Pattern Recognition 43 (2010) 1688–1699

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Robust metric calibration of non-linear camera lens distortion

Carlos Ricolfe-Viala *, Antonio-José Sánchez-Salmerón

Universidad Politecnica de Valencia, Engineering Systems and Automatic Control Department, Camino de Vera s/n, 46022 Valencia, Spain

ARTICLE INFO

Article history:
Received 21 October 2008
Received in revised form
10 July 2009
Accepted 8 October 2009

Keywords:
Lens distortion
Camera calibration
Metric method
Robust estimator

ABSTRACT

Camera lens distortion is crucial to obtain the best performance cameral model. Up to now, different techniques exist, which try to minimize the calibration error using different lens distortion models or computing them in different ways. Some compute lens distortion camera parameters in the camera calibration process together with the intrinsic and extrinsic ones. Others isolate the lens distortion calibration without using any template and basing the calibration on the deformation in the image of some features of the objects in the scene, like straight lines or circles. These lens distortion techniques which do not use any calibration template can be unstable if a complete camera lens distortion model is computed. They are named non-metric calibration or self-calibration methods.

Traditionally a camera has been always best calibrated if metric calibration is done instead of self-calibration. This paper proposes a metric calibration technique which computes the camera lens distortion isolated from the camera calibration process under stable conditions, independently of the computed lens distortion model or the number of parameters. To make it easier to resolve, this metric technique uses the same calibration template that will be used afterwards for the calibration process. Therefore, the best performance of the camera lens distortion calibration process is achieved, which is transferred directly to the camera calibration process.

1. Introduction

Camera calibration has always been an important issue in computer vision. It consists of finding the mapping between the 3D space and the camera plane. This mapping can be separated into two set of transformations. The first one involves the location of the camera in the scene and it is described by the extrinsic parameters of the camera model. The second one is the mapping of the 3D points in the scene to the 2D coordinates in the image. This mapping is modelled by the intrinsic parameters which give the geometry of the camera and the optical features of the sensor. In general, these two transformations can be represented by the ideal pin-hole camera model which simplifies many geometric considerations in which cameras are involved. These simplifications assume an ideal behaviour of the camera, which is never true. Imperfections arise if low-cost video optics, especially wide-angle or fish-eye lenses, are used, or in applications where high accuracy is required. When lens distortion is not negligible, a distortion-free model may result in a high calibration error [1]. In these cases, the pin-hole model is not sufficient and more parameters should be estimated to take into account lens distortions.

Camera lens distortion was first introduced by Conrady in 1919 with the decentring lens distortion. Afterwards, Brown [2,3] proposed the radial, decentring and prism distortion model which has been widely used [4–7]. Some modifications to this model have been reported focused on mathematical treatment of the initial model [8–13], or from a conceptual point of view, without any quantitative evaluation [14]. Recently, a non-parametric model for camera distortion has been proposed by Hartley [15] which only considers the radial distortion. Although the radial component of lens distortion is predominant, it is coupled with the tangential component and therefore it should be taken into account. In fact, the basic formula which represents the conventional model of lens distortion is a sum of three terms corresponding to the radial, decentring and thin prism components. Also, Basu [9] and Devernay [7] have defined specific models for fish-eye lens distortion.

Up to now, a variety of methods for camera calibration have been developed to accommodate various applications [3–6,16–18] (to cite a few). One family of pin-hole calibration methods resolves the camera model using metric information of a 3D, 2D, 1D template. The other family does not use any metric information from the scene in what is called self-calibration methods. The

* Corresponding author. Tel.: +34 963 877 007x88231; fax: +34 96 387 98 16.
E-mail addresses: cricolfe@isa.upv.es (C. Ricolfe-Viala), asanchez@isa.upv.es (A.-J. Sánchez-Salmerón).
URL: http://robotica.isa.upv.es/.

