. Hiding t_3 the equations can be written as

$$\begin{bmatrix} & A(t_3) & \end{bmatrix} \begin{pmatrix} f \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu} \\ 1 \end{pmatrix} = 0, \tag{17}$$

where $A(t_3) = \sum_{k=0}^{n} t_3^k A_k$ and A_k are constant matrices. Using the same approach as in Section 4.1.1 or Section 4.1.2 we can solve for the unknown translation t_3.

4.3. Fronto-parallel Plane Degeneracy

It is well-known that estimating the absolute pose jointly with focal length is degenerate for fronto-parallel planar scenes. The problem is that it becomes impossible to disambiguate forward motion (changes in t_3) from changes in the focal length. Consider (15): For fronto-parallel planar scenes we may w.l.o.g. assume $z = 0$ for all points. In this case the unknowns in (15) only appear as quotients of t_3, *i.e.* $f/t_3, \mu_i/t_3^{2i}, \lambda_i/t_3^{2i}$. Thus any change in t_3 can be compensated by the corresponding scaling of $f, \boldsymbol{\mu}, \boldsymbol{\lambda}$.

4.4. Recovering Distortion Parameters

Once the forward translation t_3 has been determined, it is easy to recover the remaining unknowns. Inserting t_3 into (10) or (15) and multiplying with the denominators yields linear equations in $f, \boldsymbol{\mu}$ and $\boldsymbol{\lambda}$. For each solution t_3 returned from the solver, these equations will have a single exact solution. Of course, since these are just linear equations it also naturally extends to estimation from non-minimal samples, by simply solving the equations in a least squares sense, *i.e.*

$$\min_{f,\boldsymbol{\mu},\boldsymbol{\lambda}} \|M(f, \boldsymbol{\mu}, \boldsymbol{\lambda})^\top - \boldsymbol{b}\|^2. \tag{18}$$

In practice we found that adding a small dampening factor to the distortion parameters to prevent overfitting, *i.e.*

$$\min_{f,\boldsymbol{\mu},\boldsymbol{\lambda}} \|M(f, \boldsymbol{\mu}, \boldsymbol{\lambda})^\top - \boldsymbol{b}\|^2 + \epsilon\|\boldsymbol{\mu}\|^2 + \epsilon\|\boldsymbol{\lambda}\|^2, \tag{19}$$

D(μ, λ)		μ			
		0	1	2	3
λ	0	1	5	11	**19**
	1	3	6	**14**	24
	2	9	**10**	15	27
	3	**17**	20	21	28

U(μ, λ)		μ			
		0	1	2	3
λ	0	1	2	3	**4**
	1	1	2	**3**	4
	2	1	**2**	3	4
	3	**1**	2	3	4

Table 1: Number of solutions for rational model for radial distortion. Problems which are minimal together with camera pose estimation are highlighted in bold. The first column in in the right table corresponds to the division solver from Kukelova et al. [20].

for some small $\epsilon > 0$ usually gives slightly better results.

In [14], the authors show how to solve such a optimization problems with additional constraints, e.g. that the first and second derivative of the distortion function should be positive (which is the case for barrel distortion), using SDP relaxations. This method could essentially be used as a drop-in replacement of the linear least squares solution (19). However, since the method comes with a significantly higher computational cost we do not consider it in the experimental evaluation.

4.5. Implementation Details

In the previous sections we derived a general method which can handle distortion models of arbitrary degrees. We have implemented code for generating C++/MATLAB solvers using the proposed approach. The complexity of the solvers varies with the complexity of the distortion model. Table 1 shows the number of solutions w.r.t. the degree of the rational function. Note that this is for the upgrade step where we estimate the forward translation t_3. It is performed for each pose estimate returned by the 1D radial solver (at most four). In [23], it was reported that radial P5P returns on average 1.7 real solutions for non-planar scenes.

Runtime. The runtime of a C++ implementation of the solver is shown in Table 2. The table includes the solvers based on extracting a univariate polynomial (Section 4.1.1) as well as the polynomial eigenvalue formulation (Section 4.1.2), which are significantly slower in comparison.

