



Article

# Radial Distortion from Epipolar Constraint for Rectilinear Cameras

Ville V. Lehtola <sup>1,2,\*</sup>, Matti Kurkela <sup>1</sup> and Petri Rönholm <sup>1</sup>

<sup>1</sup> Department of Built Environment, Aalto University, P.O. Box 15800, 00076 Aalto, Finland; matti.kurkela@aalto.fi (M.K.); petri.ronholm@aalto.fi (P.R.)

<sup>2</sup> National Land Survey of Finland, Finnish Geospatial Research Institute FGI, PL 84, 00521 Helsinki, Finland

\* Correspondence: ville.lehtola@iki.fi

Academic Editor: Gonzalo Pajares Martinsanz

Received: 10 August 2016; Accepted: 13 January 2017; Published: 24 January 2017

**Abstract:** Lens distortion causes difficulties for 3D reconstruction, when uncalibrated image sets with weak geometry are used. We show that the largest part of lens distortion, known as the radial distortion, can be estimated along with the center of distortion from the epipolar constraint separately and before bundle adjustment without any calibration rig. The estimate converges as more image pairs are added. Descriptor matched scale-invariant feature (SIFT) point pairs that contain false matches can readily be given to our algorithm, EPOS (EpiPolar-based Solver), as input. The processing is automated to the point where EPOS solves the distortion whether its type is barrel or pincushion or reports if there is no need for correction.

**Keywords:** lens distortion; radial distortion; center of distortion; camera calibration; epipolar constraint; relative pose; fundamental matrix

## 1. Introduction

The bottleneck in turning every consumer-grade digital camera into a 3D scanner is in how to automatically self-calibrate the lens distortion from arbitrary image pairs. In contrast to capturing objects by encircling them with a camera, which is a well-studied problem (see, e.g., [1]), problems arise when capturing large-scale (indoor) environments with minimum image overlap in order to reduce the effort of image acquisition. Such a method, if “black-boxed”, would enable multiple applications, for example, in real-estate management and brokering (see, e.g., [2]).

Lens distortion is a combination of lens properties and imperfections in the camera lens manufacturing process [3]. It is well known that lens distortion changes with the focal length, which in turn is altered by both zoom and focus settings. The first is significant, while the latter can be omitted beyond focused distances of around 15-times nominal focal length [4,5]. For zoom lenses, the form of distortion may change from barrel at the wide-end to pincushion at the tele-end. In other words, the sign of the radial distortion parameters may change with zoom. Computationally, this may be dealt with, e.g., by using two separate models [6].

The image processing in photogrammetry, traditionally, consists of two steps [7–9]. First, the lens distortion is solved in a camera calibration together with other interior orientation parameters that are the focal length and the principal point, or the center of distortion. A checkerboard [10], other calibration targets [11] or robust features [8] are used to ensure accuracy. Second, (sparse) 3D reconstruction of the wanted scene is made using bundle adjustment, usually from different images than what was used in the first step. The other way, which is popular in computer vision, is to streamline these two steps into one by embedding a non-linear lens distortion model directly into the 3D reconstruction bundle adjustment [12–16]. This way, same images can be used throughout the

whole process, and some of the most important intrinsic camera parameters, such as the focal length, and the first two radial distortion coefficients can be recovered [14,17].

Despite the success of the one-step bundle adjustment, the inherent problem in the calculation of the reprojection error is that in addition to the lens distortion, also the external camera parameters and the parameters of the structure must be estimated. In other words, if one is first interested in only solving the lens distortion, the structure consisting of 3D points brings unwanted free parameters. This is especially problematic when attempting to reconstruct a large area with an uncalibrated camera, for example, in the built environment. Then, the camera network is loosely connected, i.e., it has weak geometry. Different unknown variables intertwine in bundle adjustment so that, for example, lens distortion deforms shapes in the object space [18]. Moreover, the effort of collecting data is unnecessarily increased by the requirement that all data must fulfill the redundancy requirement to solve lens distortion. Even if one prefers to perform a 3D reconstruction with minimal data, e.g., three images covering each point, it typically leads to having dozens or even hundreds of images. It would be beneficial if all of these available data could be used for the self-calibration of lens distortion. Therefore, it is evident that the lens distortion should be solved separately, before entering bundle adjustment.

The largest component of lens distortion is the radial component, in contrast to decentering and in-plane terms [19]. Radial lens distortion has an ordered correlation pattern, radial symmetry or  $r$ -symmetry, which can be detected, to some extent, with a blind approach for the purpose of removing this correlation [20], but in order to achieve further accuracy, more a priori knowledge is required. On the one hand, the epipolar constraint has been studied [21–25]. On the other hand, these studies typically rely on a single image pair, which as an approach contains the inherent need to control, not only noise, but also the scene properties in terms of appropriate feature detection, to avoid risks related to method instability. What has not been studied is utilizing multiple image pairs (that may also be from different scenes), which circumnavigates the problem by turning the question of stability, “whether an image pair is suitable for distortion detection”, into a question of convergence: “how many image pairs are needed to accurately detect the distortion”.

In this paper, we set out to exploit the  $r$ -symmetry in determining a global estimate for the radial distortion and, in order to do so, also construct a global constraint for the center of distortion that becomes more and more accurate as more image pairs are added. To our best knowledge, this has not been attempted before. We use uncalibrated images and present results for simulated data for multiple rectilinear lenses and zoom settings. In addition, real data containing false matches are used. The processing is automated to the point where our algorithm tells us whether the correction with the used model succeeds or if it is even needed.

The rest of this paper is organized as follows. In Section 2, the literature related to the lens distortion and the epipolar constraint is reviewed. In Section 3, as a continuation to this previous work, we show how epipolar constraints from different image pairs can be merged to estimate radial distortion. However, as this method depends on  $r$ -symmetry, it requires a good estimate for the center of distortion. Therefore, in Section 4, we introduce a new symmetry metric to obtain the estimate. Results are in Section 5, and the conclusion thereafter.

## 2. Related Work

Correcting the lens distortion before bundle adjustment (BA) reduces the amount of unknown parameters needed during it, thus increasing the robustness of the BA solution. Simultaneously, this paradigm allows an automated black-box-type use of the correction method; see the bottom arrow of Figure 1. For these reasons, it is meaningful to study what parts of distortion can be solved already with the singular image-pair correlation matrix, i.e., the fundamental matrix. The first attempt by Zhang [21] was to expand the epipolar constraint with a distortion function so that

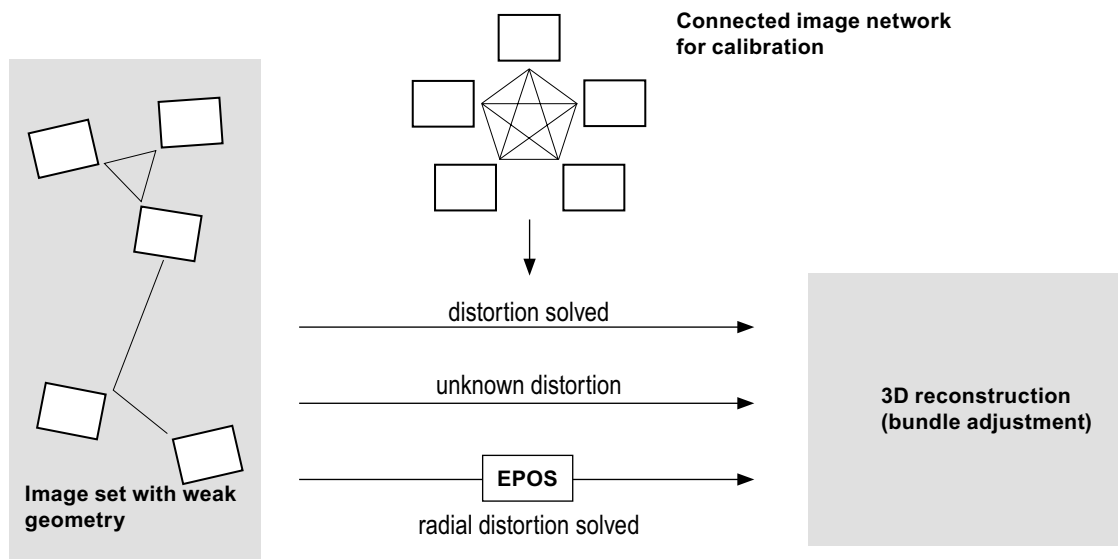
$$\tilde{u}^T F \tilde{u}' = 0, \quad (1)$$

