

Parameter-Free Radial Distortion Correction with Center of Distortion Estimation

Richard Hartley and Sing Bing Kang

Abstract—We propose a method of simultaneously calibrating the radial distortion function of a camera and the other internal calibration parameters. The method relies on the use of a planar (or, alternatively, nonplanar) calibration grid which is captured in several images. In this way, the determination of the radial distortion is an easy add-on to the popular calibration method proposed by Zhang [24]. The method is entirely noniterative and, hence, is extremely rapid and immune to the problem of local minima. Our method determines the radial distortion in a parameter-free way, not relying on any particular radial distortion model. This makes it applicable to a large range of cameras from narrow-angle to fish-eye lenses. The method also computes the center of radial distortion, which, we argue, is important in obtaining optimal results. Experiments show that this point may be significantly displaced from the center of the image or the principal point of the camera.

Index Terms—Radial distortion, camera calibration, fundamental matrix.



1 INTRODUCTION

RADIAL distortion is a significant problem in the analysis of digital images. Although this problem was widely studied by photogrammetrists striving for extreme accuracy, it has been largely ignored in the extensive literature on structure and motion of the past decade or so. (Less than five pages are devoted to this topic in [12].) Almost exclusively, methods have involved iterative techniques such as bundle adjustment.

At the same time, several different camera models have been proposed for different types of cameras. The most popular radial distortion model is the even-order polynomial model that models radial distortion as scaling by a factor $1 + \kappa_1 r^2 + \kappa_2 r^4 + \dots$.

Different papers suggest the alternatives of multiplying or dividing by this factor. The idea of using only even order terms is questionable at best and, in any case, such a model does not hold for fish-eye lenses, where the distortion becomes infinite toward the periphery and, hence, cannot be accurately modeled by a polynomial. Iterative optimization techniques can (in theory) handle any parameterized model, but issues of convergence, initialization, and overfitting can be troublesome. Parametrized distortion curves may not extrapolate well. In addition, by building the model into the bundle-adjustment problem, one cannot take advantage of more sophisticated techniques of polynomial or function approximation (see [1]).

In this paper, we prefer to ignore the issue of choosing any particular radial distortion model by adopting a model-free approach. The only assumption we make on the radial

distortion function is that it is monotonic. Despite this, it is possible to determine the radial distortion curve. This way, the problem of fitting a parameterized function model (if required) is separated out from the estimation of the distortion curve and makes our method applicable to all (or at least most) lenses.

Iterative optimization methods can be troublesome due to lack of convergence, choosing an initial estimate, and determining a stopping criterion. The advantage of our algorithm is that it is entirely noniterative. This makes it fast and immune to these problems of iterative techniques. Sometimes, the cost of using a noniterative technique is that it can minimize some arbitrary cost function unrelated to the noise model. Thus, they can be very sensitive to noise. On the other hand, the linear models that we use are closely associated with the optimal model. Although iterative refinement gives some improvement, it is minimal.

Another feature of our method that distinguishes it from most of the computer-vision approaches to radial distortion is that we also compute the center of distortion.

It is common in the literature to assume that the center of distortion, e , is known, usually at the center of the image. This is not a safe assumption in general. The center of distortion can be displaced from the image center by many factors, such as offset of the lens center from the CCD center, slight tilt of the sensor plane with respect to the lens, misalignment of the individual components of a compound lens, and cropping of the image. In cheap consumer cameras, available at a cost of a few hundred dollars, it should not be assumed that the optics of the camera are accurate since these effects make little difference to subjective image quality.

The importance of determining the center of distortion has long been recognized in the photogrammetry community. As enunciated by Clarke et al. [6]: "attention must be paid to even small details, such as the location of the principal point,¹ if residuals are to be optimally minimized." They further state: "There will be an optimum location of

1. Clarke et al. use the term principal point to mean the center of distortion.

• R. Hartley is with the Department of Information Engineering, Australian National University and National ICT Australia, ACT 0200, Australia. E-mail: Richard.Hartley@anu.edu.au.

• S.B. Kang is with Microsoft Research, Microsoft Corporation, One Microsoft Way, Redmond, WA 98052. E-mail: sbkang@microsoft.com.

Manuscript received 2 Feb. 2006; revised 14 July 2006; accepted 28 Sept. 2006; published online 18 Jan. 2007.

Recommended for acceptance by K. Daniilidis.

For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0115-0206. Digital Object Identifier no. 10.1109/TPAMI.2007.1147.

this point as a result of correctly modelling the distortion. It will not be possible to correct for [errors due to incorrect estimation of the center of distortion] by using the decentering distortion parameters to compensate [...]. Furthermore, it will not be possible to correct for this error using the exterior orientation parameters [...]."

By experimentation, we show that the usual assumption that the center of distortion is at the center of the image is not valid.

The disadvantage of our method is that it requires a simple calibration grid. We briefly explore the extension of our method to autocalibration methods that do not use a calibration grid. This is, indeed, possible and works with exact measurements, but our observation is that it is extremely sensitive to noise.

2 PRIOR WORK

The work on radial distortion removal is extensive. Photogrammetry methods usually rely on known calibration points or structures (for example, [3], [4], [7], [21]). Most often cited is the plumb line technique of Brown [4], which carries out radial distortion correction by straightening the images of straight lines in the world. Tsai [21] uses corners of regularly spaced boxes of known dimensions for full camera calibration, including radial distortion, whereas Faig [7] requires that the points used be coplanar. Wei and Ma [22] use projective invariants to recover radial distortion coefficients.

Becker and Bove [2] map straight lines onto a unit sphere and find both radial and decentering coefficients that minimize the vanishing point dispersion. (The user has to manually group parallel lines together—each group should have a unique vanishing point.) Swaminathan and Nayar [17] proposed a user-guided self-calibration approach. The distortion parameters are computed from user-selected points along projections of straight lines in the image. Stein [16] describes a more flexible approach, requiring only point correspondences between multiple views. He uses epipolar and trilinear constraints and searches for radial distortion parameters that minimize the errors in these constraints.

Interestingly, there is much less work published on fish-eye lens calibration. Most approaches use an ideal projection model (for example, [5]) or use the distortion model meant for rectilinear lenses by adding more nonlinear terms (for example, [15]). Xiong and Turkowski [23] use a cubic polynomial to represent the mapping between the latitude angle and the polar distance for fish-eye lenses. They use multiple overlapping images to extract these coefficients (minimizing the brightness difference in the overlap areas). Strictly speaking, fish-eye lenses are noncentral, that is, they do not have a single point of projection. However, the deviation of light rays from an effective single point of projection is slight (see, for example, [14]) and we make the central assumption for fish-eye lenses in our work. In addition, because of the wide field of view of fish-eye lenses, we characterize radial distortion in terms of angles rather than pixels.

There has recently been renewed interest in estimating radial distortion, based on techniques of projective geometry. By incorporating *one* lens distortion parameter into the epipolar constraint involving the fundamental matrix, Fitzgibbon [8] simultaneously solves for both. He casts the problem as a quadratic-eigenvalue problem, which can then be easily solved using a numerical library. This technique was

later generalized to omnidirectional cameras [13]. Independently of our work, Thirithala and Pollefeys [20] proposed a linear technique to recover radial distortion using multifocal tensors. They assume the center of distortion is known (at the image center) and use the observation that the vectors between the distorted and undistorted points pass through a common point to reduce the problem to estimating a 2D trifocal tensor. They extended this work to nonplanar scenes using a quadrifocal tensor in [19]. Most recently, Tardif and Sturm [18] have considered self-calibration of a general circularly symmetric calibration model, including the estimation of radial distortion using geometric techniques.

3 OVERVIEW

Our main technique for radial distortion correction involves the use of a calibration pattern on which are marked points with known locations. The pattern may be planar or 3D, though, in practical applications, a planar calibration pattern is perhaps simpler to use and works just as well. Points on a planar calibration pattern would ideally be mapped to undistorted points in the image under perspective camera geometry by a 2D homography. In reality, in the presence of radial distortion, we do not observe these ideal or undistorted points. Nevertheless, the essence of radial distortion calibration is to compute this ideal homography, thereby correcting the radial distortion.

A sequence of images of the calibration pattern are taken in different orientations. The method can work with a single image, though the best results are obtained with several images (10 images would be a reasonable number). This acquisition of calibration images is similar to the way it is done in the popular method of Zhang [24] for camera calibration. Once the images are acquired, the next step is to identify and find the coordinates of the known calibration points in the images. It is not necessary to find all such points. To simplify this procedure, we prefer to use a calibration grid with a checkerboard pattern since it is relatively easy to extract the vertices automatically from the images. However, any other pattern may be used.

Our procedure for computing the radial distortion of the camera is based on two main ideas. The first idea is to exploit the analogy between the radial displacement of points under radial distortion and the motion of points away from a focus of expansion when a camera moves toward a scene. By using techniques originally applied to two-view structure and motion (specifically, the fundamental matrix), we are able to compute the center of radial distortion and also, up to three remaining degrees of ambiguity, the homographies mapping the points on the calibration grid to their corresponding *undistorted* or ideal points in the image.

The second idea is that the remaining ambiguity in these homographies may be resolved by enforcing two simple constraints: that the distortion is circularly symmetric, meaning that the amount of distortion is dependent only on a point's distance from the center of distortion, and that the correspondence between the undistorted radius of a point and its distorted radius is a monotonic function—in other words, the order of points in terms of their distance from the center of distortion is not changed by the radial distortion. With these two simple assumptions, it is possible to remove the ambiguity and compute the imaging homographies

uniquely. In fact, with a slight simplification, this may be done by a linear algorithm.

Once the image homographies have been computed, we are in possession of both undistorted and distorted points. If many images are used, then we have a large number of such pairs. A scatter plot of the undistorted radius versus the distorted radius shows the form of the calibration curve. This curve may then be used to correct any other images captured with the same camera.