Univ. D(μ, λ)		μ			
		0	1	2	3
λ	0	0.2	1.4	5.3	15.7
	1	0.7	2.0	8.7	25.6
	2	3.5	4.6	9.8	42.8
	3	12.4	16.8	21.5	64.0

Univ. U(μ, λ)		μ			
		0	1	2	3
λ	0	0.2	0.4	0.7	1.1
	1	0.2	0.3	0.7	1.1
	2	0.3	0.4	0.7	1.2
	3	0.4	0.6	0.9	1.4

PEP D(μ, λ)		μ			
		0	1	2	3
λ	0	-	3.8	16.3	49.0
	1	1.3	4.8	24.7	80.2
	2	11.2	13.1	30.2	104.5
	3	38.0	53.3	60.1	111.2

PEP U(μ, λ)		μ			
		0	1	2	3
λ	0	-	0.3	1.4	**2.3**
	1	0.0	0.3	**1.3**	2.3
	2	0.0	**0.3**	1.4	2.3
	3	**0.2**	0.5	1.6	2.4

Table 2: Runtime of the solvers in microseconds (µs).

Inverting distortion models. For the rational distortion models, (4), that we consider, there will in general not exist any closed form expressions for the inverse. However, in some cases it is necessary to locally find the inverse, e.g. if we have a distortion model and want to undistort the image, or similarly if we have an undistortion model and want to project something into image space. In either case we are given some $\boldsymbol{x}$ and want to find $\boldsymbol{x}'$ such that $h(\|\boldsymbol{x}'\|)\boldsymbol{x}' = \boldsymbol{x}$. Since the distortion is purely radial we can w.l.o.g. assume that $\boldsymbol{x}' = \frac{r}{\|\boldsymbol{x}\|}\boldsymbol{x}$ for some $r \in \mathbb{R}$. This reduces the problem to solving the univariate equation, $h(r)r = \|\boldsymbol{x}\|$, which can be done using iterative methods (starting from $r = \|\boldsymbol{x}\|$).

5. Experimental Evaluation

The methods proposed in Section 4.1 and Section 4.2 work for arbitrary degrees of the rational distortion models. In the experimental evaluation, we focus on five distinct models (*c.f.* Table 3). These models were chosen since they are commonly used in 3D vision applications such as Structure-from-Motion [38]. We note that the undistortion model $\mathtt{U}(0,1)$ corresponds to the state-of-the-art solver from Kukelova et al. [20], where the problem reduces to a linear system. Kukelova et al. [20] can linearly solve also for $\mathtt{U}(0,2)$ and $\mathtt{U}(0,3)$ models, however these models are less common in SfM applications and according to our experiments they do not bring a significant improvement over $\mathtt{U}(0,1)$. For all other models however, there are currently no previous solvers which are able to estimate the distortion parameters jointly with camera pose.

We first evaluate the numerical stability of our solvers through synthetic experiments in Section 5.1. Sections 5.2 and 5.3 then consider two applications of our solvers.

5.1. Synthetic Data Experiments

We evaluate the numerical stability of the proposed solvers on synthetic data. We follow the setup used in [50, 51] to generate synthetic scenes. For each camera model, we select intrinsic and distortion parameters which approximately correspond to a GoPro camera with a wide-field-of-view settings (see Figure 3 for some examples). For each synthetic scene, we use the solvers to estimated the camera poses. We report the error in the translational component[3]. Each solver creates multiple pose hypotheses. We use the hypothesis closest to the ground truth pose when measuring the numerical stability of the solvers.

In the experiment, we compare using a root-bracketing method with Sturm sequences [15] (Section 4.1.1) and converting the problem into a polynomial eigenvalue problem (PEP) (Section 4.1.2). Figure 2 shows the distribution of translation errors for our new solvers, generated from 1000 synthetic scenes each. As can be seen, it is necessary to use the more computationally expensive PEP solver for the more complicated models $\mathtt{D}(3,0)$ and $\mathtt{D}(3,3)$. As can be expected, complex models with more parameters are numerically less stable than models with fewer parameters. Still, all solvers are sufficiently stable.

[3]Since we solve an exact instance, all errors are qualitatively similar.