where  $\tilde{u}^T = [\tilde{x}, \tilde{y}, 1]$  and  $\tilde{u}'^T = [\tilde{x}', \tilde{y}', 1]$  denote a pair of matching undistorted keypoints on two separate images, and  $F$  is the obtained fundamental matrix. The keypoints, in turn, are obtained from observations  $u^T = [x, y]$  and  $u'^T = [x', y']$  by using a (radial) distortion correction

$$\begin{cases} \tilde{x} = xD(r) \\ \tilde{y} = yD(r) \end{cases} \quad (2)$$

$$D(r) = 1 + k_1 r^2, \quad (3)$$

where  $r^2 = (x - x_p)^2 + (y - y_p)^2$ . This first attempt was flawed in two ways. First, rigorously, the distortion center  $p = (x_p, y_p)$  is defined by the distortion model [26], meaning that for each lens distortion model, there exists a unique center of distortion specific for that model. However, in the work by Zhang [21], it was assumed to reside at the center of the image.



**Figure 1.** The proposed method, and its implementation EPOS, introduce a new automated ‘black-box’ paradigm in solving lens distortion for image sets with weak geometry, compared to the existing alternatives of either using a pre-calibrated camera (top arrow) or employing a non-linear distortion correction term inside bundle adjustment (middle arrow).

Second, the attempt by Zhang [21] and that of Fitzgibbon [22] relied on the so-called nine-point method for solving the fundamental matrix, which does not enforce the rank-two constraint. However, even in the studies by Kukulova and Pajdla [23] and Liu and Fang [24], where the rank-two constraint was enforced, the center of distortion was still assumed to be at the center of the image, i.e., the first flaw still persisted. We shall return to these issues in the next section.

Successful attempts to determine the location of the center of distortion have been made, although these are limited to specific conditions. Hartley and Kang [27] were able to construct a parameter-free distortion model with a center of distortion estimator by using a calibration rig, but their model, although theoretically sound, could not handle normal in situ noise without the rig. Brito et al. [25] used the curvature of epipolar lines to determine the location of the center of distortion, but this method’s success was limited to images with curvilinear lens-type distortion, i.e., very large distortion, due to problems in detecting differences in small curvatures. In addition, one-image pair methods typically experience vulnerabilities with respect to noise. According to Brito et al. [25] “with more than one pixel noise the given implementation produces very large errors”.

With multiple image pairs, however, these vulnerabilities related to noise, in addition to those related to false matches and inadequately distributed features, do not turn into a question of stability,

but rather, into a question of convergence. All data, even from different scenes, can be used to converge the result. Therefore, the convergence of the radial distortion coefficient as a function of the number of inlier point pairs, or image pairs—since the amount of points per image is typically fixed by the natural properties of the scene—is of major importance.

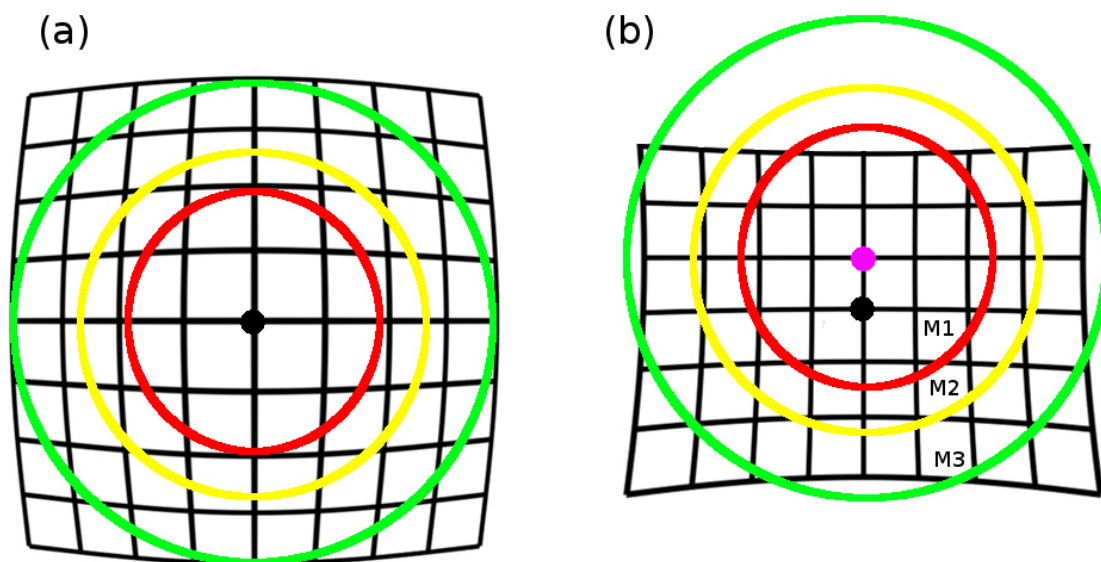
### 3. Iterative Solution for Radial Distortion

To order to determine the lens distortion using multiple image pairs, we may extend Equation (1) as

$$\min_{k_1} \left| \sum_{\text{image pairs}} \sum_{\text{point pairs}} \left| \tilde{u}^T F \tilde{u}' \right| \right| = \epsilon, \quad (4)$$

where vertical bars denote the  $L_1$  norm. At the limit of no noise,  $\epsilon \rightarrow 0$  with an ideal distortion model. Despite its theoretical simplicity, Equation (4) typically contains difficult terms that follow from camera geometry, noise and false matches. We shall return to discuss these later.

Zhang's idea of optimizing  $F$  and  $k_1$  in consecutive loops, so that one variable is estimated and the other is kept fixed, and then vice versa until both converge separately, is bright, but not numerically stable for one pair of images [21]. However, we argue that this idea may well work for multiple image pairs, especially if the previous two flaws are overcome, i.e., the center of distortion is properly estimated (see Figure 2), and  $F$ 's rank-two constraint is enforced. Considering the first, we can then exploit  $r$ -symmetry properly. Considering the latter, the more data we have available, the more likely a larger area of the image plane is covered by the matching point pairs. Hence,  $k_1$  cannot get in an artificial numerical coupling (caused by finite data) with a single  $F$ . Furthermore, as the amount of data grows, the statistical error in  $k_1$  diminishes. This is, descriptively, something solid against which  $k_1$  can converge.



**Figure 2.** Illustrations of radial symmetry; see Equation (5). (a) Barrel distortion,  $\eta < 0$ ; (b) pincushion distortion with a (magenta) center of distortion different from the (black) center of the image. Symmetry is employed while solving the fundamental matrices  $F_k$  and the radial correction parameter of Equation (9) consecutively. Here, each iteration increases the amount of inliers,  $M_1 < M_2 < M_3 < \dots$ , until the convergence of Equation (11) is reached, and  $\eta > 0$  is obtained.

Therefore, we make  $k_1$  and all  $F_k$  to converge in consecutive loops. In the following notation, we use image coordinates that are normalized with  $a = w/4$  to avoid  $F$  matrix ill-conditioning, where  $w$  is the image width in pixels. The symbol  $\eta$  is used to mark the counterpart of  $k_1$ , to distinguish these two from each other. To obtain dimensionless quantities for comparison, we note that the distance from the center of distortion  $r = a\hat{r}$ , and  $\eta = \hat{\eta}/a^2$ , where the ‘hat’ symbol stands for these dimensionless quantities.

Given a proposed center of distortion  $p_{pr}$  and a radial distortion model

$$D(r) \equiv 1 + \eta r^2, \quad (5)$$

the distortion coefficient  $\eta$  is computed as follows.