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.10.003

problem of self-calibration is that if all camera parameters are unknown, it is very unstable [19]. In these cases, known camera motion helps in getting more stable and accurate results [20,21], but it is not always easy to get "pure camera rotation". Better results are always computed with metric calibration methods. Referring to the computed camera models, some only compute the pin-hole model parameters [18], and others solve the pin-hole model together with the distortion parameters [4–6]. Since the distortion is most often coupled with the internal and external camera parameters, methods which extend the calibration of the pin-hole model to obtain the camera distortion parameters result in high errors on the internal parameters [5]. Therefore, it is necessary to use a different method to estimate the camera distortion model apart from the pin-hole model. Furthermore, if distortion parameters are known before, it allows correcting observed positions of points in the image, achieving quasi-ideal camera behaviour similar to the pin-hole model and the camera can be calibrated accurately.

To compute the camera distortion model only, several methods have been proposed which do not rely on any knowledge of the scene points, nor do they need calibration objects or any known structure. They are called non-metric calibration or self-calibration methods. These methods use geometric invariants of some image features like straight lines [3,7,22–24], vanishing points [25] or the image of a sphere [26]. Methods [3,7,22–24] rely on the fact that straight lines in the scene must always be perspective projected on straight lines in the image. In addition, Becker and Dove [25] use the minimum vanishing point dispersion constraint between three mutually orthogonal sets of parallel lines to recover the distortion parameters. In this case, the inconvenience of the method is to find triplets of orthogonal lines. Other methods use correspondences between points in different images from multiple views to compute camera distortion parameters [27–29]. They are not easy to solve and are likely to produce some false data to the distortion algorithm.

In this paper we propose a metric method for lens distortion calibration. Since metric information of the scene is going to be used in the pin-hole calibration process, we propose using this information prior to calibrating the lens distortion. In this way, the camera calibration should be understand as a two-step procedure where first the camera lens distortion is computed and corrected and second, the pin-hole camera model is calibrated. With the metric method proposed in this paper, camera lens distortion can be fully defined without risk of instabilities. It has been reported in [3,23] that including both the distortion centre and the decentring coefficients in the non-linear optimization may lead to instabilities of the non-metric lens distortion estimation algorithm. To avoid it, some researchers, e.g. [23], used a coarse-to-fine exhaustive search for the distortion centre around the image centre but in the end, it has resulted in a prolonged search with no guaranties of stability [24]. The proposed method offers the possibility of computing all the parameters which model the camera distortion with guaranties of stability and without any additional work since metric information from the scene is used to compute camera pin-hole model parameters afterwards. Taking into account the experience with the calibration of the pin-hole camera model, metric calibration methods are always more stable and give better results than non-metric methods, called self-calibration.

The proposed method relies on the idea that an image of a structure maintains its portions according to a perspective projection. Therefore, there are different magnitudes within the structure that are fixed independently of the position, orientation and characteristics of the camera that takes the image. What is proposed in this paper is a method for correcting the detected points in the image to fulfil constrains given by the features of the structure and to calibrate the lens using these two sets of points, the detected and the corrected ones. From the point of view of the pin-hole model calibration, this method is very useful since full structured templates like chessboards are used to resolve them. This means that the same information can be used to calibrate the lens distortion before the camera is calibrated. For this paper, only the planar template has been evaluated since the camera calibration process using a two dimensional template gets better performance according with Sun et al. experiments [30]. Similar dissertation could be done if a three dimensional template is wanted to be used but it is out of the scope of this paper.

This paper is organized as follows. Section 2 gives a brief description of the lens distortion model. Section 3 proposes the geometric invariants of a structure independent of perspective projection and how the location of the set of points in the image is corrected. Section 4 derives the estimate of the camera lens distortion model using both sets of points. Several experimental results are reported on both real and synthetic data and are also compared with non-metric camera lens distortion calibration methods. The paper ends with some concluding remarks.