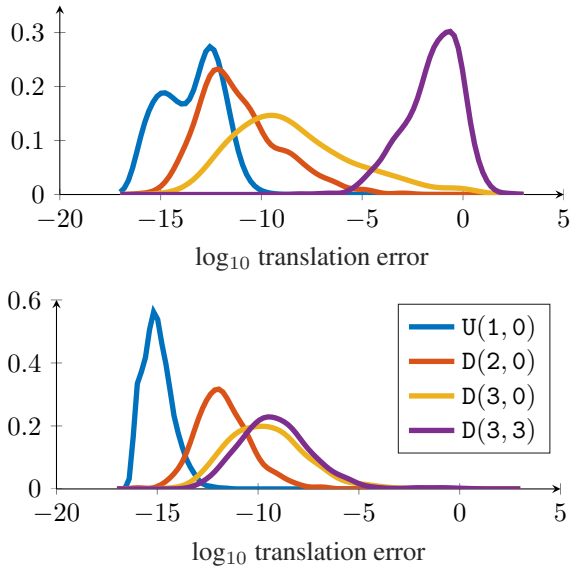

Figure 2: Solver stability on synthetic data. *Top:* Root-bracketing with Sturm chains (Section 4.1.1). *Bottom:* Polynomial eigenvalue formulation (Section 4.1.2).

5.2. Application: Calibration from a Checkerboard

During offline calibration, a calibration target such as a checkerboard pattern is commonly used to obtain an accurate estimate of the camera's intrinsic parameters. The use of the checkerboard results in highly accurate 2D-3D matches between image pixels and points on the pattern. In the following, we evaluate our solvers in such a scenario.

We use a dataset consisting of 32 images captured with a GoPro camera with a wide field-of-view and 26 images captured with the medium field-of-view setting (*c.f.* Figure 3). An initial estimate for the camera poses and calibration was obtained via a calibration toolbox [2]. We then refit a three parameter division model for undistortion and perform bundle adjustment with shared intrinsic/distortion parameters over all images. Compared to the distortion model used in [2] ($\mathtt{D}(3,0)$ with additional tangential terms), this significantly improved the reprojection error. We use the resulting poses and calibration as ground truth.

We found that the $\mathtt{U}(0,3)$ model performed comparably to fitting a $\mathtt{D}(3,3)$ model and chose the former to avoid a bias in favor of our solvers.

Evaluation with ground truth poses. In a first experiment, we use the ground truth poses (except for the translation component t_3 that is estimated by all solvers). This allows us to evaluate the solvers from Section 4.1 and Sec-

	Type	n_μ	n_λ	New	Comment
U(1,0)	Undistortion	1	0	✓	Used in VisualSfM [48, 49]
U(0,1)	Undistortion	0	1		Linear solver. Most common model for minimal solvers, see *e.g.* [6, 16, 23–25, 28, 34].
D(2,0)	Distortion	2	0	✓	Used in Bundler [41], RadialCameraModel and OpenCVCameraModel† in COLMAP [38], PinholeCameraModel in TheiaSfM [42]
D(3,0)	Distortion	3	0	✓	Used in Bouguet's calibration toolbox† [2], PinholeRadialTangentialCameraModel† in TheiaSfM [42]
D(3,3)	Distortion	3	3	✓	OpenCV camera model, FullOpenCVCameraModel† in COLMAP [38]

Table 3: Radial distortion models used for the experiments in Section 5. †: ignoring tangential distortion terms.

Figure 3: Images of a calibration pattern recorded by a GoPro camera in medium and wide field-of-view mode.

tion 4.2 independently of the initial 1D radial pose estimation, which is used by all solvers as detailed in Section 3.2. As shown in [20], the camera extrinsics computed by the 1D radial pose solver are similarly accurate a those obtained by minimal point solvers that jointly estimate the pose and undistortion parameters.

For each image and distortion model from Table 3, we randomly select 1000 minimal samples and estimate the translation t_3, focal length f and distortion parameters from them. Note that the size N of the minimal samples depends on the degree of the model used by the solver and is given as $N = 2 + n_\mu + n_\lambda$. Since the minimal solvers return multiple solutions, we project the corners in the calibration pattern into the image and only keep the solution which has the largest consensus set.

Table 4 shows the results of this experiment. All solvers provide comparable results on this dataset. The new solvers U(1,0), D(2,0) and D(3,0) are slightly less accurate than the existing U(0,1) solver and the new D(3,3) solver.