The fundamental matrices  $F_k$  and the respective inliers are computed from an initial set of point pairs  $\{(u, u')_i\}$  using RANSAC and the proper rank-two constraint for  $F_k$  (these initial observations are replaced on later iteration rounds with the corrected points  $\{(\tilde{u}, \tilde{u}')_i\}$ , which is explained later). Image pairs are indexed with  $k$ , point pairs with  $i$ , and the inlier mask  $M$  indicates whether a point pair is an inlier or not,  $M_k(i) = 1$  or  $= 0$ , respectively. To prevent systematic errors in  $F_k$  computation, we require that at least 15 point pairs must be inliers, or otherwise, the corresponding image pair is omitted. This is similar to the amount presented in the work of Hartley and Kang [27], who theoretically determined that 80 points in a connected network of four images is needed to estimate (radial) distortion. In  $F_k$  calculation, we employed the OpenCV implementation of RANSAC [28] with a tolerance distance of three pixels. Different confidence estimate values are tested in the Results Section with real data.

For each inlier point pair, we consider the point with the largest  $r$ , regardless on which of the two images it lies and, with a little abuse of notation, denote it by  $u$ . In other words, the distortion on the other image is yet neglected. For each  $u$ , extending its (radial) vector  $\vec{u}$  from the proposed center of distortion  $p_{pr}$  onto the epipolar line gives us the undistorted keypoint vector

$$\vec{\tilde{u}} = \vec{u} + \frac{|\vec{e} - \vec{u}|}{\cos \alpha} \hat{u}, \quad (6)$$

where the angle  $\alpha$  between  $\vec{u}$  and the line perpendicular to the epipolar line is obtained as

$$\cos \alpha = \frac{\vec{u} \cdot (\vec{e} - \vec{u})}{|\vec{u}| |\vec{e} - \vec{u}|}. \quad (7)$$

In Equation (6),  $\hat{u} = \vec{u}/|u|$  denotes a unit vector, and  $\vec{e}$  is a vector from  $p_{pr}$  to point  $e$  that is the closest point of  $u$  on the epipolar line; see Figure 3.

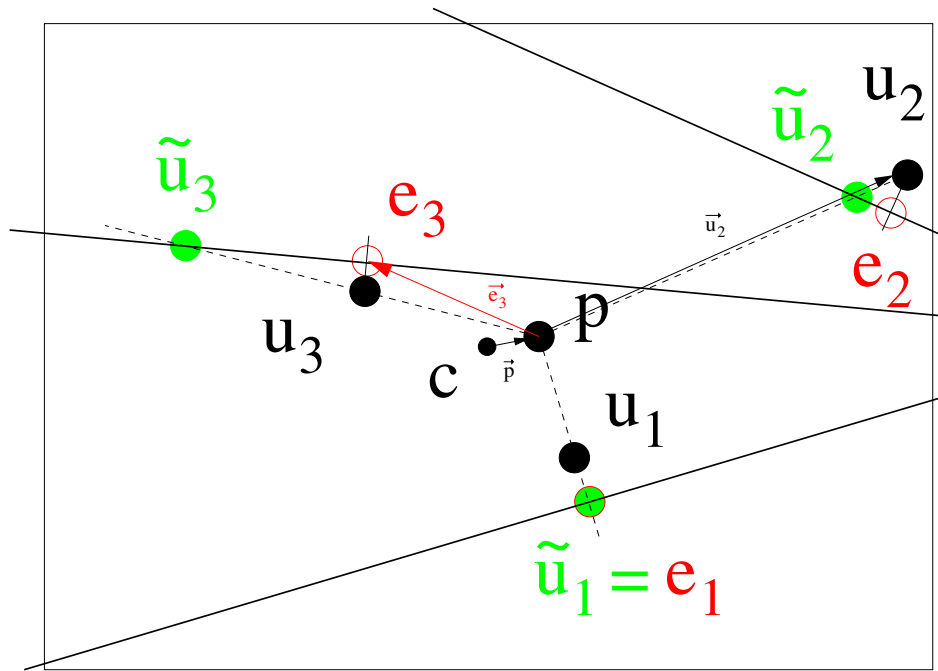
From Equation (5), we have

$$\tilde{r} = r(1 + \eta_i r^2) \iff \eta_i = \frac{\tilde{r} - r}{r^3}, \quad (8)$$

where  $\tilde{r} \equiv |\vec{\tilde{u}}|$ . Specifically, one trial value  $\eta_i$  is obtained for each point pair  $i$ . The radial distortion coefficient is obtained by taking a logarithmic average

$$\eta = \exp \left[ \frac{1}{N^*} \sum_i \alpha_i \log |\eta_i| \right], \quad (9)$$

where  $N^* = \sum \alpha_i$ , and the weights  $\alpha_i$  are either zero or one, so that only one third of the terms with the lowest absolute value  $|\eta_i|$  are retained. Then, if the majority of all of the remaining  $\eta_i$  were negative,  $\eta$  is also declared negative.



**Figure 3.** Properties for observations  $\{u_i\}$  and their respective epipolar lines (black solid lines). The radial correction function works best when the radial vector is perpendicular to its respective epipolar line, such as  $\tilde{u}_1$ , pointing from  $p$  to  $u_1$ , because the geometry is then optimal. Then, the intersection of the radial vector and the epipolar line is, at the same time, the closest point on the line,  $\tilde{u}_1 = e_1$ . In contrast, the radial vector of point  $u_3$  is almost parallel to its respective epipolar line. Here, the function is bound to overestimate the correction, even if the closest point on the line  $e_3$  is close to  $u_3$ . The most typical case  $u_2$  is in between the previous two. The center of distortion  $p$  differs from the center of the image  $c = (w/2, h/2)$ .

In selecting the terms within Equation (9), one third is a rough estimate based on the geometry that effectively circumnavigates the trouble caused by epipolar lines being close to parallel with radial lines (see Figure 3), which produces overly large values for  $\eta_i$ . Moreover, this offers robustness with respect to noise and false matches present in situ data, in contrast to the minimization criterion of Equation (4).

After solving  $\eta$  from Equation (9), the original uncorrected points are corrected with it, namely

$$\{(\tilde{u}, \tilde{u}')_i\} = D(\eta) \{(u, u')_i\}. \quad (10)$$

Here, the correction is applied to both (or all) images. However, due to the fact that the distortion on the other image was neglected before, the strength of the correction is prone to be underestimated. Because of this, when new fundamental matrices  $F_k$  are obtained using corrected point pairs  $\{(\tilde{u}, \tilde{u}')_i\}$  from Equation (10), the inlier mask  $M_k$  computation may result in some of the inlier point pairs still being treated as outliers. Therefore, the process of solving new  $\eta$  and new  $F_k$  needs to be repeated iteratively until convergence, which is reached when  $\eta$  is no longer underestimated. This, in turn, is seen from the fact that the amount of inliers in  $M_k$  stops growing.