2. Non-linear lens distortion model

Pin-hole model represents ideal camera behaviour. Normally, image formation is distorted since a camera has mechanical imperfections and lens misalignments. These camera manufacturing deficiencies produce image distortions because photons do not follow a straight line from the object in the scene to the CCD. Therefore, the mapping between the 3D points and 2D image points is deviated from the ideal pin-hole camera. As a result, the observed point in an image has geometrical displacements from its ideal position. The image distortion model [31] is usually given as a mapping from the distorted image coordinates $\boldsymbol{q_d}=(u_d, v_d)$, which are observables in the images, to the undistorted image coordinates which are not physically measurable $\boldsymbol{q_p}=(u_p, v_p)$. This mapping is given by

$$\begin{aligned} u_p &= u_d - \delta_u \\ v_p &= v_d - \delta_v \end{aligned} \tag{1}$$

such that

$$\begin{aligned} \delta_u &= \Delta u_d \cdot (k_1 \cdot r_d^2 + k_2 \cdot r_d^4 + \ldots) + p_1(3\Delta u_d^2 + \Delta v_d^2) + 2p_2 \cdot \Delta u_d \cdot \Delta v_d + s_1 \cdot r_d^2 \\ \delta_v &= \Delta v_d \cdot (k_1 \cdot r_d^2 + k_2 \cdot r_d^4 + \ldots) + 2p_1 \cdot \Delta u_d \cdot \Delta v_d + p_2(\Delta u_d^2 + 3\Delta v_d^2) + s_2 \cdot r_d^2 \end{aligned} \tag{2}$$

This represents a non-linear camera polynomial distortion model (NLPD) where r is the distance from the point $\boldsymbol{q_d}=(u_d, v_d)$ to the distortion centre defined as, $\boldsymbol{c}=(u_0, v_0)$, $\Delta u_d = u_d - u_0$, $\Delta v_d = v_d - u_0$. r_d is computed as $r_d^2 = \Delta u_d^2 + \Delta v_d^2$. Radial distortion is modelled using coefficients k_1, k_2, k_3,... and produces a displacement of point position along the line connecting with the distortion centre, produced because of defects in the lens curve. Negative radial displacement is also called barrel distortion; positive radial displacement is known as cushion distortion. Another kind of distortion is called image decentring or tangential distortion, arising because optical lens centres are not in the same line. It is modelled with p_1, p_2, p_3,... Finally, if the camera lenses are not correctly aligned and are not perpendicular to the camera optical axis, a prism distortion is produced modelled with s_1, s_2, s_3,... Effective distortion is the sum of the three distortions previously characterized. In view of the physical meaning of each distortion parameter and possible coupling between them, a simplified non-linear distortion model can be obtained [32]. Normally, terms higher than 2 are comparatively insignificant and they are discarded [4,5,33].

3. Geometric invariants

To find the transformation or undistortion that maps the actual camera image plane onto an image following the perspective camera model, it is necessary to calibrate the distortion with some of the models described in Section 2. This transformation gives the exact positions of observed points in the image which are displaced due to lens distortion. If the perspective camera model was followed, the position of these points in the images would be the ones given by the perspective projection of these points in the scene. Thus, if points in the scene fulfil some ratios and constrains between them, they will be true in the image of these points taking into account the perspective projection. Therefore, to get the right position of these points in the distorted image, it is necessary to known these ratios and constrains in the scene and try to making them true for these distorted points. When both sets of distorted and corrected points are known, it will be possible to adjust these transformations following one of the models described in Section 2.

Now the key is to define the right ratios and constrains of these points in the scene. Since the perspective projection is unknown a priori, these ratios and constrains must also be invariants with the perspective projection. These invariants make it possible to correct the point's positions in the image without knowing the perspective camera model. In addition ratios and constrains must be minimal to make it easier to implement the algorithm. To define the minimum number of constrains an image of a regular structured scene must be taken. Thus, if the structure has an equal ratio and constrains between predefined points, both should be true when all the right positions of all the distorted points in the image will be found. If no regular structures were used different metrics should be used to correct the distorted points in the image. To make it easier, the chosen template is the "*chessboard*" shown in Fig. 1 since ratios between different corners are equal and different corners form straight lines which are parallel and orthogonal between them.