Evaluation with estimated poses. We repeated the experiment with the but this time using pose estimates from the 1D radial solver as in [20]. As shown in Table 5, the results follow the same trend as those obtained with ground truth poses. One interesting observation is that D(3,0) recovers fewer inliers than the other solvers but still estimates poses and calibrations of similar accuracy as U(0,1) and D(2,0).

Evaluation of the estimated distortion parameters. In a next experiment, we evaluate the quality of the distortion parameters estimated by the different models. We measure quality as reprojection errors in pixels rather than in deviation from ground truth parameters to obtain comparable and interpretable results: We sample a dense equidistant grid of $\sim$ 8000 points on the checkerboard pattern in 3D (*c.f.* Figure 4). These points are then projected into the images using the ground truth camera poses and calibration. This provides us with a set of *ground truth* 2D positions that are used to measure reprojection errors for each estimated (un-)distortion model. Note that the sampled points

	U(0,1)	U(1,0)	D(2,0)	D(3,0)	D(3,3)
Position (mm)	0.42	0.81	0.46	0.44	**0.27**
Focal (%)	**0.21**	0.78	0.36	0.33	0.25
Inliers (%)	**99.8**	94.1	95.4	94.5	98.1
Failure (%)	**0.0**	2.9	0.7	0.7	0.1
Dist. inliers (%)	**99.5**	90.6	91.2	90.0	95.5

Table 4: Median error in position and focal length using ground truth camera poses (except for t_3). Inliers correspond to points with a reprojection error of less than 10px. *Failure* shows the percentage of instances where the solver failed to return a valid solution, *e.g.*, when there were no real roots or not all points were in front of the camera.

	U(0,1)	U(1,0)	D(2,0)	D(3,0)	D(3,3)
Position (mm)	1.73	2.23	2.05	2.13	**1.49**
Focal (%)	0.82	1.55	1.21	1.31	**0.80**
Inliers (%)	**90.5**	85.4	83.4	80.8	87.4
Failure (%)	0.9	3.3	1.0	0.5	**0.4**
Dist. inliers (%)	**88.6**	81.3	77.8	74.5	83.0

Table 5: Median error in position and focal length using estimated camera poses. See the caption of Table 4 for details.

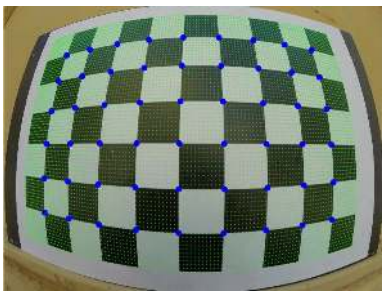

Figure 4: Densely sampled grid (green) used to evaluate the quality of the distortion models. Only corner detections (blue) are used for estimation.

extend outside of the corners used for estimation. For this experiment, we evaluate all solvers using both ground truth and estimated camera poses.

Figure 5 shows the cumulative histogram of over the median reprojection error in pixels, and the average number of inliers (< 10 px) for the grid points is shown at the bottom of Table 4 and Table 5.

5.3. Application: Joint Localization & Calibration

In some applications, *e.g.*, visual localization in an AR cloud providing services to smart phones [33], obtaining an

	Rig Camera 4						Rig Camera 6					
	[28]	[20]	U(1,0)	D(2,0)	D(3,0)	D(3,3)	[28]	[20]	U(1,0)	D(2,0)	D(3,0)	D(3,3)
Rotation (deg.)	3.47	**3.46**	3.47	**3.46**	3.47	3.47	2.65	2.65	**2.64**	**2.64**	2.65	**2.64**
Position (m.)	**0.02**	**0.02**	**0.02**	**0.02**	**0.02**	**0.02**	**0.05**	**0.05**	**0.05**	**0.05**	**0.05**	**0.05**
Focal (%)	**0.60**	0.61	1.79	0.90	0.85	2.14	1.63	1.60	1.74	1.37	**1.28**	1.76
Inliers (%)	**95.0**	**95.0**	93.7	94.7	94.8	**95.0**	93.9	94.1	94.0	94.6	94.3	**94.7**
Dist. inliers (%)	90.7	**91.0**	86.7	87.5	88.4	88.1	79.8	81.2	80.1	81.7	**81.9**	81.6

Table 6: *ETH3D Playground*: Average errors obtained after RANSAC. Both [28] and [20] use U(0,1).

