# Conic-based Lens Distortion Estimation

No Author Given

No Institute Given

**Abstract.** In this paper we present a procedure to estimate camera lens distortion using a set of tuples of collinear points of known cross-ratio. The method computes the center of distortion and two correction parameters, which are equivalent to modeling the first radial and decentering distortion coefficients. It is based on the projective invariance of the cross-ratio of four collinear points. Using this invariant, we compute the center of lens distortion as the intersection of a set of conics. Experimental results show how the system degrades in the presence of noise in in the input data and how the estimation error affects the correction of the distorted points.

## 1 Introduction

Most computer vision models used nowadays assume that images are acquired with a *pinhole* camera. This camera model establishes a linear relation between the projective coordinates of points in the scene and their projection on the image. Thanks to this linearity, the geometrical relations between multiple images become tractable. However, as a result of several type of imperfections in the design and assembly of lenses in the optical system, cameras normally used in computer vision and robotics never behave as perfect pinholes. So, they are commonly modeled to be the composition of a non-linear lens distortion model and a linear projective (pinhole) camera model. Normally, these models are estimated separately in a two step process. First the image is preprocessed to estimate and eliminate lens distortion and, in a second step, multiple view geometry is computed. This paper deals with the estimation of the non-linear lens distortion model.

In the classical calibration techniques used in photogrammetry, lens distortion is estimated along with the linear camera projection model parameters by bundle adjusting a set of 3D to 2D correspondences obtained from the images of an object of known structure. This approach, known as "stellar" calibration, was also used by Tsai [11] and Lenz and Tsai [5], who proposed the radial alignment constraint to decompose the camera model into linear and non linear parts, which were estimated separately. Within this same approach, Weng [12] uses an iterative minimization scheme in which lens distortion and camera projection parameters are fixed in turn and jointly estimated. More recently Heikkilä [4] introduces a procedure to obtain unbiased estimates of the location of calibration control points and forward and backward mapping models to compute a minimum variance estimate of camera parameters. The problem common to these techniques which jointly minimize all camera parameters is that the strong coupling that exists between projective parameters and lens distortion parameters produces biased and unstable estimates [8, 12].

The second group of methods, called "non-metric," rely on projective constraints between images or between an image and the scene. A camera with lens distortion cannot be considered a perfect projective device and, in consequence, certain projective constraints are not fully satisfied. These methods estimate lens distortion independently of the camera projective parameters by searching for the distortion parameters which make the undistorted image satisfy the selected projective constraint. Different projective constraints have been used in the literature. "Plumb-line" methods are based on the fact that, under perspective projection, straight lines in space project to straight lines in the image. With this constraint at least one image of a set of straight lines is needed to recover the image distortion parameters. Although Ahmed and Farag [1] present a closed-form solution to the distortion coefficients, most other plumb-line methods are based on iterative minimization procedures that search for the lens distortion parameters that straighten the lines [2, 8, 10]. Other non-metric methods are based on iteratively minimizing a cost function based on projective constraints between two images (epipolar geometry) [13] or three images (trilinear relations) [7]. In [3] closed-form solutions for one term of radial lens distortion and epipolar geometry (fundamental matrix) is obtained from a set of correspondences between two images.

The work most related to the estimation method presented in this paper is [6]. In this work G. Q. Wei and Song de Ma present procedure that iteratively searches for the lens distortion parameters that keep invariant the cross-ratio between sets of four points in space and their projection on the image. This method is half-way between the stellar and non-metric methods as it uses knowledge of the structure of the scene and the invariance of a projective constraint between an image and the scene.

In this paper we present a procedure to estimate camera lens distortion using the image of a calibration grid from which we know the cross-ratio of a set of 4-tuples of collinear points. Our method can be seen as an extension of the work in [6], as it is also based on the projective invariance of the cross-ratio. We go one step further and using some classic results from Projective Geometry we derive geometric constraints based on conics to find the center of lens distortion.