The procedure outlined computes the homographies relating the calibration grid and the image. This allows us to now use the calibration procedure of Zhang [24] for calibration of a projective (pinhole) camera. At least three images are required for this. Thus, both the radial distortion and the internal camera calibration parameters are computed from the same set of calibration images.

Using a 3D calibration pattern. Much the same procedure applies in the case where a 3D calibration pattern is used. Instead of image homographies, we are concerned with pinhole camera projection. The role of the fundamental matrix in the above description is taken by an analogous 3×4 matrix with similar properties.

Methods not using a calibration grid. The use of the calibration grid allows for the greatest degree of accuracy. However, in some instances, it might be convenient to do without a calibration grid at all. This is a natural extension, analogous to the way ordinary calibration of a camera may be performed using a calibration grid or, alternatively, by autocalibration using only correspondences between multiple images. Some autocalibration methods involve the use of multiview tensors [12] that relate corresponding points in several views. Similarly, our methods for calibration-grid-free radial distortion correction involve the use of multiview tensors.

We briefly describe methods for computing radial distortion using images of a planar or nonplanar scene using multiview tensors. These involve the use of a trifocal $3 \times 3 \times 3$ tensor for the planar scene case or a similar quadrifocal tensor for the case of a general scene. In these methods, we also, as before, compute the center of distortion of the camera. Although these methods are correct in theory, it has been our observation that they are sensitive even to extremely small amounts of noise and are not suited for practical use. For this reason, we do not carry out extensive testing.

The situation is much more favorable if we assume that the center of distortion is known. In this case, the tensors involved become much smaller. In the planar scene case, instead of a trifocal tensors of size $3 \times 3 \times 3$, a tensor of size $2 \times 2 \times 2$ may be used. For the general case, the quadrifocal tensor decreases from size $3 \times 3 \times 3 \times 3$ to $2 \times 2 \times 2 \times 2$. Radial distortion correction with known center of distortion using these tensors has been investigated thoroughly in independent work by Thirithala and Pollefeys [20], [19] and, for that reason, will not be considered further in this paper.

The range of techniques covered in this paper is shown in Fig. 1. The emphasis is on techniques associated with the use of a planar calibration grid. The method is simple, robust, has a linear solution, and is an easy add-on to the popular calibration method of Zhang [24]. The other methods are included for completeness, but are not likely to be as useful as this preferred method.

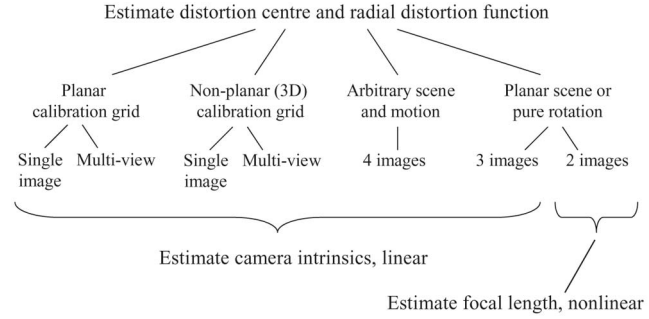


Fig. 1. Overview of techniques for radial distortion correction.

4 THE RADIAL DISTORTION MODEL

In our model for radial distortion correction, we ignore decentering distortion, which is commonly due to lack of alignment of different lens elements. This can lead to nonradial (tangential) components of lens distortion. We assume that distortion is radial. The imaging process therefore is made up of several steps:

1. **Projection.** Points are projected onto the image plane by the ideal camera via the mapping

$$\tilde{\mathbf{x}}^u = [\mathbf{I} | \mathbf{0}] \mathbf{X},$$

where \mathbf{X} is expressed in the camera coordinate frame.

2. **Radial distortion.** The distorted point $\tilde{\mathbf{x}}^d$ is given by

$$\tilde{\mathbf{x}}^d = \tilde{\mathbf{e}} + \lambda(\tilde{\mathbf{x}}^u - \tilde{\mathbf{e}}),$$

where λ represents the distortion ratio and $\tilde{\mathbf{e}}$ is the *center of distortion*. The distortion ratio λ depends on the point $\tilde{\mathbf{x}}^u$ and might perhaps more properly be written as $\lambda(\tilde{\mathbf{x}}^u)$.

3. **Pixel sampling.** The details of the pixel coordinate system are encapsulated in the calibration matrix \mathbf{K} . The point $\tilde{\mathbf{x}}^d$ is mapped to the pixel image point given by

$$\mathbf{x}^d = \mathbf{K} \tilde{\mathbf{x}}^d.$$

In this formulation, the tilde in $\tilde{\mathbf{x}}^u$, $\tilde{\mathbf{x}}^d$, and $\tilde{\mathbf{e}}$ indicates that it is measured in geometric focal length units. We may also define $\mathbf{x}^u = \mathbf{K} \tilde{\mathbf{x}}^u$ and $\mathbf{e} = \mathbf{K} \tilde{\mathbf{e}}$ to get the undistorted point and center of distortion in pixel coordinates. Since \mathbf{K} represents an affine transformation of coordinates and affine transformation preserve length ratios along straight lines, we see that the pixel coordinates are related by

$$\mathbf{x}^d = \mathbf{e} + \lambda(\mathbf{x}^u - \mathbf{e}),$$

with the same λ as before.

Ambiguity. There is an ambiguity between the degree of radial distortion, represented by λ , and the magnification factor (focal length parameter) of the calibration matrix \mathbf{K} . In particular, if λ is replaced by $\alpha\lambda$ (where α is constant for all points) and \mathbf{K} is multiplied on the right by the diagonal matrix $\text{diag}(\alpha, \alpha, 1)$, then the resulting mapping from world to pixel coordinates in the image is unchanged. A reasonable and usual way of resolving this ambiguity is to choose the radial distortion ratio $\lambda(\tilde{\mathbf{x}}^u)$ such that, to first order, it is

equal to unity near the center of distortion. Thus, to first order near the center of distortion, we may write $\mathbf{x}^d = \mathbf{x}^u$.

In this paper, we will not always enforce this condition during the computations, but the estimated radial distortion function λ and calibration matrix K can be corrected at the end to conform to this convention.

Radial symmetry. Since lenses are generally ground to be circularly symmetric, without decentering distortion, it may be assumed that the distortion factor λ is a function only of the radius $\tilde{r} = \|\tilde{\mathbf{x}}^u - \tilde{\mathbf{e}}\|$. It is often assumed that it may be expressed as a function of the “pixel radius,” $r = \|\mathbf{x}^u - \mathbf{e}\|$. However, this is not the same thing unless the image has square pixels, in which case K represents a scaled Euclidean coordinate transform. At one point in the description of the algorithm, it will be assumed that the radial distortion factor λ is circularly symmetric. Later on, it will be pointed out that there is a practical way in which nonsquare pixels can be dealt with effectively.

5 DISTORTION ESTIMATION USING A PLANAR PATTERN

We now proceed to describe our method for estimating the radial distortion curve.

5.1 Finding the Distortion Center

Throughout this paper, our major assumption relating to radial distortion is that a point in an image is moved radially from its undistorted point \mathbf{x}^u to its distorted point \mathbf{x}^d . Thus, the distortion is referenced to a center of radial distortion \mathbf{e} according to the relationship

$$\mathbf{x}^d = \mathbf{e} + \lambda(\mathbf{x}^u - \mathbf{e}),$$

where

$$\mathbf{x}^u = K[\mathbf{I} \mid \mathbf{0}] \mathbf{E} \mathbf{X} = \mathbf{P} \mathbf{X},$$

with \mathbf{E} being a 3D Euclidean coordinate transformation that maps the point \mathbf{X} into the camera coordinate system.

Our simplest method for estimating the center of radial distortion involves the use of a calibration rig consisting of a plane with several distinguishable points. The positions of the points are assumed known in an Euclidean coordinate frame on the plane. A suitable such rig would be a checkerboard pattern, with the vertices of the squares forming our set of points; it is not necessary to recognize a distinguished vertex since any vertex can serve as the coordinate origin.

We give an intuitive description of our idea before writing the mathematical formalities. The pattern of known points is projected into the image by an ideal nondistorted camera, and then the points are each moved from their “initial” to their “final” position by expansion away from (or toward) a center of distortion. We can compare this with the motion of points seen by a camera moving forward toward a scene. In this case, the points also undergo a radial expansion and the center of expansion is known as the center of expansion or, more generally, the epipole. It is well known how to find the epipole—we compute the fundamental matrix [12]. The situation is entirely analogous here and we can compute the center of radial distortion by computing the fundamental matrix relating the known coordinates of points on our calibration pattern and the measured positions of the points in the distorted image. The

center of radial distortion is simply the epipole computed from this fundamental matrix.

Let \mathbf{x}_i^c be points with known coordinates on a planar calibration grid and \mathbf{x}_i^d be the corresponding points in the distorted image. The calibration pattern points \mathbf{x}_i^c and the undistorted image points \mathbf{x}_i^u (in pixel coordinates) are related by a homography \mathbf{H} according to $\mathbf{x}_i^u = \mathbf{H}\mathbf{x}_i^c$. Note that the superscripts u, d , and c are used to distinguish the *undistorted*, *distorted*, and *calibration* points and the subscript i runs over all points.