	Rig Camera 4						Rig Camera 6					
	[28]	[20]	U(1,0)	D(2,0)	D(3,0)	D(3,3)	[28]	[20]	U(1,0)	D(2,0)	D(3,0)	D(3,3)
Rotation (deg.)	4.81	4.80	**4.75**	4.79	4.78	4.78	2.76	2.72	**2.71**	3.44	2.72	2.73
Position (m.)	0.18	0.17	0.23	**0.12**	**0.12**	**0.12**	0.32	0.26	0.27	0.25	**0.23**	**0.23**
Focal (%)	1.51	1.42	2.48	0.97	**0.95**	1.45	3.82	3.15	3.16	3.33	**2.29**	2.31
Inliers (%)	95.0	95.1	94.5	95.2	94.9	**95.5**	92.6	93.9	93.8	94.3	94.4	**94.5**
Dist. inliers (%)	70.2	71.7	64.4	**79.5**	78.7	77.2	52.9	50.9	50.6	**54.8**	53.3	54.7

Table 7: *ETH3D Delivery area*: Average errors obtained after RANSAC. Both [28] and [20] use U(0,1).

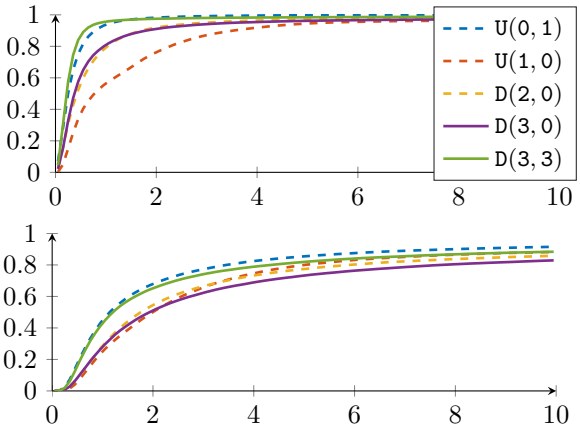

Figure 5: Cumulative histogram over the median reprojection errors (in pixels) for the densely sampled grid points (*c.f.* Figure 4). *Top:* Using ground truth poses (except for t_3). *Bottom:* Using estimated camera poses (Section 3.2).

accurate intrinsic calibration in an offline stage is not possible. Rather, such approaches need to jointly estimate the pose of the camera together with its intrinsic calibration.

In our final experiment, we consider such an application scenario. We use two different images sequences from the ETH3D dataset [40], *playground* (2x240 images) and *delivery area* (2x237 images). Using the provided ground truth poses and calibrations, we constructed a sparse 3D reconstruction using the DSLR images. We then matched features extracted from the lower resolution multi-camera rig images against features in the DSLR images, resulting in a set of 2D-3D correspondences with the sparse model. The multi-camera rig consists of two stereo pairs. Since each pair has two identical cameras we only consider one camera from each pair. To perform the experiment, we then estimated the pose and distortion parameters for each model in Table 3 using LO-RANSAC [30]. The inlier threshold was set to 5px and we ran 100 RANSAC iterations per image.

To evaluate the accuracy of the estimated calibration, we sample a dense grid in the image, which we undistort with the ground truth distortion model, and then re-distort using the estimated distortion model. For the undistortion solvers (U(1,0) and U(0,1)), we do the opposite, *i.e.*, undistort with the estimated model and then apply re-distort with the ground truth model. The results can be seen in Table 6 and Table 7. We also include the results from the minimal U(0,1) solver from [28], which requires four 2D-3D matches. On these two datasets, the proposed solvers have comparable performance to the state-of-the-art undistortion solvers. The comparison with [28] shows that using a two-stage approach instead of simultaneously estimating all parameters does not degrade performance.

6. Conclusion

Following a two-stage approach inspired by [20], we have shown that it is possible to create minimal solvers for 6DOF camera pose estimation under radial distortion that use rational (un)distortion models of an arbitrary degree. In particular, this paper presented the first minimal solvers based on a distortion rather than undistortion model. We have shown that our new solvers achieve a similar accuracy as existing undistortion-based methods. As such, our work has closed an important gap in the literature as applications such as Structure-from-Motion or Augmented Reality predominantly use distortion models. In general, our work provides additional options for practitioners who might have their own motivations for using one model over another.