## 2 Camera model

Let $P(X, Y, Z)$ be a point in the scene, $\hat{p}(\hat{u}, \hat{v})$ be the ideal undistorted projection of that point, and $p(u, v)$ be the the actual (distorted) projection of that point onto the camera image plane. The *pin-hole* model projectively relates a scene point and its ideal projection through

$$\lambda \begin{bmatrix} \hat{u} \\ \hat{v} \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} [\,\mathbf{R} \,|\, t\,] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix},$$

where $\lambda \neq 0$, $f$ is the camera focal length, and $\mathbf{R}$ and $t$ are respectively a $3 \times 3$ rotation matrix and a translation vector representing the relative position of scene and camera reference systems.

The following expression introduces lens distortion in the image formation model

$$\hat{u} = u + \delta_u(u, v) \tag{1}$$
$$\hat{v} = v + \delta_v(u, v), \tag{2}$$

where $\delta_u$, $\delta_v$ are non-linear functions that model the displacement caused by lens distortion.

In this paper we will consider radial and decentering distortion. Radial distortion is caused by imperfect lens shape and results in different magnification scales for different imaging angles. It causes points to radially spread or crowd together around the distortion center. It can be modeled by the following expression

$$\delta_u^r(u, v) = u(1 + \kappa_1(u^2 + v^2) + \kappa_2(u^2 + v^2)^3 + \ldots) \tag{3}$$
$$\delta_v^r(u, v) = v(1 + \kappa_1(u^2 + v^2) + \kappa_2(u^2 + v^2)^3 + \ldots). \tag{4}$$

Decentering distortion is caused by the lack of orthogonality between the lens components and the image detector with respect to the optical axis. It has both radial and tangential components, which can be analytically described by the following expression

$$\delta_u^d(u, v) = \mathcal{P}_1(3u^2 + v^2) + 2\mathcal{P}_2 uv + \ldots$$
$$\delta_v^d(u, v) = 2\mathcal{P}_1 uv + \mathcal{P}_2(3u^2 + v^2) + \ldots.$$

In our camera model model we will take into account the first term of radial distortion and we will consider that the camera principal point is different from the image distortion center, $(\tilde{\imath}_0, \tilde{\jmath}_0)$. By doing this we are also implicitly considering one term of the decentering distortion, see [9] for a proof.

Finally, the digitization process that transforms image plane coordinates to pixel coordinates can be modeled by the following affine model

$$i = v\, d_v^{-1} + \tilde{\imath}_0 \tag{5}$$
$$j = u\, d_u^{-1} + \tilde{\jmath}_0, \tag{6}$$

where $d_u$ and $d_v$ are respectively the horizontal and vertical size of an image pixel.

## 3 Proposed Method

Let $P_{i,i=1\ldots4}$ be four known collinear points in the scene and let $p_{i,i=1\ldots4}$ be their distorted projection onto the image plane. Ideally, under a perfect pinhole camera model, these points would project to $\hat{p}_{i,i=1\ldots4}$, which lie on a straight line.

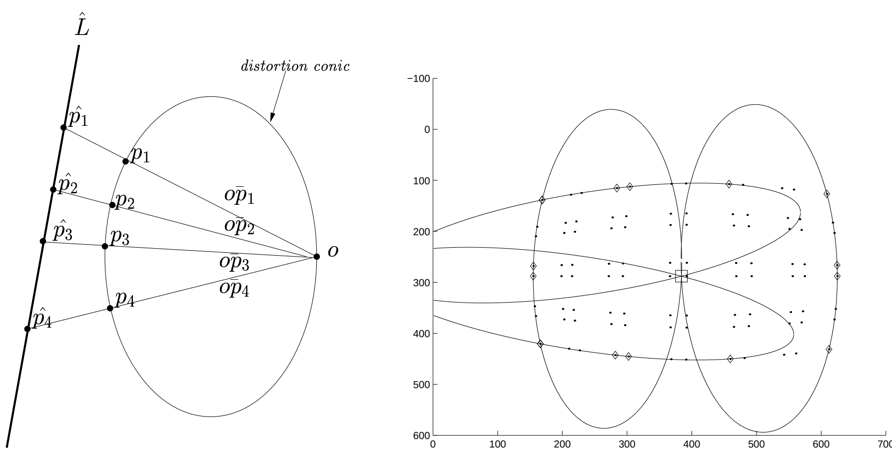
The pencil of points $P_i, i=1\ldots4$ on the 3D line $L$ and the pencil of ideally projected points $\hat{p}_i, i=1\ldots4$ on the 2D line $\hat{L}$ are related through a *perspectivity* whose center is the camera optical center. Also, points $\hat{p}_{i,i=1\ldots4}$, and the lines joining these points with the center of distortion, $\bar{op}_{i,i=1\ldots4}$, form another perspective mapping whose center is the center of distortion (see Fig. 1(a)), so