The points \mathbf{x}_i^u are next distorted radially away from the center of distortion \mathbf{e} to give

$$\mathbf{x}_i^d = \mathbf{e} + \lambda_i(\mathbf{x}_i^u - \mathbf{e}).$$

Note that the distortion factor λ_i is typically different for each point. We multiply this expression on the left by $[\mathbf{e}]_\times$ (the skew-symmetric 3×3 matrix representing the cross product), resulting in $[\mathbf{e}]_\times \mathbf{x}_i^d = \lambda_i [\mathbf{e}]_\times \mathbf{x}_i^u$, where the terms \mathbf{e} disappear when multiplied by $[\mathbf{e}]_\times$. However, since $\mathbf{x}_i^u = \mathbf{H}\mathbf{x}_i^c$, we have

$$[\mathbf{e}]_\times \mathbf{x}_i^d = \lambda_i [\mathbf{e}]_\times \mathbf{H}\mathbf{x}_i^c.$$

Finally, multiplying on the left by \mathbf{x}_i^{dT} and observing that $\mathbf{x}_i^{dT} [\mathbf{e}]_\times \mathbf{x}_i^d = 0$, because $[\mathbf{e}]_\times$ is skew symmetric, we obtain

$$0 = \lambda_i \mathbf{x}_i^{dT} ([\mathbf{e}]_\times \mathbf{H}) \mathbf{x}_i^c.$$

Writing $\mathbf{F} = [\mathbf{e}]_\times \mathbf{H}$, we have the usual fundamental matrix relation $\mathbf{x}_i^{dT} \mathbf{F} \mathbf{x}_i^c = 0$. The matrix \mathbf{F} may be called the *fundamental matrix for radial distortion*. \mathbf{F} may be computed in the usual way [12] from several point correspondences and the center of radial distortion can be extracted as the left epipole.

In the case where there is no radial distortion at all, the above computation of the fundamental matrix is unstable and the estimated value of \mathbf{e} is essentially arbitrary and meaningless. If there is no radial distortion, then it does not make much sense to talk about a center of distortion. Without radial distortion, the distortion factor λ_i will equal unity for each point and the distortion equation $\mathbf{x}_i^d = \mathbf{e} + \lambda_i(\mathbf{x}_i^u - \mathbf{e})$ reduces to $\mathbf{x}_i^d = \mathbf{x}_i^u$, independent of \mathbf{e} . This degenerate situation is easily detected during the computation of the fundamental matrix. In such a case, there will exist a two-parameter family of possible fundamental matrices consistent with the data [12]. An example of this situation is given in Section 6.

5.2 Multiview Methods

Instead of using a single image for computation of the distortion center, we may consider how to take advantage of several views of a calibration grid. We consider all of the points in all images indexed by a single index i and denote by $k(i)$ the image that the i th point belongs to. The fundamental equation then becomes

$$\mathbf{x}_i^{dT} ([\mathbf{e}]_\times \mathbf{H}_{k(i)}) \mathbf{x}_i^c = \mathbf{x}_i^{dT} \mathbf{F}_{k(i)} \mathbf{x}_i^c = 0.$$

From this, one can compute each of the matrices \mathbf{F}_k individually.

Next, one wishes to find the vector \mathbf{e} as the simultaneous left null-space generator of all the \mathbf{F}_k . Since the matrices \mathbf{F}_k are computed separately, there will not be a single vector satisfying $\mathbf{e}^T \mathbf{F}_k = 0$ for all k . Instead, we find the least-squares solution to the set of equations $\mathbf{e}^T \mathbf{F}_k = 0$, where this is solved

using all F_k at once. For each value of k indexing all of the available images, we obtain a set of three equations of the form $\mathbf{e}^\top \mathbf{f}_{kj} = 0$. Here, $j = 1, \dots, 3$ and \mathbf{f}_{kj} is the j th column of F_k . The least-squares solution to the set of all such equations gives an estimate of the center of distortion.

5.3 Mapping Calibration Grid to Image Plane

Next, we wish to find the homographies H_k mapping the calibration grid to the image plane. We can solve for the homography matrix H_k by factoring the fundamental matrix as $F_k = [\mathbf{e}]_\times H_k$. Note, however, that this factorization is not unique, because $[\mathbf{e}]_\times$ is singular. In addition, an exact factorization is not possible unless the vector \mathbf{e} is in the left null space of F_k .

The result of the computation of \mathbf{e} in the previous section will be a vector \mathbf{e} that does not exactly satisfy $\mathbf{e}^\top F_k$ for any k . It is possible to correct each F_k by projecting onto the subspace perpendicular to \mathbf{e}^\top . However, this a posteriori correction of each of the matrices F_k is not ideal. A preferable method is to repeat the computation of each F_k with known epipole. In doing this, it is convenient to change the image coordinates first so that \mathbf{e} is the coordinate origin $(0, 0, 1)^\top$ (in homogeneous coordinates). We carry out all subsequent computations in this coordinate system and make an appropriate correction at the end.

Thus, we may compute F_k and H_k directly by solving the equations

$$\mathbf{x}_i^{d\top} F_k \mathbf{x}_i^c = \mathbf{x}_i^{d\top} ([\mathbf{e}]_\times H_k) \mathbf{x}_i^c = 0 \quad (1)$$

individually for each image k using points belonging to that image. Note that, if $\mathbf{e} = (0, 0, 1)^\top$, then the final row of F_k is zero, so we only solve for the six other entries of F_k . Then, the homography H_k can be written down directly from F_k . Namely, $H_k = [\mathbf{f}_2^\top; -\mathbf{f}_1^\top; 0]$, where the semicolon means that we stack the rows \mathbf{f}_i^\top of F on top of each other.

This technique of computing epipoles first and subsequently reverting to the original point correspondences to solve for the projection matrices was used in [12], [11] for computing the trifocal tensor. A further refinement described in [11] is to iterate over the value of the epipole \mathbf{e} to minimize the residual corresponding to the solution of the system (1).

Each H_k is not uniquely defined by this method, since it computes only the first two rows of each H_k . In fact, the last row of H_k may be arbitrary. Generally, in factoring $F = [\mathbf{e}]_\times H$, we may replace H by $H_k + \mathbf{e}\mathbf{v}^\top$ for any arbitrary \mathbf{v} without changing the form of the equation. This is because $\mathbf{e}\mathbf{v}^\top$ cancels with $[\mathbf{e}]_\times$. Note that, since $\mathbf{e} = (0, 0, 1)^\top$, only the third row of $\mathbf{e}\mathbf{v}^\top$ is nonzero, so all of the ambiguity of H_k occurs in the third row.

We now turn to finding the final row of each homography H_k . Various methods of determining the unknown vector \mathbf{v} are available. Essentially, this problem is solved in [20] by considering a specific polynomial parametrization of the radial distortion curve. We prefer a parameter-free method of doing this based on two assumptions:

1. The distortion is circularly symmetric. Thus, the radial distortion of an image point depends only on its distance from the center of distortion.
2. An ordering or monotonicity condition, that is, the radial distance of points from the radial center after

distortion is a monotonic function of their distance before distortion.

The first condition will not hold in general unless the pixels are square. However, it turns out, in practice, not to be critical to the success of our method, which works well even with nonsquare pixels, and, in fact, the aspect ratio of the pixels naturally falls out of the computation. The second condition is an essential property of any camera and, indeed, it would be a strange camera that did not satisfy this condition since it would mean that the correspondence between distorted and undistorted image points was not one-to-one. Otherwise stated, some points in the scene would appear at more than one point in the image.

We consider a single homography H_k and temporarily drop the subscript k . Once we know the center of distortion of the camera, we may change coordinates in the image so that the center of distortion is the origin of pixel coordinates. In this case, $\mathbf{e} = (0, 0, 1)^\top$ in homogeneous coordinates and, so, the ambiguity in $H + \mathbf{e}\mathbf{v}^\top$ exists only in the last row of H . Let \hat{H} consist of the first two rows of H so that $H = [\hat{H}; \mathbf{v}^\top]$ (this notation means that \mathbf{v}^\top is the final row of H). In the following discussion, \mathbf{x}^d and \mathbf{x}^u are intended to represent two vectors, the dehomogenized vector representation of the points. On the other hand, \mathbf{x}^c represents a homogeneous three-vector representation of the calibration point. Now, defining $(\hat{x}^u, \hat{y}^u)^\top = \hat{\mathbf{x}}^u = \hat{H}\mathbf{x}^c$, we see that $H\mathbf{x}^c = [\hat{H}\mathbf{x}^c; \mathbf{v}^\top \mathbf{x}^c]$ and, dehomogenizing, we obtain

$$\mathbf{x}^u = (\hat{x}^u, \hat{y}^u) / (\mathbf{v}^\top \mathbf{x}^c) = \hat{H}\mathbf{x}^c / (\mathbf{v}^\top \mathbf{x}^c).$$

Thus, the effect of \mathbf{v}^\top is to stretch the point $\hat{\mathbf{x}}^u$ by the factor $1/(\mathbf{v}^\top \mathbf{x}^c)$, which depends on the point \mathbf{x}^c .

We now compute the radii of the distorted and undistorted points, setting $r^d = \|\mathbf{x}^d\|$ and $\hat{r}^u = \|\hat{\mathbf{x}}^u\|$ and, finally, $r^u = \|\hat{\mathbf{x}}^u / (\mathbf{v}^\top \mathbf{x}^c)\| = \hat{r}^u / |\mathbf{v}^\top \mathbf{x}^c|$. The notation $\|\cdot\|$ is used here to mean the radius of the point (its distance from the center of distortion).

These are the positive radii of the points. However, we need to consider a *signed* radius r^u , whereby the positive radial direction for a point \mathbf{x}^u is oriented toward the distorted point \mathbf{x}^d . Consequently, we define the signed radius \hat{r}^u to be positive or negative, depending on whether $\mathbf{x}^{d\top} \hat{\mathbf{x}}^u$ is positive or negative. That is,

$$\hat{r}^u = \text{sign}(\mathbf{x}^{d\top} \hat{\mathbf{x}}^u) \|\hat{\mathbf{x}}^u\|$$

and

$$r^u = \hat{r}^u / (\mathbf{v}^\top \mathbf{x}^c),$$

where the absolute value no longer appears in the denominator. This radius r^u may be positive or negative and it is positive if the radial vector toward \mathbf{x}^u is in the same direction as the distorted point \mathbf{x}^d . If \mathbf{v}^\top is correctly chosen, both the distorted and undistorted points will be in the same direction, so r^u will be positive.