Acknowledgements. Viktor Larsson was supported by the ETH Zurich Postdoctoral Fellowship program and the Marie Sklodowska-Curie Actions COFUND program. Zuzana Kukelova was supported by the ESI Fund, OP RDE programme under the project International Mobility of Researchers MSCA-IF at CTU No. CZ.02.2.69/0.0/0.0/17_050/0008025.

References

[1] Cenek Albl, Zuzana Kukelova, Viktor Larsson, and Tomas Pajdla. Rolling shutter camera absolute pose. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 2019. 2

[2] Jean-Yves Bouguet. Camera calibration toolbox for matlab (2008). *URL http://www. vision. caltech. edu/bouguetj/calib_doc*, 1080, 2008. 1, 2, 6, 7

[3] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000. 2

[4] José Henrique Brito, Roland Angst, Kevin Köser, and Marc Pollefeys. Radial distortion self-calibration. *Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[5] Duane C Brown. Decentering distortion of lenses. *Photogrammetric Engineering and Remote Sensing*, 1966. 2

[6] Martin Bujnak, Zuzana Kukelova, and Tomas Pajdla. New efficient solution to the absolute pose problem for camera with unknown focal length and radial distortion. In *Asian Conference on Computer Vision (ACCV)*, 2010. 1, 2, 3, 7

[7] Federico Camposeco, Torsten Sattler, and Marc Pollefeys. Non-parametric structure-based calibration of radially symmetric cameras. In *International Conference on Computer Vision (ICCV)*, 2015. 2, 3

[8] David Claus and Andrew W. Fitzgibbon. A rational function lens distortion model for general cameras. In *Computer Vision and Pattern Recognition (CVPR)*, pages 213–219, June 2005. 2

[9] Alexander Eugen Conrady. Decentred lens-systems. *Monthly notices of the royal astronomical society*, 79(5):384–390, 1919. 2

[10] David A Cox, John Little, and Donal O'shea. *Using algebraic geometry*, volume 185. Springer Science & Business Media, 2006. 4, 5

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

[12] Andrew W Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Computer Vision and Pattern Recognition (CVPR)*, 2001. 1, 2

[13] Robert M. Haralick, Chung-nan Lee, Karsten Ottenberg, and Michael Nölle. Review and analysis of solutions of the three point perspective pose estimation problem. *International Journal of Computer Vision (IJCV)*, 13(3):331–356, 1994. 1, 2

[14] Jan Heller, Didier Henrion, and Tomáš Pajdla. Stable radial distortion calibration by polynomial matrix inequalities programming. In *Asian Conference on Computer Vision (ACCV)*, 2014. 5

[15] David G. Hook and Ross McAree. *Using Sturm Sequences to Bracket Real Roots of Polynomial Equations*. Academic Press, Jan. 1990. 4, 6

[16] Klas Josephson and Martin Byröd. Pose estimation with radial distortion and unknown focal length. In *Computer Vision and Pattern Recognition (CVPR)*, 2009. 1, 2, 3, 7

[17] Laurent Kneip, Davide Scaramuzza, and Roland Siegwart. A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation. In *Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

[18] Zuzana Kukelova, Cenek Albl, Akihiro Sugimoto, and Tomas Pajdla. Linear solution to the minimal absolute pose rolling shutter problem. In *Asian Conference on Computer Vision (ACCV)*, 2018. 2

[19] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Polynomial eigenvalue solutions to minimal problems in computer vision. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 34(7):1381–1393, 2012. 4

[20] Zuzana Kukelova, Martin Bujnak, and Tomas Pajdla. Real-time solution to the absolute pose problem with unknown radial distortion and focal length. In *International Conference on Computer Vision (ICCV)*, 2013. 1, 2, 3, 4, 5, 6, 7, 8

[21] Zuzana Kukelova, Jan Heller, Martin Bujnak, Andrew W. Fitzgibbon, and Tomás Pajdla. Efficient solution to the epipolar geometry for radially distorted cameras. In *International Conference on Computer Vision (ICCV)*, pages 2309–2317, 2015. 1, 2

[22] Zuzana Kukelova, Jan Heller, Martin Bujnak, and Tomás Pajdla. Radial distortion homography. *Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 2