To define an invariant ratio, projective geometry preserves neither distances nor ratios of distances. However, the cross-ratio, which is a ratio of ratios of distances, is preserved and, therefore, it is a useful concept which makes it possible to correct the point's positions without knowing the perspective projection represented by the camera model. Cross-ratios are invariants of projective geometry in the sense that they are preserved by projective transformations. Zhang et al. [34], used cross-ratio to calibrate just a first order radial distortion. In this paper a full method is described to calibrate different types of distortions models in a wide range of orders.

On the other hand, to define a constrain that all points in the image of the template obey, it is necessary to consider that the straight lines always will be straight [7,24]. Following the pin-hole camera model, the projection of every line in the space to the camera is a line. Consequently, since all observed points in the image of the chosen template should be the intersections of two lines, all of them must fit in two lines of the image perfectly. Therefore, this constrain can be used to correct the observed point's position in the image and to get the right approach where points should be. In the template, straight lines are parallel separated a fixed distance and therefore corrected points will form straight lines separated a distance defined by the cross-ratio.

3.1. Cross-ratio invariability for perspective projection

Using the well known cross-ratio, a rule is defined which remains invariable under perspective projection [35]. In this way, assuming four points A,B,C,D as lying on a line L, the ratio of two simple ratios for four such collinear points is the cross-ratio, defined as

$$CR(A,B,C,D) = \frac{AC}{CB} / \frac{AD}{DB} \tag{3}$$

where the points A and B are the datum points, and the points C and D are the reference points. This invariance property of the cross-ratio can be applied to correct observed points in the image. For four points of the planar calibration template $\boldsymbol{p}_1=(x_1, y_1)$, $\boldsymbol{p}_2=(x_2, y_2)$, $\boldsymbol{p}_3=(x_3, y_3)$, $\boldsymbol{p}_4=(x_4, y_4)$, lying on a same line, their cross-ratio can be obtained in advance, and written as

$$CR(p_1,p_2,p_3,p_4) = \frac{d_{13}\cdot d_{24}}{d_{14}\cdot d_{23}} \tag{4}$$

where d_{ij} represents the distance between the point $\boldsymbol{p_i}$ and $\boldsymbol{p_j}$ defined as $d_{ij}^2 = (x_i - x_j)^2 + (y_i - y_j)^2$. Assuming the image coordinates of points $\boldsymbol{p}_1$, $\boldsymbol{p}_2$, $\boldsymbol{p}_3$, $\boldsymbol{p}_4$ are $\boldsymbol{q_{1p}}=(u_{1p}, v_{1p})$, $\boldsymbol{q_{2p}}=(u_{2p}, v_{2p})$, $\boldsymbol{q_{3p}}=(u_{3p}, v_{3p})$, $\boldsymbol{q_{4p}}=(u_{4p}, v_{4p})$, thus there exists the following equation based on the cross-ratio invariability.

$$CR(q_{1p},q_{2p},q_{3p},q_{4p}) = \frac{s_{13}\cdot s_{24}}{s_{14}\cdot s_{23}} = CR(p_1,p_2,p_3,p_4) \tag{5}$$

where s_{ij} represents the distance between the point $\boldsymbol{q_{ip}}$ and $\boldsymbol{q_{jp}}$ defined as $s_{ij}^2 = (u_{ip} - u_{jp})^2 + (v_{ip} - v_{jp})^2$. Since planar template has a regular distribution of all interesting points, $CR(\boldsymbol{p}_1,\boldsymbol{p}_2,\boldsymbol{p}_3,\boldsymbol{p}_4)$ is equal to all four set of points in the planar template which are equally distributed and in consequence its corresponding points in the image. Distorted coordinates $\boldsymbol{q_{i,d}}$ will satisfy the cross-ratio $CR(\boldsymbol{p}_1,\boldsymbol{p}_2,\boldsymbol{p}_3,\boldsymbol{p}_4)$ if they are corrected to its right positions $\boldsymbol{q_{i,p}}$.