5.4 The Monotonicity Assumption for Radial Distortion Function

If we were able to select the correct value of the vector \mathbf{v} , then the points (r^d, r^u) would lie along (or, with noise, close to) a monotonic curve, as illustrated in the scatter plot of Fig. 8. For any other (incorrect) value of \mathbf{v} , the scatter plot would be irregularly scattered because of the different scaling of each value r^u according to $\mathbf{v}^\top \mathbf{x}^c$. It is our goal to

find the value of \mathbf{v} that reduces the scatter plot to a monotonic smooth curve. Perhaps surprisingly, this can be accomplished well by simple least-squares techniques.

We begin by arranging the values of r^d in ascending order and indexing them in order by an index i : We write $r_i^d; i = 1, \dots, N$. Note that the ordering of the radii r_i^d is done only *once*. The corresponding value of $r_i^u = \hat{r}_i^u / (\mathbf{v}^\top \mathbf{x}_i^c)$ may be thought of as a discrete function of the values r_i^d . We define the *total squared variation* of this function to be

$$V = \sum_{i=1}^{N-1} (r_{i+1}^u - r_i^u)^2.$$

If the values of r_i^u are a monotonic function of r_i^d , then the total squared variation of this function will be relatively small compared to that of an irregular function. In fact, it is easily seen that, for a monotonic function, $V < (r_N^u - r_1^u)^2$ and, in fact, if the values of r_i^u are evenly scattered between r_1^u and r_N^u , then $V \rightarrow 0$ as $N \rightarrow \infty$. In measurements involving a large number of points, the radii will, in general, be well scattered and the value of the total squared variation will be small, although we do not expect it to vanish completely. Our method, therefore, is to minimize the total squared variation V as a function of the parameter vector \mathbf{v} .

The above discussion extends easily to the case of multiple images. In this case, we consider the radii of the points in all images together and order them in one single list. The undistorted radius r^u is equal to $\hat{r}^u / (\mathbf{v}_k^\top \mathbf{x}_i^c)$, where k is an index representing the image number. For each image k , there will be a different homography \mathbf{H}_k and a different corresponding vector \mathbf{v}_k . The total square variation must be minimized over the choice of all \mathbf{v}_k .

5.5 Minimizing Total Squared Variation

We wish to minimize

$$\sum_{i=1}^{N-1} (r_{i+1}^u - r_i^u)^2 = \sum_{i=1}^{N-1} \left(\frac{\hat{r}_{i+1}^u}{\mathbf{v}_{k(i+1)}^\top \mathbf{x}_{i+1}^c} - \frac{\hat{r}_i^u}{\mathbf{v}_{k(i)}^\top \mathbf{x}_i^c} \right)^2, \quad (2)$$

where $k(i)$ represents the image number corresponding to the i th point. As it stands, this is not a linear least-squares problem. However, multiplying each term by $(\mathbf{v}_{k(i+1)}^\top \mathbf{x}_{i+1}^c)^2 (\mathbf{v}_{k(i)}^\top \mathbf{x}_i^c)^2$ leads to

$$\sum_{i=1}^{N-1} (\hat{r}_{i+1}^u \mathbf{v}_{k(i)}^\top \mathbf{x}_i^c - \hat{r}_i^u \mathbf{v}_{k(i+1)}^\top \mathbf{x}_{i+1}^c)^2, \quad (3)$$

which must be minimized over all \mathbf{v}_k . Note that, apart from the values of \mathbf{v}_k , all other quantities appearing in this expression are known. The minimization problem is one of minimizing the squared norm of the $(N-1)$ -dimensional vector with entries

$$\hat{r}_{i+1}^u \mathbf{v}_{k(i)}^\top \mathbf{x}_i^c - \hat{r}_i^u \mathbf{v}_{k(i+1)}^\top \mathbf{x}_{i+1}^c, \quad (4)$$

which are linear in the entries of the vectors \mathbf{v}_k .

Evidently, as it stands, the norm of the above vector is minimized by setting all \mathbf{v}_k to zero. In order to avoid this, we may impose the supplementary condition that $\sum_k \|\mathbf{v}_k\|^2 = 1$. As discussed in Section 4, there is an ambiguity between the focal length of the camera (or equivalently, in this case, the scale of the homography) and a multiplicative factor applied to the radial distortion function that allows us to enforce such a condition.

Now, we may write the vector with entries given in (4) as $\mathbf{A}\mathbf{V}$, where \mathbf{V} is a vector containing all of the entries of the \mathbf{v}_k and \mathbf{A} is the matrix of coefficients, which may be easily constructed from the known values of all the \hat{r}_i^u and \mathbf{x}_i^c . This problem can be solved by minimizing $\|\mathbf{A}\mathbf{V}\|$, subject to the condition $\|\mathbf{V}\| = 1$. An alternative (which we prefer) is to impose the condition on $\mathbf{v}_{k(N)}$ that ensures that $r_N^d = r_N^u = \hat{r}_N^u / (\mathbf{v}_{k(N)}^\top \mathbf{x}_N^c)$, namely, that the distorted and undistorted radii are equal for the point most distant from the center of distortion. This gives a linear equality condition on $\mathbf{v}_{k(N)}$ and leads to a simple linear least-squares estimation problem. (Note that we are free to make this assumption that $r_N^d = r_N^u$ because of the ambiguity between the scale of the distortion and the overall scale represented by the calibration matrix, specifically the focal length.)

Note that we linearized the problem by multiplication by $(\mathbf{v}_{k(i+1)}^\top \mathbf{x}_{i+1}^c)^2 (\mathbf{v}_{k(i)}^\top \mathbf{x}_i^c)^2$, which results in an unequal weighting of the individual equations and means that we do not exactly minimize the total squared variance of the distortion curve. However, this effect is quite benign since the values of $\mathbf{v}_{k(i)}^\top \mathbf{x}_i^c$ represent the depth of the calibration point in the direction of the camera's principal ray. Under the assumption that of all these depths are approximately equal, all of the equations are weighted approximately equally. The advantage is that it changes a nonlinear estimation problem into a linear one. When testing this algorithm, we followed the linear estimation step by nonlinear refinement to minimize the true total squared variance, as well as various other geometrically derived conditions designed to lead to a smooth monotonic distortion curve. Suffice it to say that the improvements achieved were minimal.²

5.6 Local Linearity of Radial Distortion

An alternative method for enforcing the monotonicity constraint on the radial distortion function makes use of a different criterion than the total squared-variance criterion discussed above.

Instead, we use an assumption of local linearity, which means that the radial distortion curve (see Fig. 8 for an example) is locally linear, at least on the scale of the distance between consecutive radial samples. As seen in the graph, typically, we have a very large number of such samples and the distance between them (radially) is quite small. Thus, an assumption of linearity is easily justifiable.

Let r_{i-1}^d , r_i^d , and r_{i+1}^d be three consecutive values for the distorted radial distance from the center of distortion, ordered by their magnitude as before. There exists a constant α_i such that $r_i^d = (1 - \alpha_i)r_{i-1}^d + \alpha_i r_{i+1}^d$. These values, α_i , may be computed in advance. Under an assumption of linearity, the undistorted radii satisfy a similar relationship: $r_i^u = (1 - \alpha_i)r_{i-1}^u + \alpha_i r_{i+1}^u$. Therefore, we define the following error term:

$$\begin{aligned} e_i &= (1 - \alpha_i)r_{i-1}^u - r_i^u + \alpha_i r_{i+1}^u \\ &= (1 - \alpha_i) \frac{\hat{r}_{i-1}^u}{\mathbf{v}_{k(i-1)}^\top \mathbf{x}_{i-1}^c} - \frac{\hat{r}_i^u}{\mathbf{v}_{k(i)}^\top \mathbf{x}_i^c} \\ &\quad + \alpha_i \frac{\hat{r}_{i+1}^u}{\mathbf{v}_{k(i+1)}^\top \mathbf{x}_{i+1}^c} \quad \text{for } i = 2, \dots, N-1. \end{aligned} \quad (5)$$

2. To be precise, nonlinear refinement caused less than 1.2 percent decrease in residual in the example of Fig. 3 and no improvement in the final result.

Since three potentially different terms appear in the denominator of this expression, there is no obvious way in which this can be made into a linear function of the \mathbf{v}_k . Nevertheless, it is possible to apply iterative minimization techniques (such as Levenberg–Marquardt) to this problem to minimize the cost over all choices of the vectors \mathbf{v}_k .

Note that it is not vitally essential that all of the values of the r_i^d be precisely ordered during this minimization. Approximate ordering is sufficient for an assumption of local linearity to be valid. During nonlinear iterative optimization of this function, the value (and, hence, the ordering) of a radial distance value r_k^d does not change. However, it is possible to include the coordinates of the radial distortion center as a parameter in a full bundle adjustment to refine the final solution. In this case, the radial distances change and so does their order. Note that the radial distance changes most significantly for points near the center of distortion; however, for these points, the radial distortion function is most nearly linear, so the local linearity assumption is still valid during movement of the center of distortion.

5.7 Estimation of the Radial Distortion Function

In this paper, we deliberately do not consider parameterization of the distortion function. This distortion function is often modeled as $r^d = r^u(1 + \kappa_1 r^{u2} + \kappa_2 r^{u4} + \dots)$. Such a distortion function does not work well for large distortions such as those given by fish-eye lenses. It is a strength of our method that it does not rely on any particular distortion model and it has been tested successfully on fish-eye, wide-angle, and narrow-angle lenses.