We summarize the convergence condition, i.e., when the iteration loop is exited. Original uncorrected points are corrected as in Equation (10) using the latest  $\eta$  of Equation (9) on each iteration round. New  $\{F_k\}$  and their respective new inlier masks  $\{M_k\}$  are computed based on these. Specifically,  $M_k(i) = 1$  or  $= 0$ , if the point pair  $i$  is correlated, or is not correlated, via  $F_k$ , respectively. Hence, as the value of  $\eta$  becomes less and less underestimated, the inlier set of corrected points grows on each iteration round  $d \in \mathbb{N}$ , notation  $M^d$ , as  $M^1 \{(\tilde{u}, \tilde{u}')_i\} \subseteq M^2 \{(\tilde{u}, \tilde{u}')_i\} \subseteq \dots \subseteq \{(\tilde{u}, \tilde{u}')_i\}$ , and when it stops growing, we have the convergence condition, namely



$$M^d \{(\tilde{u}, \tilde{u}')_i\} \equiv M^{d+1} \{(\tilde{u}, \tilde{u}')_i\}, d \in \mathbb{N}. \quad (11)$$

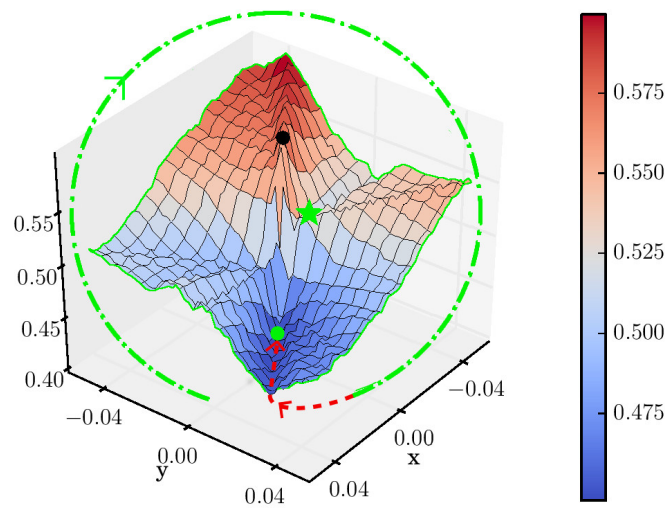
See Figure 2b for an illustration of growing  $M^d$ . The algorithm pipeline is outlined in Algorithm 1. Next, we introduce how to obtain a good estimate for  $p_{pr}$ , which is required by Equation (6).

---

**Algorithm 1** EPOS algorithm.

---

1. Extract initial point pairs from images
  2. Find a proposition for the center of distortion (Section 4).
    - (a) Determine inliers by calculating fundamental matrices.
    - (b) Circle around the image, and use the inliers to find the bottom of the ‘symmetry valley’. See Figure 4.
    - (c) Walk along the bottom of the valley.
    - (d) Stop at the high gradient. Obtain  $p_{pr}$  from here.
  3. Given  $p_{pr}$ , determine radial distortion (Section 3).
    - (a) Compute fundamental matrices  $F_k$  and inlier masks  $M_k$  from  $\{(\tilde{u}, \tilde{u}')_i\}$ , for all image pairs  $k$ .
    - (b) Exit if the inlier amount in  $M_k$  did not grow (not on the first round).
    - (c) Obtain the distortion coefficient  $\eta$  from Equation (9).
    - (d) Correct initial  $\{(u, u')_i\}$  by plugging  $\eta$  in Equation (10) to obtain updated  $\{(\tilde{u}, \tilde{u}')_i\}$ .
    - (e) Go to 3 (a).
  4. Locally search for better  $p_{pr}$  using 3. Choose the best result according to the inlier percentage.
- 



**Figure 4.** Energy surface depicting the symmetry ratio  $r_s = r_s(p_{pr})$ , where the x- and y-axes span the location of  $p_{pr}$ . A low/blue (high/red) ratio indicates a favorable (unfavorable) position for  $p_{pr}$  in terms of barrel distortion symmetry. The surface is non-convex. The best trial for the center of distortion  $p_{pr}$  is found at the bottom of the valley. The EPOS result is shown with a big (green) dot in the deepest valley. The (black) dot on the ridge represents its mirror location. The reference result from bundle adjustment (BA) is shown with a big star in an upper valley. Green and red dotted lines represent how the scouting of the symmetry landscape is done; see Section 4.2.

#### 4. Finding the Center of Distortion

The radial distortion model of Equation (5) has an obvious one-dimensional symmetry, which we will use in introducing a new way to divide the two-dimensional problem of finding  $p = (x_p, y_p)$  in an easier form of two one-dimensional problems.

##### 4.1. Symmetry Ratio Measure

To employ  $r$ -symmetry successfully in Section 3, the quality of a proposed center of distortion  $p_{pr}$  should be measured solely in terms of whether it provides good  $r$ -symmetry. This is in contrast with the methods that rely on minimizing some Euclidean distance, BA for instance. We propose a weighted binary-form symmetry metric that is based on the notion on which side of its respective epipolar line a point lies. If a point is on the same side as the symmetry center, it carries a positive weight; otherwise, its weight is zero. Formally

$$\Theta(p_{pr}) = \begin{cases} w, & \text{if } r_u^2 \leq r_e^2 \\ 0, & \text{if } r_u^2 > r_e^2 \end{cases} \quad (12)$$

where  $r_u^2 = |u - p_{pr}|^2$  is the squared distance from the proposed center of distortion  $p_{pr}$  to the keypoint  $u$ , and  $r_e^2 = |e - p_{pr}|^2$  is the squared distance from  $p_{pr}$  to the point  $e$ . See Figure 3. Point  $e$  resides on epipolar line  $u'^T F$  and is the closest point to  $u$ . The weight  $w \in (0, 1]$  is the normalized dot product of  $r_u$  and the vector from the keypoint  $u$  to the point  $e$ . Hence, point pairs with close to parallel vectors (“good geometry”) have a greater impact than pairs with close to orthogonal vectors (“bad geometry”).

Barrel distortion manifests in observations lying (ideally: only) on the far side of the epipolar line, farther away from the center than they should be. In an ideal case, the symmetry metric of Equation (12) is equivalent for all point pairs  $i$ , i.e.,  $\Theta_i \equiv 0$ , meaning barrel distortion, or  $\Theta_i = w \forall i$ , meaning pincushion distortion.

In practice, however, exceptions occur due to noise and the properties of the used distortion model, for instance. To illustrate non-ideal symmetry, three feature points are shown in Figure 3, with  $\Theta = 1, 0$  and  $\approx 0.2$ , for  $u_1, u_2$  and  $u_3$ , respectively. Thus, the symmetry measure for  $p_{pr}$ , for a set of image pairs, can be written as

$$r_s(p_{pr}) = \sum_{\text{image pairs}} \frac{1}{W_k} \sum_{i=1}^{N_k} \Theta_i, \quad (13)$$

where  $N_k$  is the total amount of point pairs for the image pair  $k$ , and  $W_k = 1/N_k \sum_{i=1}^{N_k} w_i$  is the respective total weight, including also pairs with  $r_u^2 > r_e^2$ .

The best  $r$ -symmetry is found at the point  $p_{pr}$  that minimizes (or maximizes) Equation (13), namely

$$\arg \min_{p_{pr}} r_s(p_{pr}). \quad (14)$$

One key property of the  $r_s$  measure of Equation (13) is that it can be computed quickly. It can be thought of as a symmetry ratio, as the value it can obtain is limited in the range  $[0, 1]$ ; see Figure 4. For barrel distortion, for which  $\eta < 0$ , a (blue) valley is formed around the spot where the best symmetry, i.e., the best  $p_{pr}$ , resides. In contrast, a (red) ridge rises to mark non-optimal symmetry. For pincushion distortion, for which  $\eta > 0$ , the ridge offers the best symmetry and valley the worst. The proposed center of distortion  $p_{pr}$  is moved, as explained in Section 4.2, until the condition of Equation (14), i.e., the bottom of the valley (or the top of the ridge), is reached.



## 4.2. Symmetry Landscape

The center of distortion  $p = (x_p, y_p)$  is a two-variable unknown. The most important property of the symmetry measure of Equation (13) is that it allows us to separately solve these variables, when the problem is cast into polar coordinates. In particular, since both the ridge and the valley extend to infinity by the definition of Equation (12), their orientation can be detected from almost any perimeter drawn around the center of the image. The correct radial positioning along the bottom of the valley can then be considered. We thus proceed in two steps.

The orientation is solved first by looking for the bottom of the valley; see the green circular arrow shown in Figure 4. Specifically, a sufficiently large square of 0.25 side length that can be said to capture all centers of distortions is drawn around the image center, and the symmetry measure  $r_s$  of Equation (13) is computed on discrete points  $\{p_m\}$  that reside at the perimeter and are spaced 0.004 apart from each other (in  $w/4$  units). In other words, solving the whole 2D landscape is not necessary, since we can only compute the 1D perimeter. After obtaining each  $r_s(p_m)$ , we calculate a weighted orientation vector

$$v = \sum_m p_m r_s,$$

and norm it  $\hat{v} = \frac{v}{|v|}$ , in order to get the orientation of the valley  $\hat{v}$ . In practice, local averages for perimeter points are calculated using six nearest neighbors. The perimeter point with the lowest value is at the bottom of the symmetry valley and indicates its orientation.