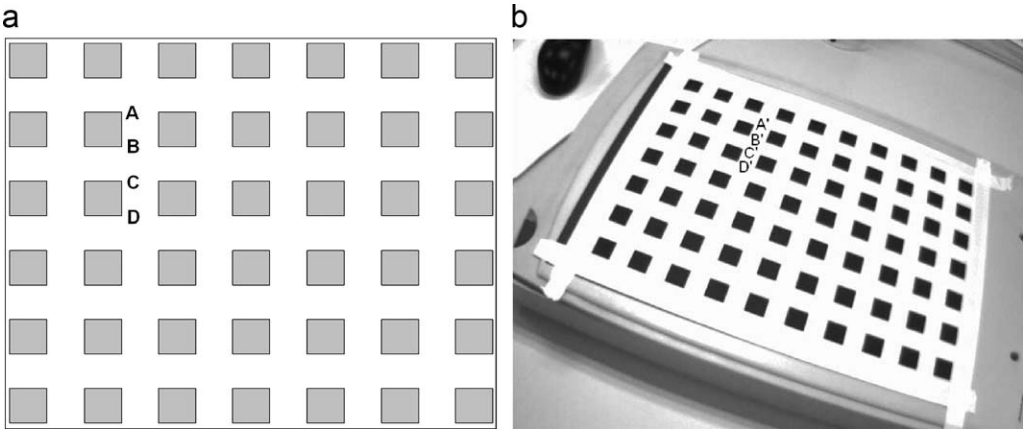

Fig. 1. (a) Chessboard used as template. (b) Image of the chessboard. Cross-ratio of named corners does not change when the image perspective change. Also straight lines remain straight. With this template restrictions are true in both directions.

Points in the image are separated in n sets of m points which form straight lines, where n is the number of straight lines in the calibration template and m is the number of points in each line. So, $\boldsymbol{q_{k,l}}$ is a point k of the straight line l in the image, $l=1\ldots n$, $k=1\ldots m$. To find the right position $\boldsymbol{q_{k,l,p}}$ of each distorted point in the image $\boldsymbol{q_{k,l,d}}$, a non-linear search starting with $\boldsymbol{q_{k,l,d}}$ must be done to minimize the following equation:

$$J_{CR}=\sum_{l=1}^{n}\sum_{k=1}^{m-3}(CR(q_k,q_{k+1,l},q_{k+2,l},q_{k+3,l})-CR(p_1,p_2,p_3,p_4))^2 \quad (6)$$

Taking into account that chosen planar template has straight lines in vertical and horizontal directions, this equation must be true for horizontal and vertical straight lines. Each point in the image $\boldsymbol{q_i}$ belongs to two lines and it represents their intersection. Thus, the cross-ratio of this point $\boldsymbol{q_i}$ will be true with all its neighbourhoods in both directions when Eq. (6) is minimized. $CR(\boldsymbol{p_1},\boldsymbol{p_2},\boldsymbol{p_3},\boldsymbol{p_4})$ is computed previously when the planar template is designed.