Of course, ultimately, the curve must be approximated by some technique to be useful for image correction. There is a large literature on the topic of function approximation (see [1] for a good summary). Separating out the problem of determination of a parameterized approximation of the correction curve allows more sophisticated techniques and algorithms to be used. These include approximation of the distortion function by Chebyshev polynomials or the Remes minimax algorithm. Direct estimation of the polynomial coefficient (κ_i) or use of a Taylor expansion is ill-conditioned or suboptimal [1]. The method we preferred for the examples given later involved approximation of the distortion function expressed in terms of angular distortion of points on a spherical rather than planar retina, though that is not a critical choice, except for very wide angle lenses. An alternative is to compute a parameter-free model for the distortion curve by computing the radial displacement at each radial value as a sliding median of local measurements.

5.8 Camera Calibration

Estimation of the vectors \mathbf{v}_k allows us to complete the homography matrices \mathbf{H}_k . From this, we can compute the true undistorted projection of each of the calibration points \mathbf{x}^c into the image. Given these homographies for at least three images, we are able to compute the calibration matrix of the undistorted cameras using the algorithm of [24].

Specifically, knowing the homographies \mathbf{H}_k , we may project the images of the circular points [12], namely, $(1, \pm i, 0)$ into the image plane. The images of the circular points lie on the image of the absolute conic (IAC), denoted by $\omega = (\mathbf{K}\mathbf{K}^\top)^{-1}$. Since the IAC is a conic, it may be estimated using five or more points, hence, three or more images. Although these points are complex points, satisfying $\mathbf{x}^\top \omega \mathbf{x} = 0$, separation of real and

imaginary parts gives two linear equations in the five different entries of the symmetric 3×3 matrix ω . The equations take the form $\mathbf{h}_1^\top \omega \mathbf{h}_1 = \mathbf{h}_2^\top \omega \mathbf{h}_2$ and $\mathbf{h}_1^\top \omega \mathbf{h}_2 = 0$, where \mathbf{h}_1 and \mathbf{h}_2 are the first two columns of the homography matrix \mathbf{H}_k . Finally, the calibration matrix is computed by solving $\omega = (\mathbf{K}\mathbf{K}^\top)^{-1}$ by inversion and Cholesky factorization.

Due to the ambiguity of the distortion function and the focal length, it is customary to scale the distortion function so that undistorted and distorted points are the same in the limit for small radii. This is what we do for the reported results.

5.9 Nonsquare Pixels

If the pixels are not square, then distortion is still radial, but the radial distortion function is not symmetric.

We are faced with a chicken-and-egg situation: If we knew the aspect ratio of the pixels, then we could correct for it before trying to estimate the distortion curve. However, we cannot estimate the camera calibration without first computing the homographies, which is tied to computing the distortion curve.

Despite this, we found that, by simply proceeding with the algorithm as described, estimation of the homographies and distortion curve, followed by internal calibration, led to an accurate estimate of the aspect ratio of the pixels. This would allow us to correct for the aspect ratio and recompute the distortion curve. The explanation for this is that, even with errors in the expansion ratio due to nonsquare pixels, the distortion curve has the correct shape and leads to the correct value of the \mathbf{v}_k . The effect of assuming square pixels when they, in fact, are not leads to a slight broadening of the distortion curve but not to a change in its shape. Note that the assumption of square pixels is only required in the phase of the algorithm involving computation of the last row of each \mathbf{H}_k . Computation of the center of distortion and the first two rows of each \mathbf{H}_k is independent of this assumption.

5.10 Bundle Adjustment

The algorithm we have described is entirely noniterative and is consequently very rapid. We do not suggest that an iterative refinement of the results would not improve accuracy. However, this method provides a quick and not-so-dirty estimate of the camera calibration and radial distortion. Iterative methods could be used:

1. In estimation of the fundamental matrices (we used the simple normalized eight-point algorithm). For the multiview case, the condition that all of the epipoles are the same could be used.
2. In estimation of the vectors \mathbf{v}_k , completing the homographies. We tried iterative techniques that minimized the total squared variance exactly and also the method that tends to enforce local linearity using the cost function (5). The improvement was quite minor.
3. In computation of the calibration matrix \mathbf{K} .
4. For final bundle adjustment in which we should minimize the reprojection error over the parameters of motion (pose of each camera), the calibration matrix, and some description of the distortion curves. This would surely give some small decrease in reprojection error. However, parameterizing the distortion curve explicitly and nonlinear estimation can lead to overfitting. Our linear results obtained a root-mean-square

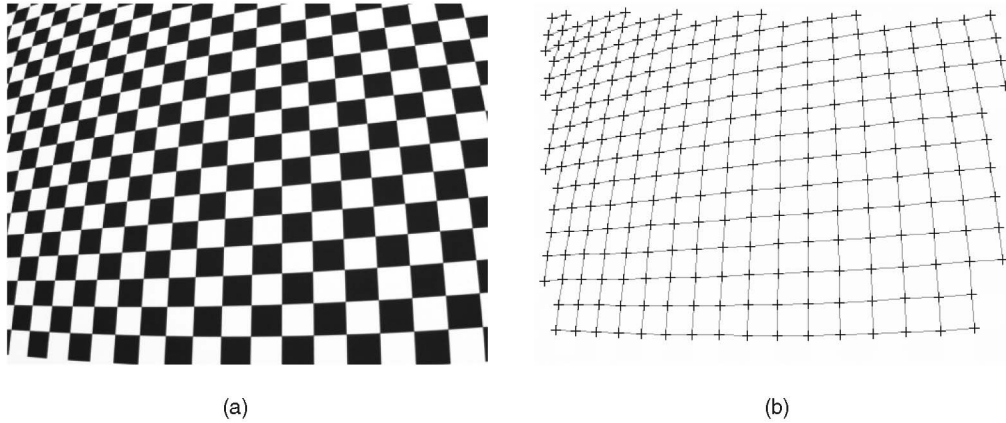


Fig. 2. (a) An image of a checkerboard and (b) the detected grid.

(RMS) reprojection error of 0.4 pixels, which is close to the minimum one might expect.

6 RESULTS FOR THE PLANAR CALIBRATION GRID

In this section, we present distortion removal results for a known planar calibration grid.

The calibration pattern is a checkerboard with black-and-white squares (Fig. 2a). The corners of the checkerboard are extracted by finding edges and computing intersection points; sample results are shown in Fig. 2b. We used 19 images of the checkerboard taken at various poses.

After we applied our technique as described in Sections 5.1-5.8, we obtain the graph of radial shifts as a function of radial position as shown in Fig. 3.

Fig. 4 shows the distribution of the estimated distortion center in three cases: 1) Each image used separately, 2) five images randomly chosen from 19, and 3) 10 randomly chosen images. The distortion center estimated using all 19 images is (306.7, 260.5) and the image center is (320, 240). The principal point distributions extracted using the same experiments are shown in Fig. 5. (The principal point, unrelated to the center of distortion, is the foot of the perpendicular from the projection center to the image plane.) The principal point estimated using all 19 images is (312.0, 244.8). Note that,

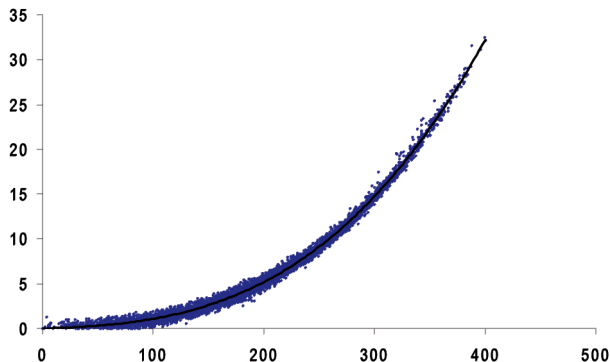


Fig. 3. Graph of radial distortion $r^u - r^d$ versus radial position r^d using corner points from 19 checkerboard images. The number labels are all in pixels; the image resolution is 640×480 . In this case, we achieved an RMS error of 0.4 pixels in modeling the measured image points.

while the estimated principal point is close to the image center, the estimated distortion center is not.

We used the estimated image noise (based on RMS error fit of $\sigma = 0.4$) and performed Monte Carlo simulation with 1,000 random trials. The results are shown in Fig. 6. Compare the locations of the estimated distributions for the center of distortion and principal point and the image center. The mean for the distortion center is (304.3, 259.2), with a standard deviation of (0.87, 0.60). The mean for the principal point is (310.6, 244.6), with a standard deviation of (0.58, 2.26).

We applied the computed radial distortion mapping to a number of different input images for correction. Two results are shown in Fig. 7.

In another set of experiments, we applied our planar calibration technique on three additional cameras. These three cameras are of the same type (PointGrey Fleas, resolution of $1,024 \times 768$), use the same type of 4 mm lens, and have the same settings. The results based on 10 images of the checkerboard pattern are shown in Table 1. Notice the significant differences in the locations of the estimated distortion center, estimated principal point, and image center. Notice also the significant differences in the estimated distortion center and principal point for the same type of cameras.

We also applied our technique to cameras with fish-eye lenses. The first camera has a field of view close to 180° . A sample image of the checkerboard pattern (we use eight calibration images), the extracted grid, and radial distortion results are shown in Fig. 8. The image resolution is $1,536 \times 1,024$, the extracted distortion center is (744.4, 488.2), and the principal point is (767.5, 492.4) (compared to the image center (768, 512)). As can be seen, the corrected results look very reasonable.

The second camera is fitted with a fish-eye lens with a field of view of about 190° (Fig. 9). Here, the image resolution is only 640×480 ; in this case, we manually picked the locations of corners (in three calibration images) to within a pixel to limit the image errors. The extracted distortion center is (321.3, 267.2) and the recovered principal point is (346.4, 258.4), compared with the image center at (320, 240). Again, our corrected result looks reasonable.