$$L(P_1,P_2,P_3,P_4) \quad \overset{=}{\wedge} \quad \hat{L}(\hat{p}_1,\hat{p}_2,\hat{p}_3,\hat{p}_4) \quad \overset{=}{\wedge} \quad o(\bar{op}_1,\bar{op}_2,\bar{op}_3,\bar{op}_4).$$

where $\overset{=}{\wedge}$ stands for *perspectivity* or *perspective transformation.* In consequence, the pencil of points $P_{i,i=1\ldots4}$ and the pencil of lines $\bar{op}_{i,i=1\ldots4}$ are related through a *projectivity* of the form

$$L(P_1,P_2,P_3,P_4) \quad \bar{\wedge} \quad o(\bar{op}_1,\bar{op}_2,\bar{op}_3,\bar{op}_4).$$

where $\bar{\wedge}$ stands for *projectivity* or *projective transformation.* Note that a projectivity can be expressed as the combination of two o more perspectivities.




(a) Pencil of lines through the distorted and ideal points whose center is the distortion center

(b) Distortion conics generated for a group of points. Marks show the points used for computing the conics. The square location shows the ground truth for the distortion center

**Fig. 1.** Construction of the distortion conics

On the other hand, the *cross-ratio* is a projective invariant, i.e. two pencils of projective entities related through a projectivity share the same cross-ratio. This means that the cross-ratio of points $P_{i,i=1\ldots4}$ is the same as the cross-ratio of their ideal projections $\hat{p}_{i,i=1\ldots4}$. And, hence, it coincides with the cross-ratio of the pencil of lines that passes through these projections. That is

$$(P_1, P_4; P_3, P_2) = (\bar{op}_1, \bar{op}_4; \bar{op}_3, \bar{op}_2). \tag{7}$$

A well-known principle of projective geometry (*Chasles' Theorem*) states that *"if a, b, c, and d are points on a plane such that no three are collinear, a conic is the locus (collection) of points a, b, c, d, and a fifth variable point x such that* $(\bar{xa}, \bar{xd}; \bar{xc}, \bar{xb}) = c$*, where* $c \neq 0, 1$*"*. This means that $(\bar{op}_1, \bar{op}_4; \bar{op}_3, \bar{op}_2) = c$ is the equation of a conic on which the center of distortion and the four distorted points $p_{i,i=1\ldots4}$ lie.

In conclusion, given the distorted projection of four collinear scene points with known cross-ratio, there exists a unique conic that passes through those points and the center of distortion (see Fig. 1(b)). We call this curve the *distortion conic*. In order to compute the center of distortion, several 4-tuples of distorted points are needed. The center of distortion will be the intersection of the distortion conics computed from all available 4-tuples of points (see Fig. 1(b)). In the following subsection we will derive the equation of the distortion conic and present a procedure to compute the center of distortion.