[23] Zuzana Kukelova, Jan Heller, and Andrew Fitzgibbon. Efficient intersection of three quadrics and applications in computer vision. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 5, 7

[24] Zuzana Kukelova and Viktor Larsson. Radial distortion triangulation. In *Computer Vision and Pattern Recognition (CVPR)*, 2019. 1, 2, 7

[25] Zuzana Kukelova and Tomas Pajdla. A minimal solution to radial distortion autocalibration. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 33(12):2410–2422, 2011. 1, 2, 7

[26] Viktor Larsson and Kalle Åström. Uncovering symmetries in polynomial systems. In *European Conference on Computer Vision (ECCV)*. Springer, 2016. 2

[27] Viktor Larsson, Kalle Åström, and Magnus Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[28] Viktor Larsson, Zuzana Kukelova, and Yinqiang Zheng. Making minimal solvers for absolute pose estimation compact and robust. In *International Conference on Computer Vision (ICCV)*, 2017. 1, 2, 3, 7, 8

[29] Viktor Larsson, Zuzana Kukelova, and Yinqiang Zheng. Camera pose estimation with unknown principal point. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 3

[30] Karel Lebeda, Jiří Matas, and Ondřej Chum. Fixing the Locally Optimized RANSAC. In *British Machine Vision Conference (BMVC)*, 2019. 8

[31] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal of Computer Vision (IJCV)*, 81(2):155–166, 2009. 2

[32] Lili Ma, YangQuan Chen, and Kevin L. Moore. Rational radial distortion models of camera lenses with analytical so-

lution for distortion correction. *I. J. Information Acquisition*, 1:135–147, 06 2004. 2

[33] Sven Middelberg, Torsten Sattler, Ole Untzelmann, and Leif Kobbelt. Scalable 6-DOF Localization on Mobile Devices. In *European Conference on Computer Vision (ECCV)*, 2014. 7

[34] Gaku Nakano. A versatile approach for solving pnp, pnpf, and pnpfr problems. In *European Conference on Computer Vision (ECCV)*, 2016. 1, 3, 7

[35] James Pritts, Zuzana Kukelova, Viktor Larsson, and Ondřej Chum. Radially-distorted conjugate translations. In *Computer Vision and Pattern Recognition (CVPR)*, 2018. 1, 2

[36] James Pritts, Zuzana Kukelova, Viktor Larsson, and Ondřej Chum. Rectification from radially-distorted scales. In *Asian Conference on Computer Vision (ACCV)*, 2018. 1, 2

[37] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 39(9):1744–1756, 2017. 1

[38] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 6, 7

[39] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise View Selection for Unstructured Multi-View Stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 1

[40] Thomas Schöps, Johannes L. Schönberger, Silvano Galiani, T. Sattler, Konrad Schindler, Marc Pollefeys, and Andreas Geiger. A Multi-View Stereo Benchmark with High-Resolution Images and Multi-Camera Videos. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 8

[41] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. In *ACM Transactions on Graphics*, 2006. 1, 7

[42] Chris Sweeney. Theia multiview geometry library: Tutorial & reference. http://theia-sfm.org. 7

[43] Zhongwei Tang, Rafael Grompone von Gioi, Pascal Monasse, and Jean-Michel Morel. A precision analysis of camera distortion models. *IEEE Transactions on Image Processing*, 26(6):2694–2704, June 2017. 2

[44] SriRam Thirthala and Marc Pollefeys. Radial multi-focal tensors. *International Journal of Computer Vision (IJCV)*, 96(2):195–211, 2012. 2, 3

[45] Bill Triggs. Camera pose and calibration from 4 or 5 known 3d points. In *International Conference on Computer Vision (ICCV)*, 1999. 2

[46] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle Adjustment - A Modern Synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, 2000. 1

[47] Roger Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987. 2, 3

[48] Changchang Wu. VisualSFM: A Visual Structure from Motion System. http://ccwu.me/vsfm/, 2011. 7

[49] Changchang Wu. Towards Linear-Time Incremental Structure from Motion. In *International Conference on 3D Vision (3DV)*, 2013. 7

[50] Changchang Wu. P3.5p: Pose estimation with unknown focal length. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 6

[51] Yinqiang Zheng, Shigeki Sugimoto, Imari Sato, and Masatoshi Okutomi. A general and simple method for camera pose and focal length determination. In *Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 6