For fish-eye lenses that extend beyond 180° , the undistorted radii are undefined on a planar retina (since they become infinite). However, they are well defined on a

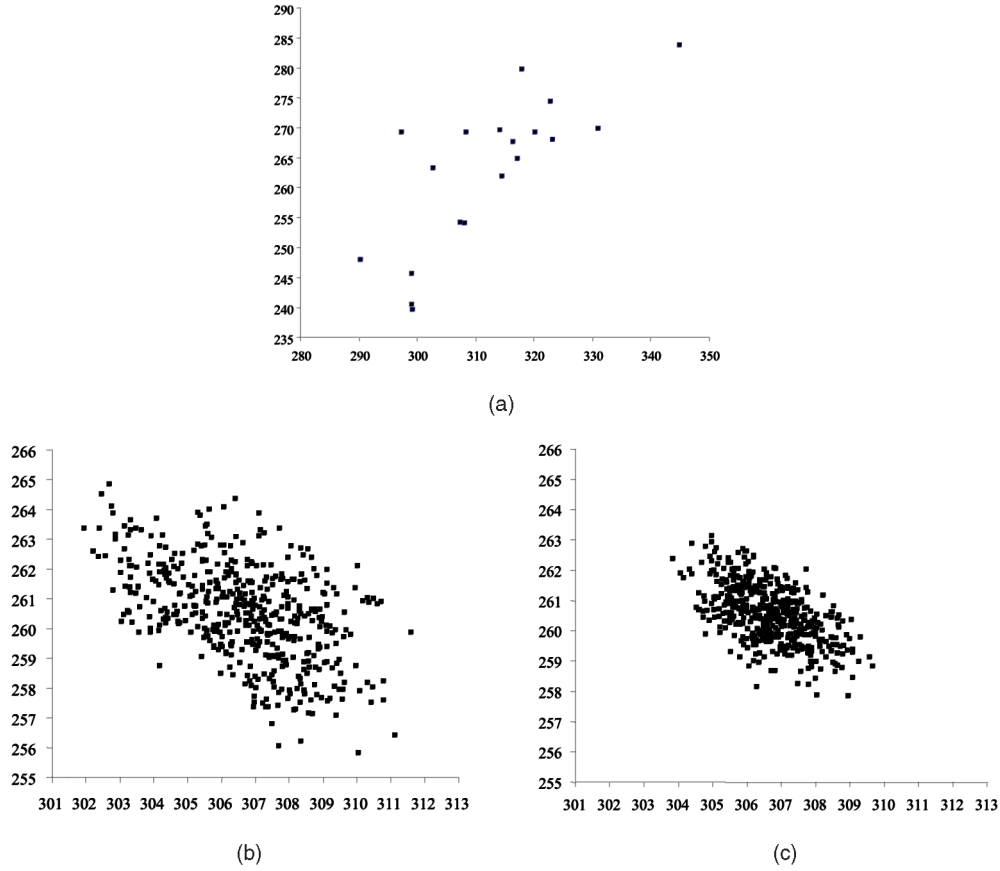


Fig. 4. Distributions of distortion center. (a) Using single image separately (19 inputs). (b) Using random sets of five images chosen from 19 (5 of 19). (c) Using random sets of 10 images (10 of 19). Note the different scales in the graphs. The number labels are all in pixels; the image resolution is 640×480 .

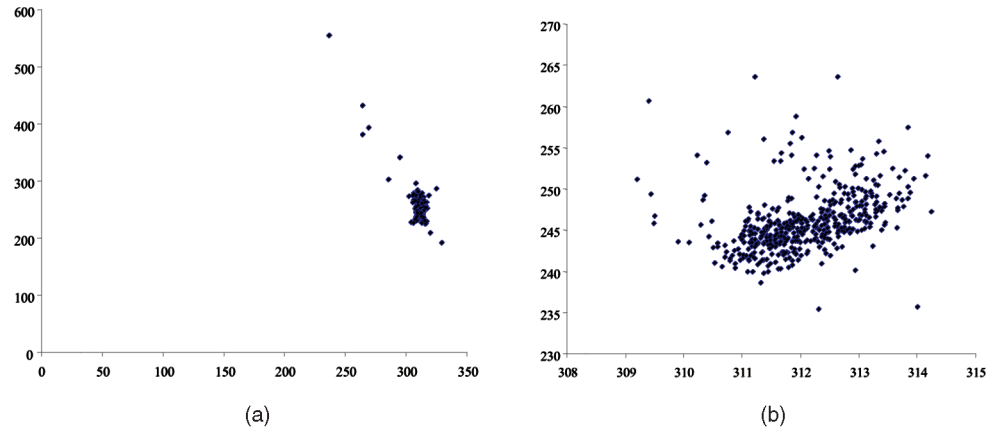


Fig. 5. Distributions of principal point. (a) Using random sets of five images (5 of 19). (b) Using random sets of 10 images (10 of 19). Note the different scales in the graphs. The number labels are all in pixels; the image resolution is 640×480 .

spherical retina. It is still possible to apply the linear algorithm for computing the distortion curve in this case because of the following observation: There is no problem at all with the phases of the algorithm involving computation of the center of distortion (Section 5.1) or the estimation of the first two rows of the homography matrices (Section 5.3). There is an apparent difficulty with the definition of the total squared variance in Section 5.5 since some of the terms of (2) may approach infinity. However, in estimating the final rows \mathbf{v}_k of the homography matrices by minimizing the total

squared variance using (2), it is not necessary to use all the points. In particular, one may omit points toward the periphery of the image in carrying out this computation, thereby avoiding the problem of points with infinite (or beyond-infinite) undistorted radii. In using the linearized cost function (3), the problem of infinite terms disappears.

No radial distortion. The previous examples were of cases where there was substantial radial distortion. The following question arises: What happens if there is no radial distortion in the first place? As pointed out previously, in this case, the center of radial distortion is

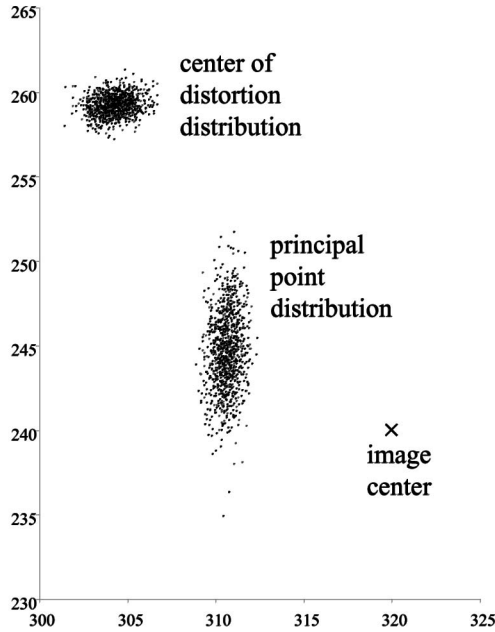


Fig. 6. Monte Carlo simulation (1,000 trials with $\sigma = 0.4$). The number labels are all in pixels; the image resolution is 640×480 .

not well defined. We suggested a test based on the degrees of freedom for the fundamental matrix. Alternatively, a sanity check can be used to check the value of e (if it is too far away from the image center, for example) and the shape of the distortion curve (if it is all close to zero distortion with random noise) to decide on the presence of distortion. In simulations, we obtain very unlikely values for e and virtually zero distortions. In one typical run, we obtained $e = (-593.9, -1, 576.4)^T$ for a noiseless 640×480 image with 100 well-distributed points and the

TABLE 1
Comparison of Results for Three Different Cameras
(Same Model and Same Type of 4 mm Lens)

#	c_d	c_p	c_0
1	(574.4, 401.5)	(555.6, 377.7)	(512, 384)
2	(545.9, 394.5)	(542.6, 396.6)	(512, 384)
3	(577.8, 392.3)	(574.4, 391.0)	(512, 384)

c_d , c_p , and c_0 are the distortion center, principal point, and image center, respectively. c_d and c_p were computed based on 10 images of the same checkerboard pattern at different poses.

amount of distortion was virtually zero. Here, the best fit fourth-order distortion polynomial was

$$\mathbf{x}^u = \mathbf{x}^d \left(1 + 1.7 \times 10^{-10} |\mathbf{x}^d| - 1.3 \times 10^{-13} |\mathbf{x}^d|^2 + 4.2 \times 10^{-17} |\mathbf{x}^d|^3 - 5.1 \times 10^{-21} |\mathbf{x}^d|^4 \right).$$

7 EXTENSIONS OF THE BASIC IDEA

We describe several extensions to this calibration method based on the general idea of using structure and motion techniques (similar to the use of the fundamental matrix above) to estimate radial distortion. We have implemented all of these methods in some form and verified their correctness.