### 3.1 Computing the center of distortion

The cross-ratio of four collinear scene points is given by

$$c = (D_1, D_4; D_3, D_2) = \frac{D_1D_3 \cdot D_2D_4}{D_3D_4 \cdot D_1D_2}. \tag{8}$$

The cross-ratio of a pencil of four lines is obtained as follows

$$(\bar{op}_1, \bar{op}_4; \bar{op}_3, \bar{op}_2) = \frac{\sin\theta_{13}\sin\theta_{24}}{\sin\theta_{12}\sin\theta_{34}} = \frac{|\bar{op}_1 \times \bar{op}_3||\bar{op}_2 \times \bar{op}_4|}{|\bar{op}_1 \times \bar{op}_2||\bar{op}_3 \times \bar{op}_4|}, \tag{9}$$

where $\theta_{st}$ is the angle between lines $\bar{op}_s$ and $\bar{op}_t$ and $sin\theta_{st} = \frac{|\bar{op}_s \times \bar{op}_t|}{\|\bar{op}_s\|\|\bar{op}_t\|}$.

Expanding $|\bar{op}_s \times \bar{op}_t|$ in terms of pixel coordinates, we have

$$|\bar{op}_s \times \bar{op}_t| = u_t v_s - u_s v_t = \{(j_t - \tilde{\jmath}_0)(i_s - \tilde{\imath}_0)d_u d_v\} - \{(j_s - \tilde{\jmath}_0)(i_t - \tilde{\imath}_0)d_u d_v\}. \tag{10}$$

From equations (8) (9) and (10), we get the distortion conic defined by points $p_{i,i=1\ldots4}$ and cross-ratio $c$:

$$\mathcal{Q}(\tilde{\imath}_0, \tilde{\jmath}_0) \equiv A\tilde{\imath}_0^2 + B\tilde{\imath}_0\tilde{\jmath}_0 + C\tilde{\jmath}_0^2 + D\tilde{\imath}_0 + E\tilde{\jmath}_0 + F = 0, \tag{11}$$

where

$A = (i_3 - i_1)(i_4 - i_2) - c(i_4 - i_3)(i_2 - i_1)$

$$B = [(i_3 - i_1)(j_2 - j_4) + (j_1 - j_3)(i_4 - i_2)] - c[(i_4 - i_3)(j_1 - j_2) + (j_3 - j_4)(i_2 - i_1)]$$
$$C = (j_1 - j_3)(j_2 - j_4) - c(j_3 - j_4)(j_1 - j_2)$$
$$D = [(i_3 - i_1)(j_4 i_2 - j_2 i_4) + (i_4 - i_2)(j_3 i_1 + j_1 i_3)] - c[(i_4 - i_3)(j_2 i_1 - j_1 i_2) + (i_2 - i_1)(j_4 i_3 + j_3 i_4)]$$
$$E = [(j_1 - j_3)(j_4 i_2 - j_2 i_4) + (j_2 - j_4)(j_3 i_1 - j_1 i_3)] - c[(j_3 - j_4)(j_2 i_1 - j_1 i_2) + (j_1 - j_2)(j_4 i_3 - j_3 i_4)]$$
$$F = [j_4 i_2 j_3 i_1 - j_3 i_1 j_2 i_4 - j_4 i_2 j_1 i_3 + j_1 i_3 j_2 i_4] - c[j_4 i_3 j_2 i_1 - j_4 i_3 j_1 i_2 - j_3 i_4 j_2 i_1 + j_3 i_4 j_1 i_2]$$

The distortion center is the point common to all distortion conics. In general, two conics intersect at most in four points (see Fig. 1(b)). So, we need several 4-tuples of collinear points in space to uniquely determine the distortion center. It can be computed through the following minimization, which computes the position of the point with minimum algebraic distance to all conics:

$$(\tilde{\imath}_0, \tilde{\jmath}_0) = \min_{i,j} \sum_{k=1}^{N>3} [A_k i^2 + B_k ij + C_k j^2 + D_k i + E_k j + F_k]^2, \qquad (12)$$

where $[A_k, B_k, C_k, D_k, E_k, F_k]$ are the parameters of the k-th distortion conic.