3.2. Straight lines

Points in the image has been separated in n straight lines of m the number of points. Point correction should be done in order to fit in the line perfectly. If the Hesse normal form is used to describe a line, a generic point in the image $\boldsymbol{q_i}$ with coordinates u_i, v_i, $\boldsymbol{q_i}=(u_i, v_i)$, fits in the straight line l, if the following expression is true:

$$\vec{r}_l\cdot\vec{q}_i-d_l=0 \quad (7)$$

The vector $\vec{r}_l$ represents the unit normal vector of the straight line l and $d_l\geq 0$ is the distance from the origin to the line as shown in Fig. 2a. The dot indicates the scalar product or dot product. Since, $\vec{r}_l$ represents the unit normal vector of the straight line l, Eq. (7) can be rewritten as

$$a_l\cdot u_i+b_l\cdot v_i-d_l=0 \quad \text{with constrain } a_l^2+b_l^2=1 \quad (8)$$

where $a_l=\cos\theta_l$, $b_l=\sin\theta_l$ and θ_l represents the angle of the unit normal vector $\vec{r}_l$ with the U-coordinate axes of the image. Straight line parameters a_l, b_l, d_l are computed with the m points belonging to the line l minimizing the Euclidean distance of the points to the line. Consequently, the Euclidean distance must be zero if all points fit in the straight line perfectly. Therefore, to correct points in the image a non-linear searching must be solved trying to minimize the following equation:

$$J_{ST}=\sum_{l=1}^{n}\sum_{i=1}^{m}(a_l\cdot u_i+b_l\cdot v_i+d_l)^2 \quad \text{with constrain } a_l^2+b_l^2=1 \quad (9)$$

Initial values for a_l, b_l, are computed with the eigen vectors of the symmetric positive-definite matrix M:

$$M=\begin{bmatrix}\sum_{i=1}^{m}u_i^2-\frac{(\sum_{i=1}^{m}u_i)^2}{m} & \sum_{i=1}^{m}u_i\cdot v_i-\frac{(\sum_{i=1}^{m}u_i)(\sum_{i=1}^{m}v_i)}{m}\\ \sum_{i=1}^{m}u_i\cdot v_i-\frac{(\sum_{i=1}^{m}u_i)(\sum_{i=1}^{m}v_i)}{m} & \sum_{i=1}^{m}v_i^2-\frac{(\sum_{i=1}^{m}v_i)^2}{m}\end{bmatrix} \quad (10)$$

d_l is computed with

$$d_l=-a_l\frac{\sum_{i=1}^{m}u_i}{m}-b_l\frac{\sum_{i=1}^{m}v_i}{m} \quad (11)$$

As before, since the chosen template has vertical and horizontal straight lines, the Eq. (9) will be minimized in both directions. Hence each point will be corrected to belong to two lines perfectly.

The line parameters needed in Eq. (9) are computed together with the set of points which accomplish the line itself. Just an initial value computed with (10), (11) is needed to start with the minimization process and they are computed