Second, we solve the radius by walking along the bottom of the valley, starting from the perimeter, illustrated by the red arrow in Figure 4. Using the same discrete step size of 0.004,  $r_s$  is computed along a straight path to the origin. The gradient  $\nabla r_s = \frac{dr_s}{dr}$  is computed using finite differences. The walk is halted at the point where the gradient diverts from its average by more than one standard deviation,  $|\nabla r_s(r) - \text{avg}(\nabla r_s)| > \sigma$ . Hence, the radius is solved, and  $p_{pr}$  is obtained from there.

Finally, there is no knowledge on whether the distortion that we are trying to solve should be of the type barrel or pincushion. Therefore, both the spot  $p_{pr}$  and its mirror position on the other side of the image center must be evaluated. These two answers are then processed, and the one yielding more inliers is selected and ultimately tested against the convergence condition of Equation (11). Hence, only one model is required in contrast to using two models [6].

## 4.3. Local Search

For an initial  $p_{pr}$  obtained from Section 4.2, a local search algorithm was used to pinpoint the best location for the center of distortion. Trial steps with a discrete step size of 0.002 were made in four directions, two perpendicular and two parallel to the radius. If the amount of inliers obtained through the solution of Section 3 remained the same, the step size was increased by 10%. The walk was terminated when no further improvement in terms of inliers was found. Experimentally, we found that the local search improved the result in terms of  $\eta$ , although the local walks were short.

## 4.4. Implementation of the Proposed Method: EPOS

Our program EPOS (EpiPolar-based Solver) is built so that it can automatically know whether it has actually managed to correct the lens distortion or whether it has failed to do so. This is done by simply comparing the amount of point pair inliers obtained with and without correction, using Algorithm 1. Distortion correction is considered successful if the amount of inliers has increased. In practice, we have two non-zero values of  $\eta$ , one for the minimum and one for the maximum of the symmetry measure (see Equation (14) and Figure 4), and the comparison is done also between these two; and against the no correction case.

The robustness of the implementation is built on a three-fold foundation. First, the computation of the fundamental matrix  $F$  enforces the rank-two constraint. Second, the symmetry metric to estimate the center of distortion is based on a majority voting that, in contrast to Euclidean-based measures,

does not contain overly large, nor “diverging” terms arising from noise and false matches. Third, the radial distortion coefficient is evaluated from the most conservative third of point pairs, which circumnavigates the troublesome geometry when the epipolar lines are parallel to the radial lines.

Descriptor matched SIFT [29] features, such as those that are used for bundler input [30], can also be given to EPOS as input. False matches are handled in fundamental matrix computation as described in Section 3. Currently, the method’s convergence may be stopped if the  $F$  computation fails so that totally wrong  $F$ ’s are present in the input. This is, however, a known problem for which solutions exist and beyond the scope of this paper. Considering the future work, one promising approach to decrease computation time and to further increase the method’s robustness with respect to  $F$  computation stability is to use graph-based point tracking [31] that minimizes the amount of initial data by reducing false matches from it.

#### 4.5. Extension to Non-Rectilinear Lenses

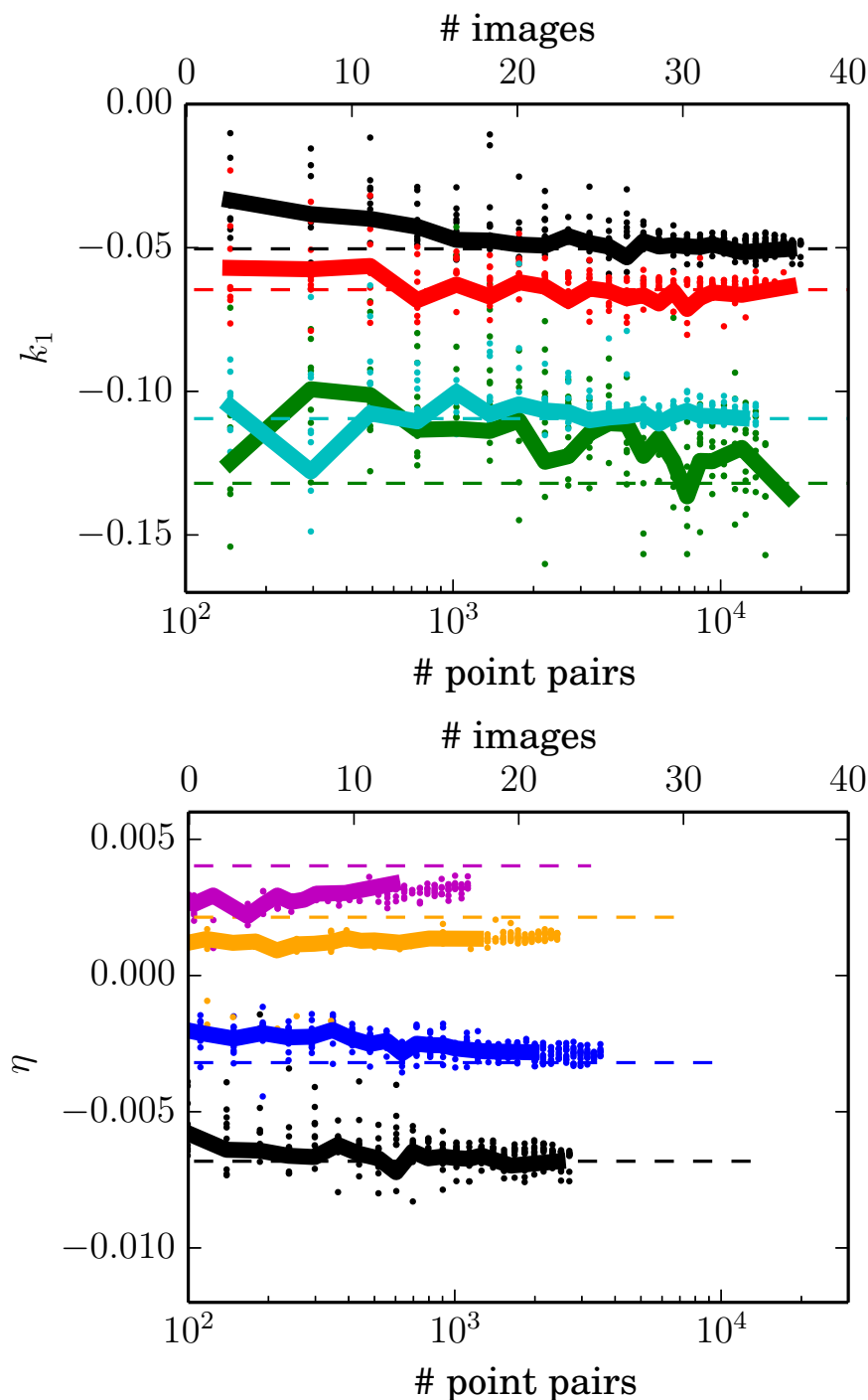
Brown’s [3] thin lens approximation, derived as a Taylor approximation from the analytical ray tracing formula, holds only for weak distortion, cf. Equation (5). Some digital cameras, however, employ a fish-eye lens instead of a rectilinear (thin) lens. The fish-eye lens, alongside having a different distortion model than its rectilinear counterpart, differs also by not having a uniquely-defined center of projection, which may break down the whole pin-hole camera model if this effect is significant ([32], p. 242). If not, the fish-eye lens data may be corrected with EPOS if the mapping used by the lens model, stereographic for example, is known, and the images are first transformed into the rectilinear presentation. However, experiments with fish-eye lenses are beyond the scope of this work.