We propose the following algorithm for computing the distortion center:

1. For each tuple of 4 collinear points in the scene, compute their cross-ratio using (8).
2. For their corresponding projections in the image, compute their distortion conic using (11).
3. Compute the common point of intersection of all conics with (12)

### 3.2 The correction parameters

Once estimated the center of distortion, we are ready to compute $\kappa_1$, the radial distortion parameter.

From equations (1-6) we can obtain the corrected coordinates $(\hat{u},\hat{v})$ in the image plane

$$\hat{u} = d_u(j - \tilde{\jmath}_0)[1 + (j - \tilde{\jmath}_0)\kappa_1 d_u^2 + (i - \tilde{\imath}_0)\kappa_1 d_v^2] \qquad (13)$$
$$\hat{v} = d_v(i - \tilde{\imath}_0)[1 + (j - \tilde{\jmath}_0)\kappa_1 d_u^2 + (i - \tilde{\imath}_0)\kappa_1 d_v^2], \qquad (14)$$

The unknowns of this equation are $\kappa_1$, the first radial distortion coefficient, $d_u$ and $d_v$, the size of the pixel in the horizontal and vertical directions respectively. In order to find the values of this unknowns we must set up a proper cost function.

Under perfect pinhole projection, straight lines in the scene are projected to straight lines in the image. But in our case, the straight lines in the image are curved due to the effects of camera lens distortion. We know, however, that ideally this lines should be straight, so this cost function will search for the distortion values that straighten all known lines in the scene. This is what all *plumb-line* methods do [1, 2, 8, 10]. We will use the cost function proposed in [6]:

$$(\xi_1, \xi_2) = \min_{\kappa_1 d_u^2, \kappa_1 d_v^2} \sum_{k=1}^{N} (\hat{v}_3^k - \hat{v}_1^k)(\hat{u}_2^k - \hat{u}_1^k) - (\hat{v}_2^k - \hat{v}_1^k)(\hat{u}_3^k - \hat{u}_1^k), \qquad (15)$$

where, $(\hat{u}_i^k, \hat{v}_i^k)$ are the corrected coordinates of the i-th point in the k-th tuple.

Minimizing this cost function for the unknowns $\kappa_1 d_u^2$ and $\kappa_1 d_v^2$ will give us the values that best fit with the distorted points. Note that we have taken $d_u$ and $d_v$ as common constant factors in the whole equation.

## 4 Experimental Results

In this section we will describe the experiments that we conducted in order to test our method. Both synthetic and real image data have been considered.

For the synthetic experiments, we generate a planar calibration pattern, consisting of points distributed in 8 rows and 12 columns, which gives us a total of 96 calibration points. These points will be displaced from their original positions according to our radial distortion model. In order to test the suitability of the algorithm we have distorted these points using two different parameters: $\kappa_1 = 5e - 7$, which represents a low level of distortion and $\kappa_1 = 1e - 6$ for medium-high level of distortion. The true center of distortion in both cases is $(\tilde{\imath}_0 = 288, \tilde{\jmath}_0 = 384)$. Finally, these points will be perturbed using Gaussian noise with standard deviation $(\sigma)$ varying from 0 to 2. The results shown in Figs. 2 and 3 represent the median value of 100 experiments made for each noise level.

The first experiment will use the synthetically generated data to evaluate the estimation of the distortion center obtained by the algorithm presented in section 3.1. In Fig. 2(a) we show the Euclidean distance between the computed value of the distortion center and the value used as ground truth. From the analysis of these results we can conclude that the procedure is quite sensitive to the noise contaminating the data. This is so because the estimation of the distortion conics parameters is quite sensitive to this noise.