7.1 Using a Nonplanar Calibration Grid

The method was described in Section 5 in terms of using a known planar calibration grid. However, the method carries over entirely to a nonplanar calibration rig. Instead of modeling the mapping of the calibration grid to the image using a 3×3 homography matrix H , the mappings

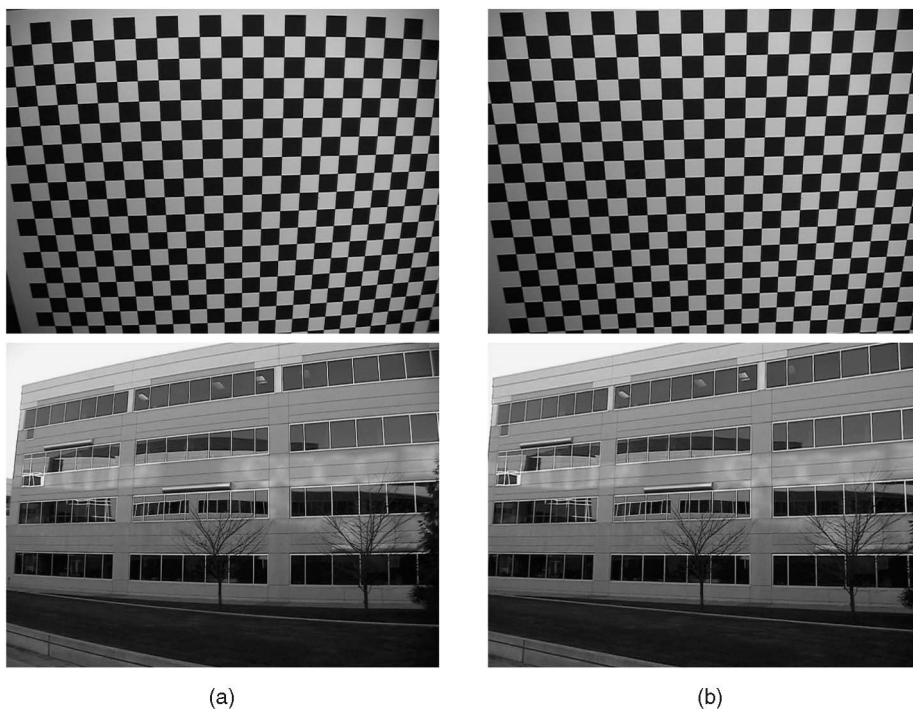


Fig. 7. Two distortion removal results: (a) inputs and (b) corrected outputs.

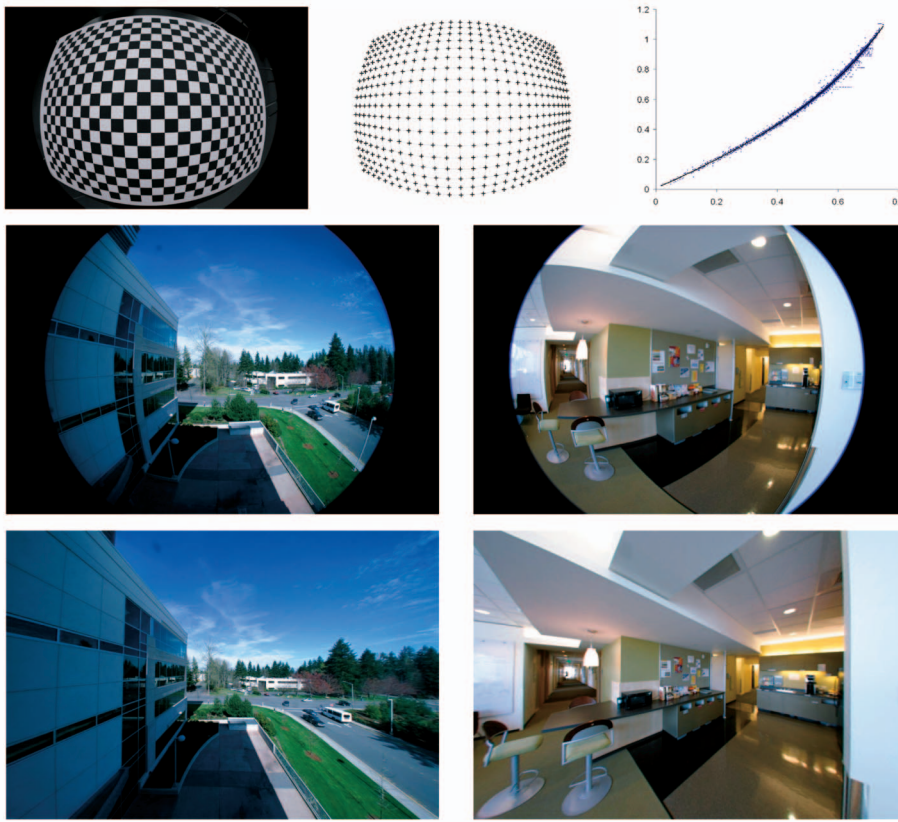


Fig. 8. Fish-eye image experiment: An image of the checkerboard and extracted grid (top row), the radial distortion curve (undistorted versus distorted angle in radians, top row, right), two originals (middle row), and two radially corrected images (bottom row).

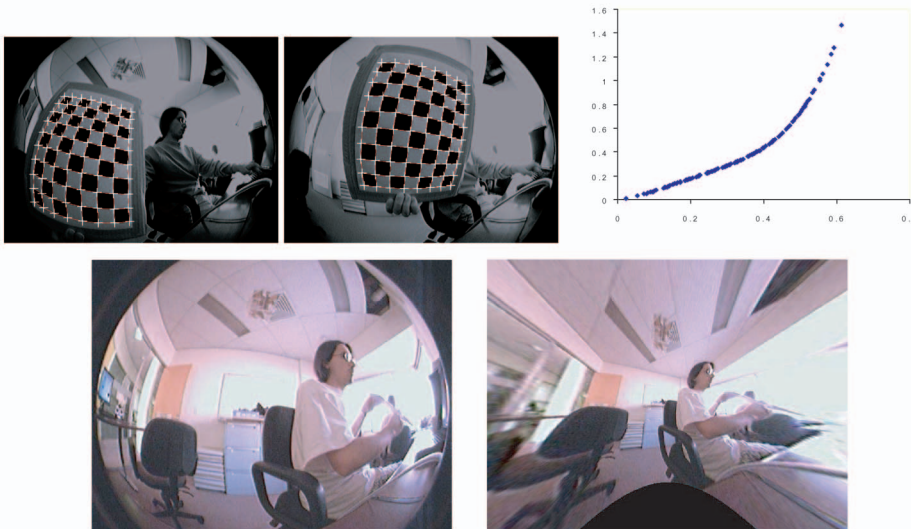


Fig. 9. Another fish-eye image experiment: two darkened images of a checkerboard and manually picked points (top row), the radial distortion curve (undistorted versus distorted angle in radians, top row, right), original and radially corrected images (bottom row) (fish-eye images courtesy of Nick Barnes and Cedric Pradalier).

are modeled as camera projections using a standard 3×4 projection matrix P . Instead of a fundamental-matrix G (which has a 3×4 dimension) in just the same way in which one estimates the fundamental matrix. Computation of the final row \mathbf{v} of the projection matrix P proceeds as before, as does estimation of the distortion curve. The only difference involves the computation of the calibration matrix. Instead of needing three views to compute K , it can be computed from a

single view. A single projection matrix P gives five equations in the entries of ω , namely, $\mathbf{p}_1^\top \omega \mathbf{p}_1 = \mathbf{p}_2^\top \omega \mathbf{p}_2 = \mathbf{p}_3^\top \omega \mathbf{p}_3$ and $\mathbf{p}_1^\top \omega \mathbf{p}_2 = \mathbf{p}_2^\top \omega \mathbf{p}_3 = \mathbf{p}_1^\top \omega \mathbf{p}_3 = 0$. Thus, we can solve for ω and, hence, K from a single view or from many views by linear least-squares techniques.

We tried this method on synthetic data and verified that it worked with similar accuracy to the planar method. We also tried the technique on a real nonplanar calibration grid. The

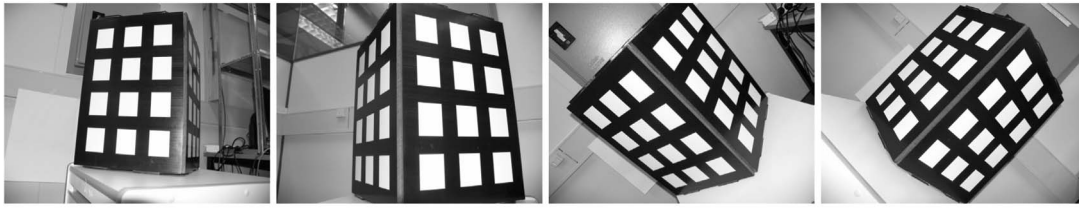


Fig. 10. Four input images of a nonplanar calibration grid.

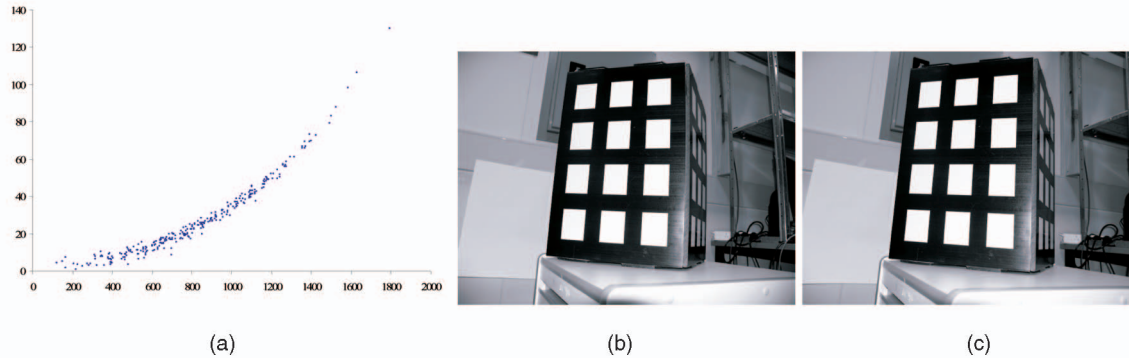


Fig. 11. Results for the nonplanar calibration experiment. (a) Extracted distortion graph. (b) Original image. (c) Corrected image.

four input images (each of $3,264 \times 2,448$ resolution) are shown in Fig. 10; here, we manually find the 2D coordinates associated with the 3D locations on the calibration box. The accuracy of the 3D locations is within about 1 mm (corresponding to 3-5 pixels) and the 2D image locations are within 1 pixel. The extracted curve and the result of undistorting an image captured using the same camera are shown in Fig. 11. As can be seen, despite the uncertainties associated with the 3D and 2D locations of the calibration points, the corrected image looked very reasonable.