### 5. Results and Discussion

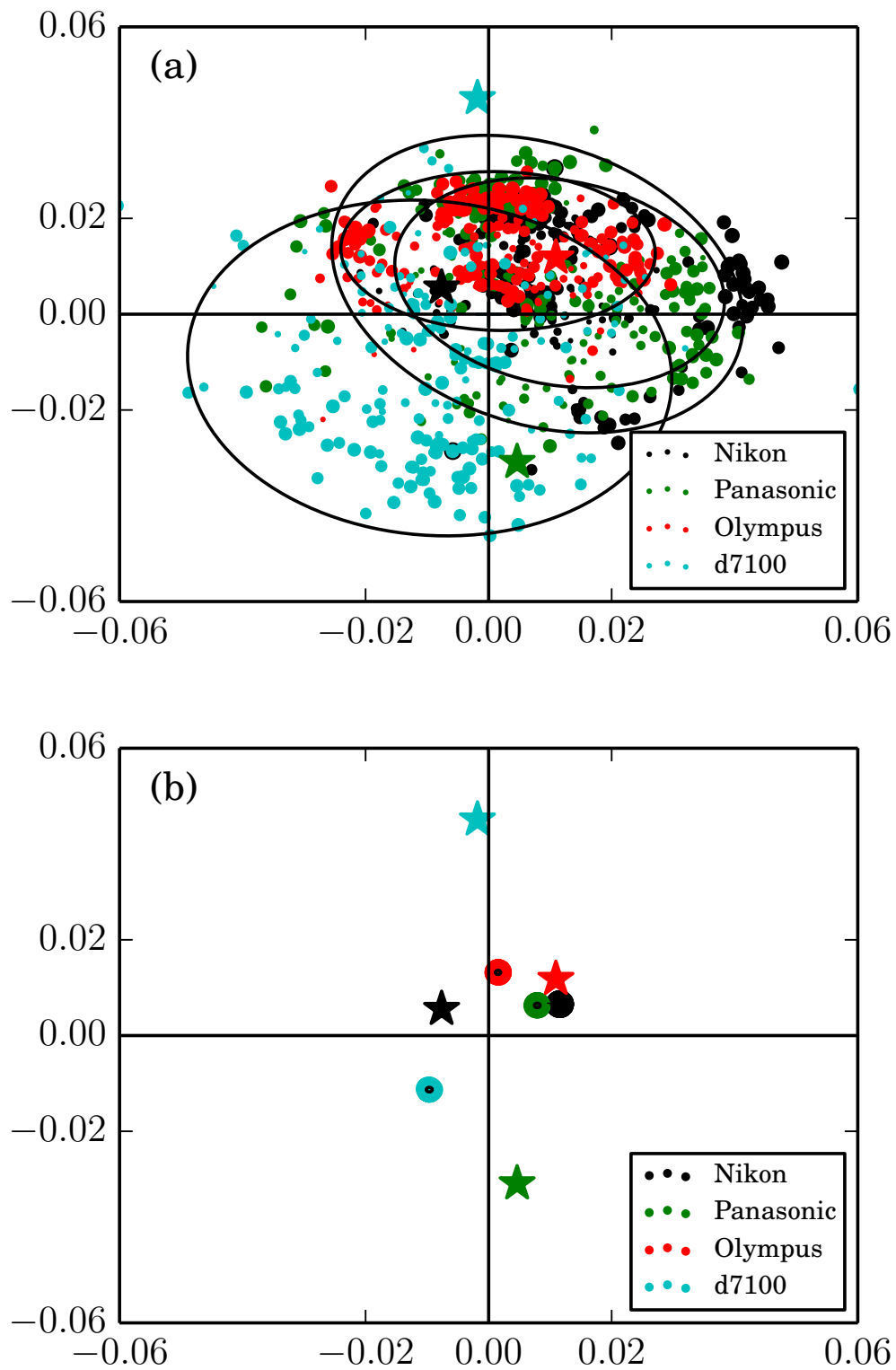
Methods relying on just one image pair are vulnerable to bad geometry and noise as already discussed in Section 2. With multiple image pairs, we speak of convergence and use a checkerboard to obtain simulated data for a thorough testing of the properties of EPOS. Choosing simulated data this way allows us to explore the relevant range for the distortion parameter that is used for off-the-shelf digital cameras that have rectilinear lenses.

#### 5.1. Simulated Data

The convergence of  $\eta$  as a function of the number of inlier point pairs (the amount of points per image is fixed here to 49 due to the checkerboard) is shown in Figure 5, with a numerical summary in Table 1. For each image set size, ten results are computed to study the statistical behavior of the algorithm. These are obtained by drawing random subsets of image pairs from the checkerboard set. In Figure 5, the thick line follows the best inlier percentage and is drawn to guide the eye. Expectedly, yet remarkably, adding more data yields a better result as  $\eta$  converges to the reference result for different lenses. The impact of unstable interior orientation is well visualized, as the inexpensive zoomable 18–105-mm lens attached to the Panasonic DMC yields the slowest convergence. The same keypoint observations are used for testing EPOS and for obtaining reference results. Each image block is treated separately with respect to the lens model and zoom setting. Results for the center of distortion, shown in Figure 6, are discussed briefly. A sample checkerboard image is shown in Figure 7. Reference results were computed with the Caltech Camera Calibration Toolbox [10]. Reference values for  $k_1$  are in pixels per focal length units, while EPOS  $\eta$  values are in the chosen pixel per  $w/4$  units. For result comparison,  $\eta$  results are transformed into  $k_1$  units in Figure 5 and Table 1 by using a multiplication factor  $\gamma^2 = (f/a)^2$ , where  $f$  is the focal length from reference bundle adjustment, and  $a = w/4$  is the chosen unit scale in EPOS.



**Figure 5.** Convergence of the value of  $\eta$  as a function of the number of inlier point pairs (and number of images). Note that as the checkerboard images form a connected network, a set of five or 20 images equals about 490 or 9310 point pairs, respectively. In other words, the amount of point pairs grows exponentially with respect to the amount of images. The dashed lines represent reference results onto which the results from EPOS (solid lines) converge. The previous, in top to bottom order, are (a) Nikon D700 24 mm, Olympus 14 mm, Nikon D7100 18 mm, Panasonic DMC 6 mm; (b) results for four different zoom settings, 24 mm, 28 mm, 35 mm and 50 mm, for Nikon D700 (bottom to top order). Note the different units for  $\eta$  and  $k_1$ ; see the text for details.



**Figure 6.** (a) Center of distortion results from EPOS are shown as dots, while reference results are marked as stars. Ellipses describe sample variances; (b) The same data displayed through weighted sample averages with reference results. See the text for details.



**Figure 7.** Results from EPOS. Distorted (left) and undistorted (right) images for D700 24 mm. The checkerboard image (top) is corrected using the converged value of  $\eta$  obtained from checkerboard data; see Figure 5. The image colors were simplified to black and white for visual reasons. Sample images of a building called Dipoli (second from top) and a sofa set (third from top) are corrected with values obtained from the respective sets. Bookshelf images taken with the Panasonic 6 mm lens are also corrected with the value obtained in situ. Dipoli set contains 11 photos, the sofa set 9 photos and the bookshelf set 5 photos. Real-world cases contain false matches and were chosen to be quite difficult.



**Table 1.** Summary of the results shown in Figure 5. Averages  $\langle k_1 \rangle$  and  $\langle \eta \rangle$  are taken over the data for which # images  $\geq 20$ , except for 50 mm over the data, for which # images  $\geq 10$ .

| Data Set Figure 5a | $\langle k_1 \rangle$  | Std dev (EPOS) | Reference $k_1$  |
|--------------------|------------------------|----------------|------------------|
| Nikon D700 24 mm   | −0.04868               | 0.00273        | −0.05038         |
| Olympus 14 mm      | −0.06319               | 0.00271        | −0.06460         |
| Nikon D7100 18 mm  | −0.10748               | 0.00257        | −0.10949         |
| Panasonic DMC 6 mm | −0.12107               | 0.01227        | −0.13201         |
| Dataset Figure 5b  | $\langle \eta \rangle$ | Std dev (EPOS) | Reference $\eta$ |
| 24 mm              | −0.00659               | 0.00037        | −0.00681         |
| 28 mm              | −0.00282               | 0.00025        | −0.00319         |
| 35 mm              | 0.00139                | 0.00015        | 0.00213          |
| 50 mm              | 0.00308                | 0.00022        | 0.00402          |

In order to verify how well EPOS handles the change of sign in  $\eta$ , we computed results for four different zoom values 24 mm, 28 mm, 35 mm and 50 mm, using a camera with stable interior orientation, Nikon D700 with a 24–70 mm objective; see Figure 5b. Convergence is achieved quickly, mostly because of the quality of the objective. At the limit of very small distortions and with a limited set of point pairs, there is a possibility that the set of point pairs is determined to all be inliers by RANSAC before EPOS enters the iterative solution loop. Then, EPOS decides that no improvement is required. Otherwise, the result converges at the limit, where the amount of point pairs is large.