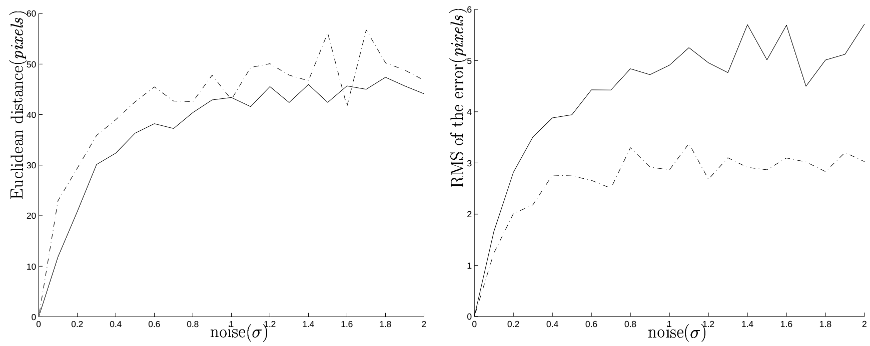
In the second experiment we test the estimation of the correction parameters $\xi_1$ and $\xi_2$ using the set of synthetically generated data and the distortion center obtained from the previous experiment. Results are shown in figure Fig. 3. Results show that the error in the computation of $\xi_1$ and $\xi_2$ increases almost linearly with the amount of noise in the images.

In order to understand how these errors affect the process of undistorting the image, we are going to make one more synthetic test. Using the ideal points of the synthetic pattern and distorting them we will use our algorithms to estimate the distortion parameters and undistort the image. Then, we compute the Euclidean distance between the original points and the corrected ones. Figure Fig. 2(b) shows the RMS of these distances.

Finally, in the last test we undistort a real image taken with a CCD Philips camera using a Cosmicar/Pentax 6mm lens. Result are shown in figure Fig. 4. As expected, after the correction process straight lines in the image appear straight.

## 5 Conclusions and Future Work

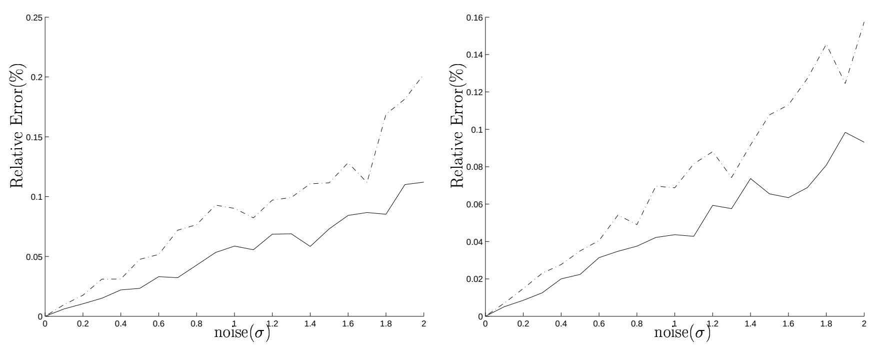
In this paper we have presented a method to estimate camera lens distortion from the image of a set of collinear points of known cross-ratio. The method




(a) Error in the computation of the distortion center vs. noise in the images.

(b) RMS of the error in the undistorting process vs noise in the images.

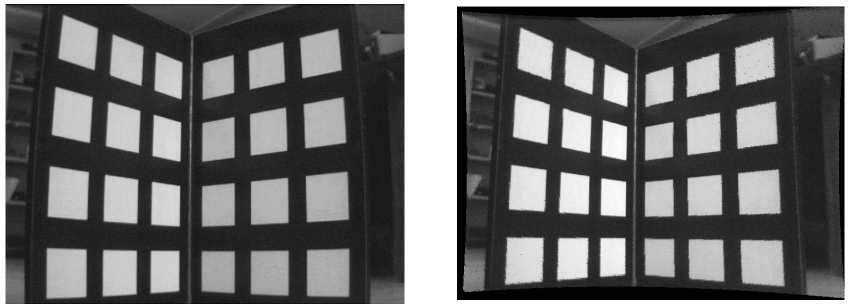
**Fig. 2.** Results for distortion center estimation and RMS of the undistortion process. Dotted lines stands for $\kappa_1 = 5e - 7$ and solid lines for $\kappa_1 = 1e - 6$.