7.2 Doing without the Calibration Grid

It is natural to ask whether we could do without the calibration grid entirely. The method described above used the fundamental matrix as the main algebraic tool. However, the process is analogous to determining the pose of a camera using line correspondences. Indeed, in the case of 3D points, the correspondence equation $\mathbf{x}^{dT} \mathbf{G} \mathbf{x}^c = 0$ is of precisely the same form as the equation $\mathbf{l}^T \mathbf{P} \mathbf{x} = 0$ for the image of a point \mathbf{x} to lie on a line \mathbf{l} when projected by camera matrix \mathbf{P} . Thus, the determination of the projection matrices and the center of distortion is similar to that of reconstruction from line correspondences. In fact, the line reprojection problem can be solved using the quadrifocal tensor with four views. Corresponding distorted points $\mathbf{x}_j = (x_j^1, x_j^2, x_j^3)^T$ in image j for $j = 1, \dots, 4$ will satisfy the relationship

$$\sum_{p=1}^3 \sum_{q=1}^3 \sum_{r=1}^3 \sum_{s=1}^3 \mathbf{x}_1^p \mathbf{x}_2^q \mathbf{x}_3^r \mathbf{x}_4^s Q_{pqrs} = 0,$$

from which it is possible to compute the quadrifocal tensor Q [10] and, hence, the projection matrices [9]. There is a slight complication that two solutions exist in both the quadrifocal and trifocal cases, as pointed out in [9].

We verified that this method will work for the present problem also. A total of 80 point correspondences are needed since each point gives only one equation. The same method works equally for calibration using unknown points on a

plane. In this case, we need only three views and a trifocal tensor is used. In this case, 26 points are needed. However, although we verified both practically and theoretically that this method will work, experiments with synthetic data showed the methods to be far too sensitive to noise to be useful, at least with an unknown center of distortion.

It is appropriate to point out the relationship of this work to that of [20]. In our work, we make no assumption on the position of the center of distortion. Thirithala and Pollefeys, on the other hand, assume that the center of distortion is known. In this case, assuming it to be at $(0, 0, 1)$, we can immediately set most of the entries of the tensor to zero, leaving only $2^4 = 16$ (in the 3D case) or $2^3 = 8$ (in the planar case) nonzero entries to estimate. This large reduction in the number of unknowns makes it likely that the method will be far more immune to noise. Thirithala and Pollefeys considered this problem for the case of 2D points in [20], but using 3D points is a natural extension, which was they explored in [19].

Since this work has been extensively studied in the work of Thirithala and Pollefeys, we omit further discussion of it from this paper.

8 CONCLUSIONS

The procedure for radial distortion calibration described in this paper represents a very reliable method of determining the center of distortion and radial distortion function for a wide range of cameras, including fish-eye lenses. At the same time, it allows computation of the internal geometric parameters of the camera. As a fast noniterative procedure, it may be used to initialize a bundle-adjustment algorithm, though, for many applications not requiring extreme precision, it can be used on its own as a standalone algorithm for camera calibration.

We argued the necessity of determining the radial distortion center and showed that, for all cameras that we tried, the distortion center was significantly displaced from

the center of the image, typically by as much as 30 pixels in a 640×480 image. Our experiments showed that this was enough to cause an extra 0.4 pixel of error for some points in the image (though the RMS error only increased by 0.1 pixel). For comparison, we were able to model the measured image points with RMS error of 0.4 pixel (see the example of Fig. 3).

Extensions to autocalibration techniques involving images of planar or nonplanar points are possible, but may be too sensitive to noise to be useful when the center of distortion needs to be computed as well.

ACKNOWLEDGMENTS

National ICT Australia is funded by the Australian Government's Department of Communications, Information Technology, and the Arts and the Australian Research Council through Backing Australia's Ability and the ICT Research Centre of Excellence programs.

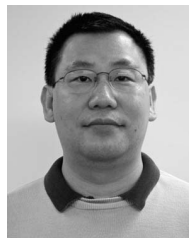
REFERENCES

- [1] K.E. Atkinson, *An Introduction to Numerical Analysis*. Wiley, 1978.
- [2] S. Becker and V.B. Bove, "Semiautomatic 3-D Model Extraction from Uncalibrated 2-D Camera Views," *SPIE Visual Data Exploration and Analysis II*, vol. 2410, pp. 447-461, 1995.
- [3] D.C. Brown, "Decentering Distortion of Lenses," *Photogrammetric Eng.*, vol. 32, no. 3, pp. 444-462, May 1966.
- [4] D.C. Brown, "Close-Range Camera Calibration," *Photogrammetric Eng.*, vol. 37, no. 8, pp. 855-866, 1971.
- [5] Z. Cao, S.J. Oh, and E. Hall, "Omnidirectional Dynamic Vision Positioning for a Mobile Robot," *Optical Eng.*, vol. 25, no. 12, pp. 1278-1283, Dec. 1986.
- [6] T.A. Clarke, J.F. Fryer, and X. Wang, "The Principal Point for CCD Cameras," *Photogrammetric Record*, vol. 16, no. 92, pp. 293-312, 1998.
- [7] W. Faig, "Calibration of Close-Range Photogrammetry Systems: Mathematical Formulation," *Photogrammetric Eng. and Remote Sensing*, vol. 41, no. 12, pp. 1479-1486, Dec. 1975.
- [8] A.W. Fitzgibbon, "Simultaneous Linear Estimation of Multiple View Geometry and Lens Distortion," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 125-132, Dec. 2001.
- [9] R. Hartley and F. Schaffalitzky, "Reconstruction from Projections Using Grassman Tensors," *Proc. Eighth European Conf. Computer Vision, Part III*, pp. 363-375, May 2004.
- [10] R.I. Hartley, "Computation of the Quadrifocal Tensor," *Proc. Fifth European Conf. Computer Vision*, pp. 20-35, 1998.
- [11] R.I. Hartley, "Minimizing Algebraic Error," *Philosophical Trans. Royal Soc. London, Series A*, vol. 356, no. 1740, pp. 1175-1192, 1998.
- [12] R.I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge Univ. Press, 2000.
- [13] B. Micusik and T. Pajdla, "Estimation of Omnidirectional Camera Model from Epipolar Geometry," *Computer Vision and Pattern Recognition*, vol. 1, pp. 485-490, June 2003.
- [14] S. Ramalingam, P. Sturm, and S.K. Lodha, "Towards Complete Generic Camera Calibration," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 1093-1098, June 2005.
- [15] S. Shah and J.K. Aggarwal, "A Simple Calibration Procedure for Fisheye (High Distortion) Lens Camera," *Proc. Int'l Conf. Robotics and Automation*, pp. 3422-3427, 1994.
- [16] G. Stein, "Accurate Internal Camera Calibration Using Rotation, with Analysis of Sources of Error," *Proc. Int'l Conf. Computer Vision*, pp. 230-236, June 1995.
- [17] R. Swaminathan and S. Nayar, "Non-Metric Calibration of Wide-Angle Lenses and Polycameras," *Computer Vision and Pattern Recognition*, pp. 413-419, June 1999.
- [18] J.-P. Tardif and P. Sturm, "Self-Calibration of a General Radially Symmetric Distortion Model," *Proc. Ninth European Conf. Computer Vision*, 2006.
- [19] S. Thirithala and M. Pollefeys, "Multi-View Geometry of 1D Radial Cameras and Its Application to Omnidirectional Camera Calibration," *Proc. 10th Int'l Conf. Computer Vision*, pp. 1539-1546, 2005.
- [20] S. Thirithala and M. Pollefeys, "The Radial Trifocal Tensor: A Tool for Calibrating the Radial Distortion of Wide-Angle Cameras," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, pp. 321-328, 2005.
- [21] Y.R. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses," *IEEE J. Robotics and Automation*, vol. 3, no. 4, pp. 323-344, Aug. 1987.
- [22] G.-Q. Wei and S.D. Ma, "Implicit and Explicit Camera Calibration: Theory and Experiments," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 5, pp. 469-480, May 1994.
- [23] Y. Xiong and K. Turkowski, "Creating Image Based VR Using a Self-Calibrating Fisheye Lens," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 237-243, 1997.
- [24] Z. Zhang, "A Flexible New Technique for Camera Calibration," *Proc. Seventh Int'l Conf. Computer Vision*, pp. 666-673, Sept. 1999.



Richard Hartley received the degree from the University of Toronto in 1976 with a thesis on knot theory. He works in computer vision at the Australian National University and at National ICT Australia, a government-funded research institute. He worked in the area of knot theory for several years before joining the General Electric Research and Development Center, where he developed computer-aided electronic design tools and created a very successful design

system called the Parsifal Silicon Compiler, which was described in his book *Digit Serial Computation*. In 1991, he was awarded GE's Dushman Award for this work. Around 1990, he developed an interest in computer vision and, in 2000, he coauthored (with Andrew Zisserman) a book on multiple view geometry. He has written papers in knot theory, geometric voting theory, computational geometry, computer-aided design, and computer vision and holds 32 US patents.



Sing Bing Kang received the PhD degree in robotics from Carnegie Mellon University in 1994. He is currently a senior researcher at Microsoft Corporation working on environment modeling from images as well as image and video enhancement. He has coedited two books on computer vision (*Panoramic Vision* and *Emerging Topics in Computer Vision*) and recently coauthored a book on image-based rendering. He has also served as area chair and

member of the technical committee for ICCV, CVPR, and ECCV and is currently a program cochair for ACCV 2007.

► For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.