In Figure 6a, the results for the center of distortion are shown for various image block sizes. The size of the point represents the amount of data used to calculate that point. All samples are weighted with inlier percentages, and these weighted averages are shown in Figure 6b. Again, the amount of inliers for each result is determined by the convergence condition of Equation (11). The fact that the recovered center of distortion is different from the reference result is noteworthy and follows from that these center of distortions actually have different definitions (remember the work of [26]). Rigorously put in other words, the proposed  $r$ -symmetry center is to be used only for lens distortion correction and not for any 3D measurements done afterwards, for which the principal point is the correct choice.

After obtaining  $\eta$ , the importance of the center of distortion diminishes. In order to see this, we undistorted the checkerboard images with one fixed value of  $\eta$ , but employed three different center of distortion values, which were the image center, the reference result and the result obtained from EPOS. Then, for each of these three undistorted image sets, we checked with the Caltech Calibration Toolbox that they all contained an essentially equivalent (and small) image distortion. Hence, even if the center of distortion plays a crucial role in solving the radial distortion parameter, as discussed in Section 4, its significance at the image correction phase is negligible, at least for the used distortion model.

The symmetry ratio plot in Figure 4 sheds light on the different centers of distortions shown in Figure 6b. The result from EPOS resides on the bottom of the deepest valley. The reference center of distortion resides at the bottom of another valley. In other words, multiple valleys and ridges may be present. This non-convex nature of the landscape highlights the importance of the ‘light-weight’ method of Section 4.2 to find the center of distortion efficiently. The data for Figure 4 were obtained from a set of 10 images, yielding 45 image pairs with a total of 2200 point pairs. For a smaller set of point pairs, the symmetry ratio surface becomes noisy, which also affects the convergence rate of  $\eta$  shown in Figure 5.

## 5.2. Real Data from In Situ Images

In order to validate EPOS beyond the simulated data that do not contain false matches, we captured images from built environment (or man-made environment). The Aalto University building named Dipoli and a set of sofas with challenging lighting and low-texture surfaces were shot with



a hand-held D700 24 mm, resulting in sets of 10 and nine images, respectively, which were then resized to  $532 \times 354$  pixels. In other words, we chose the lens with the smallest radial distortion and, in addition, notably reduce the image size to further challenge the proposed method. As the images were small, the  $F$  computation tolerance distance was reduced from three to two pixels. Using descriptor-matched SIFT feature coordinates, point pair data were given to EPOS as input. False matches and bad camera network geometry are both present. Results are summarized in Table 2 and the corresponding corrections of the images samples shown in Figure 7.

**Table 2.** EPOS results with real data using a confidence value of 0.99 are compared against the state-of-the-art VisualSFM (VSFM) software.

| Data Set  | # Images | VSFM # | VSFM $f$ | Reference $f$ | Ref. $\eta$ | EPOS $\eta$   | VSFM $\eta$ |
|-----------|----------|--------|----------|---------------|-------------|---------------|-------------|
| Dipoli    | 10       | 10     | 372.2    | 361.5         | −0.0068     | −0.0052 ± 5%  | (−)0.0048   |
| Sofa      | 9        | 2      | 354.6    | 361.5         | −0.0068     | −0.0071 ± 15% | N/A         |
| Combined  | 3 + 3    | 3      | 602.0    | 361.5         | −0.0068     | −0.0071 ± 10% | N/A         |
| Bookshelf | 5        | 5      | 803.6    | 821.7         | −0.0072     | −0.0050 ± 4%  | (−)0.00039  |

We study the impact of the confidence level parameter in fundamental matrix calculation. Obviously, a high confidence level requirement reduces the risk that the epipoles are displaced, but we also test the numerical precision of the method with respect to this parameter. For the Dipoli set, we obtain  $\eta = -0.0052$  with the confidence value of 0.99. With a lower confidence value of 0.95, EPOS occasionally fails to present a result due to problems in  $F$ -matrix calculation. With a confidence value of 0.9, most calculations fail producing a result of  $\eta = 0$ , except for sets of six and around 14 image pairs that yield  $\eta = -0.0052 \pm 5\%$ . For the sofa set, SIFT matching yields a total of 1568 initial point pairs, from which EPOS finds about 1250 inliers without radial distortion correction. With correction, EPOS finds about 1270 inliers, leaving 19% of the point pairs as outliers. The difference in inlier amount is quite small between these two, so we tested the numerical stability of the result by using different confidence values in the fundamental matrix calculation. For confidence values of 0.90, 0.95 and 0.99, we obtain −0.0087, −0.0067 and −0.0071. Relative error for  $\eta$  is estimated here to reside within 15%. We emphasize that these are very challenging sets of data.

With in situ images, typically a minimum of four image pairs is required to obtain the first decent result for  $\eta$ . In order to see if EPOS can truly handle image pairs from different scenes, we create a combined set from the previous Dipoli and sofa sets, taking the first three images from each. After  $F$  matrix calculation, this yields five image pairs. We obtain  $\eta = -0.0071$ , which is remarkably close to the ground truth −0.0068. The confidence values of 0.95 and 0.99 yielded this same result. The difference in results with this and the pure Dipoli set may be explained by the fact that features in Dipoli images are mainly on the center of the image, whereas the accurate recovery of the radial distortion requires also features near the image boundaries (as in the sofa set).

Images taken with the lens having the largest distortion, Panasonic DMC FZ8 6 mm, were reduced to  $768 \times 576$  resolution, and the obtained SIFT matches were given to EPOS. With only five bookshelf images (equaling 10 image pairs), we obtain  $\eta = -0.0050$  for both 0.95 and 0.99 confidence values; see Figure 7. The result stays within  $\pm 4\%$  even if the amount of point pairs is increased from that of four image pairs (670, 590, 1690) to that of 10 image pairs (1858, 1697, 4939), indicating pairs after correction, before correction and after SIFT match. Note that 62% of the point pairs are outliers. Even for a 0.90 confidence value, a value of  $\eta = -0.0051$  is ultimately obtained. The value of  $\eta$  obtained from these in situ images is quite close to the reference result  $\eta = -0.0072$  (shown in different units  $k_1 = -0.1320$  in Figure 5), and it leads to a rather good visual undistortion result; see the green lines in Figure 7.

In Table 2, our results are compared against a state-of-the-art BA-based open method, VisualSFM (VSFM) [33], which paradigmatically relates to the middle horizontal arrow in Figure 1. Similar to EPOS, VSFM employs only one radial distortion coefficient to estimate lens distortion. The methods

perform equally well for the Dipoli set, where a good image network may be formed. On the bookshelf set, the VSFM estimate for  $\eta$  misses with a decade, while the accuracy of EPOS remains the same. Ultimately, the sofa set and the combined set have such a bad image geometry that VSFM fails to construct an image network and simultaneously to compute a good estimate for the focal length, which leads to radial distortion being unable to be reliably estimated. This can be seen from Table 2, where the ‘VSFM #’ image network size for these sets is much smaller than the original size of the dataset, ‘# images’. On the other hand, EPOS performs well. Note that EPOS does not need to compute the focal length  $f$ , in contrast to BA-based methods, such as the VSFM or the Caltech Toolbox used to compute the reference results. Another difference is that EPOS does not require an image network of good geometry, but can operate on image pairs only. Finally, note that the reference values for  $f$  and  $\eta$  in Table 2 are the same as for Figure 5.