(a) Values of $\xi_1(\kappa_1 d_v^2)$ versus the levels of noise in the image

(b) Values of $\xi_2(\kappa_1 d_u^2)$ versus the levels of noise in the image

**Fig. 3.** Values of the correction parameters vs. the noise in the images. Dotted lines stands for $\kappa_1 = 5e - 7$ and solid lines for $\kappa_1 = 1e - 6$.



(a) Original distorted image (b) Undistorted image

**Fig. 4.** Experiments with real images.

computes the center of distortion and two correction parameters, $\xi_1$ and $\xi_2$, which implicitly model the first radial and decentering distortion coefficients. It is based on the projective invariance of the cross-ratio of four collinear points. Using this invariant, we compute the center of lens distortion as the intersection of a set of conics.

Estimating the center of lens distortion is a very difficult task. A small error in the location of image points entails a big error in the estimation of the distortion center. This problem is common to all lens distortion estimation procedures. However, the conic-based approach presented here provides a geometrical model that can be seen as a step toward more accurate estimation procedures based on, for example, knowledge of the uncertainties in the data.

In order to compute the distortion correction parameters, $\xi_1$ and $\xi_2$, a plumb-line method is used. In this case, the estimation error increases almost linearly with the noise contaminating the data.

In a final experiment with both synthetic and real data, we show how the estimation of the distortion parameters affects the correction lens distortion.

Further research is needed to model how the uncertainties in the location of image points can be used to improve the estimation of the distortion parameters.

## References


1. M.T. Ahmed, A.A. Farah. “Differential Methods for Non-metric Calibration of Camera Lens Distortion.” *Proc. Int. Conference on Computer Vision and Pattern Recognition*, Vol. II, pp.477-482, 2001.
2. F. Devernay, O. Faugeras. “Automatic calibration and removal of distortion from scenes of structured environments”. *Proc of SPIE*, Vol. 2567. 1995.
3. A.W. Fitzgibbon. “Simultaneous linear estimation of multiple view geometry and lens distortion”. *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, pp. , 2001.

4. J. Heikkilä. "Geometric Camera Calibration Using Circular Control Points." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10), pp. 1066-1077, 2000.
5. R.K. Lenz, R.Y. Tsai. "Techniques for Calibration of the Scale Factor and Image Center for High Accuracy Machine Vision Metrology." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10(5), pp. 713-720, 1988.
6. Guo-Qing Wei and Song De Ma. "Implicit and Explicit Camera Calibration: Theory and Experiments." *IEEE Transactions on Pattern Analysis and Machine Intelligence*,16(5), pp. 469-480, 1994.
7. G.P. Stein. "Lens Distortion Calibration Using Point Correspondences." *Proc. Int. Conference on Computer Vision and Pattern Recognition*, pp. 602-608, 1997.
8. G.P. Stein. "Accurate Internal Camera Calibration using Rotation, with Analysis of Sources of Error." *Proc. Int. Conference on Computer Vision*, pp. 230-236, 1995.
9. G.P. Stein. "Internal Camera Calibration using Rotation and Geometric Shapes." AITR-1426. Master's Thesis. MIT-AI Laboratory. 1993.
10. R. Swaminathan, S. Nayar. "Nonmetric Calibration of Wide-Angle Lenses and Policameras." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 22(10), pp.1172-1178, 2000.
11. R.Y. Tsai. "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-shelf TV Camera Lenses." *IEEE Journal of Robotics and Automation*, 3(4), pp.323-344, 1987.
12. J. Weng, P. Cohen, M. Herniu. "Camera Calibration with Distortion Models and Accuracy Evaluation." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(10), pp. 965-980, 1992.
13. Z. Zhang. "On the epipolar geometry between two images with lens distortion." *Proc. Int. Conference on Pattern Recognition*, pp. 407-411, 1996.