We conclude that EPOS is robust to noise, but also to false matches. However, if false matches are so abundant that the fundamental matrix calculation results in a wrong  $F$  with significantly displaced epipoles, this (obviously) prevents the algorithm convergence by making the majority voting for the center of distortion location fail. Beyond the measures taken here, this risk may further be reduced by preparing the initial data with, e.g., graph-based point tracking [31].

### 5.3. Computational Efficiency

Considering the absolute consumed CPU time, a run with 500 point pairs is processed in about 80 s of CPU time on a single 3.0-GHz core. Considering Figure 5, this corresponds to about 10–12 images. Good results can be obtained with simulated, but also with real datasets of this or a smaller size; see Figure 5 and Table 2, respectively. With real data, Table 2, current unoptimized processing times for the single CPU, full dataset, single runs were 10 min for Dipoli, 4.4 min for sofa, 68 s for combined and 10 min for bookshelf (with fewer, but larger images than Dipoli). Considering the scaling properties of the proposed method beyond this, a practical level of convergence is reached already at the image set sizes of 20; see Figure 5. Improving the computational efficiency by parallel processing and code optimization towards real-time performance will be done as a part of the future work.

## 6. Conclusions

We have presented a solution for correcting lens distortion from uncalibrated images using the epipolar constraint without any calibration rig. The solution employs  $r$ -symmetry to build a global constraint that works even for image networks with bad geometry. As a consequence, the same images may be first used to calibrate the camera for lens distortion in a separate pre-processing step, and then used for 3D reconstruction purposes. Our EpiPOLar-based Solver, or EPOS, is fully automated and benefits from multiple image pairs. EPOS delivers a negative (positive) radial distortion parameter for barrel (pincushion) distortion or it decides that no correction is needed. When the amount of data is increased, EPOS yields a converging result for the distortion model, which currently consists of the one radial distortion parameter and the center of distortion. Feature-matched SIFT points containing false matches can be used as input data. With sufficient data, the rate of convergence is dictated by the stability of the camera’s interior orientation, i.e., the physical properties of the camera. Future work includes the optimization of the EPOS code, extending it for fish-eye lenses, and the introduction of more complex, multi-parameter correction models. We plan to publish the method in an open repository (<https://github.com/vlehtola/rsymmetryEPOS>).

**Acknowledgments:** The authors wish to thank the Academy of Finland for funding this research, Grant Nos. 257755, 272195, and 293389.

**Author Contributions:** V.L. developed and implemented the proposed algorithm, and wrote the first draft of the article. M.K. designed the setup for the simulated data, and produced the reference results. P.R. contributed soundly to the theoretical part, which was put together with V.L. All authors revised the final version of the article.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Nicolae, C.; Nocerino, E.; Menna, F.; Remondino, F. Photogrammetry applied to Problematic artefacts. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2014**, *40*, 451–456.
2. Lehtola, V.V.; Kurkela, M.; Hyyppä, H. Automated image-based reconstruction of building interiors—A case study. *Photogramm. J. Finl.* **2014**, *1*, 1–13.
3. Brown, D.C. Close-range camera calibration. *Photogramm. Eng.* **1971**, *37*, 855–866.
4. Fryer, J.G.; Brown, D.C. Lens distortion for close-range photogrammetry. *Photogramm. Eng. Remote Sens.* **1986**, *52*, 51–58.
5. Fraser, C.; Shortis, M. Variation of distortion within the photographic field. *Photogramm. Eng. Remote Sens.* **1992**, *58*, 851–855.
6. Kim, D.; Shin, H.; Oh, J.; Sohn, K. Automatic radial distortion correction in zoom lens video camera. *J. Electron. Imaging* **2010**, *19*, 043010.
7. Fraser, C.; Al-Ajlouni, S. Zoom-dependent camera calibration in digital close-range photogrammetry. *Photogramm. Eng. Remote Sens.* **2006**, *72*, 1017–1026.
8. Remondino, F.; Fraser, C. Digital camera calibration methods: considerations and comparisons. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2006**, *36*, 266–272.
9. Strecha, C.; Hansen, W.V.; Gool, L.V.; Fua, P.; Thoennessen, U. On Benchmarking Camera Calibration and Multi-View Stereo for High Resolution Imagery. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
10. Bouguet, J.Y. Camera Calibration Toolbox for Matlab. Available online: [http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/) (accessed on 1 July 2016).
11. Photometrix. Australis—Camera Calibration Software. Available online: <http://www.photometrix.com.au/australis/> (accessed on 1 July 2016).
12. Jeong, Y.; Nister, D.; Steedly, D.; Szeliski, R.; Kweon, I. Pushing the Envelope of Modern Methods for Bundle Adjustment. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1605–1617.
13. Wu, C.; Agarwal, S.; Curless, B.; Seitz, S.M. Multicore Bundle Adjustment. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Colorado Springs, CO, USA, 20–25 June 2011; pp. 3057–3064.
14. Lourakis, M.I.A.; Argyros, A.A. SBA: A software package for generic sparse bundle adjustment. *ACM Trans. Math. Softw.* **2009**, *36*, 1–30.
15. Furukawa, Y.; Ponce, J. Accurate Camera Calibration from Multi-View Stereo and Bundle Adjustment. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2008, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
16. Snavely, N.; Seitz, S.; Szeliski, R. Photo Tourism: Exploring Photo Collections in 3D. *ACM Trans. Graph. (TOG)* **2006**, *25*, 835–846.
17. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334.
18. Wu, C. Critical Configurations for Radial Distortion Self-Calibration. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014.
19. Fraser, C.S. Automatic camera calibration in close range photogrammetry. *Photogramm. Eng. Remote Sens.* **2013**, *79*, 381–388.
20. Farid, H.; Popescu, A.C. Blind removal of lens distortion. *J. Opt. Soc. Am. A* **2001**, *18*, 2072–2078.
21. Zhang, Z. On the Epipolar Geometry between Two Images with Lens Distortion. In Proceedings of the 13th International Conference on Pattern Recognition, Vienna, Austria, 25–29 August 1996; Volume 1; pp. 407–411.
22. Fitzgibbon, A.W. Simultaneous Linear Estimation of Multiple View Geometry and Lens Distortion. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, Kauai, HI, USA, 8–14 December 2001; Volume 1, pp. 125–132.
23. Kukeleva, Z.; Pajdla, T. A Minimal Solution to Radial Distortion Autocalibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 2410–2422.
24. Liu, X.; Fang, S. Correcting large lens radial distortion using epipolar constraint. *Appl. Opt.* **2014**, *53*, 7355–7361.

25. Brito, J.H.; Angst, R.; Koser, K.; Pollefeys, M. Radial Distortion Self-Calibration. In Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Portland, OR, USA, 23–28 June 2013; pp. 1368–1375.
26. Willson, R.G.; Shafer, S.A. What is the center of the image? *J. Opt. Soc. Am. A* **1994**, *11*, 2946–2955.
27. Hartley, R.; Kang, S.B. Parameter-free radial distortion correction with center of distortion estimation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 1309–1321.
28. Fischler, M.A.; Bolles, R.C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* **1981**, *24*, 381–395.
29. Lowe, D. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vis.* **2004**, *60*, 91–110.
30. Snavely, N. Bundler: Structure from Motion (SFM) for Unordered Image Collections. Available online: <http://www.cs.cornell.edu/~snavely/bundler/> (accessed on 11 October 2014).
31. Olsson, C.; Enqvist, O. *Stable Structure from Motion for Unordered Image Collections*; Springer: Berlin/Heidelberg, Germany, 2011; Volume 6688, pp. 524–535.
32. McGlone, C.; Mikhail, E.; Bethel, J. *Manual of Photogrammetry*, 5th ed.; American Society of Photogrammetry: Bethesda, MD, USA, 2005.
33. Wu, C. VisualSFM: A Visual Structure from Motion System. Available online: <http://ccwu.me/vsfm/doc.html> (accessed 1 July 2016).



© 2017 by the authors; licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).