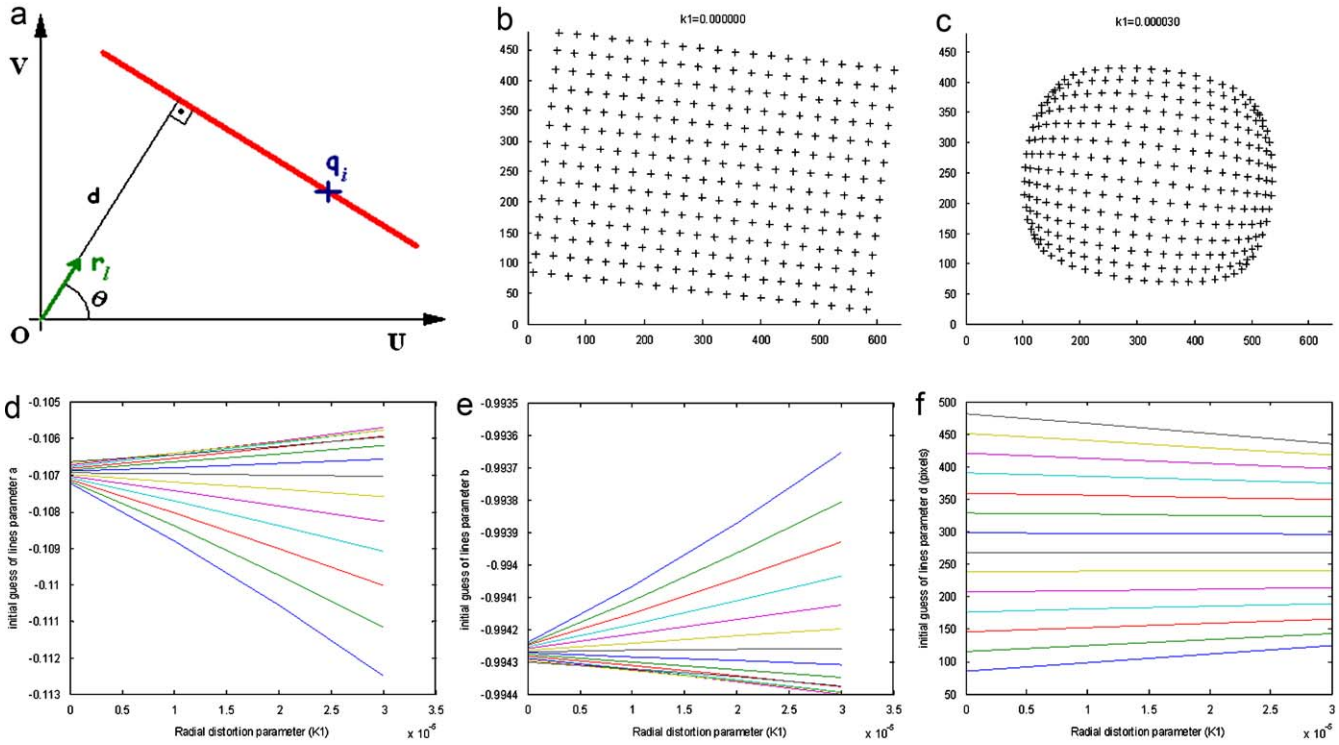

Fig. 2. (a) Line representation in Hesse normal form. (b) Points of 14 synthetic lines represented distributed over a 640×480 image. (c) Distorted image of the points with radial distortion $k_1=3\times 10^{-5}$ (d) Variation of the initial guess of parameter a of the line as a function of the radial distortion. (e) Variation of the initial guess of parameter b of the line as a function of the radial distortion. (f) Variation of the initial guess of parameter d of the line as a function of the radial distortion.

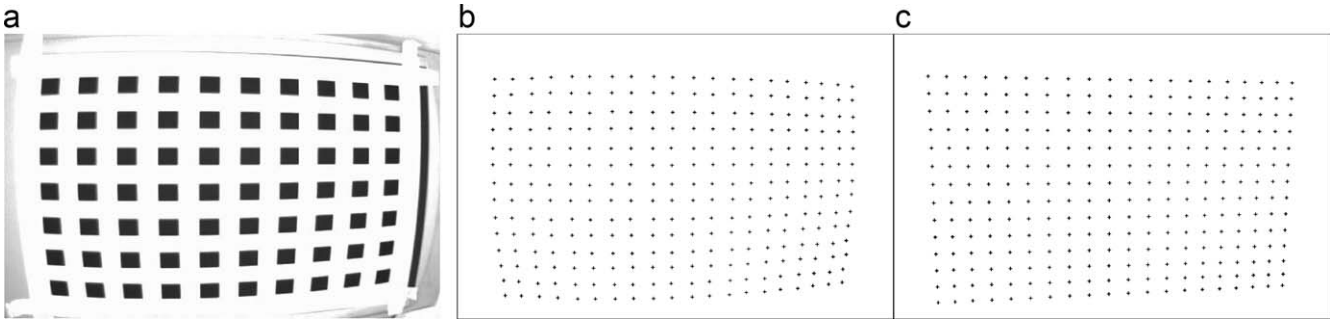

Fig. 3. (a) Image of the chessboard taken with a low-cost industrial camera VCM50. (b) Detected points in the distorted image. (c) Result of the corrected points positions according to the cross-ratio invariability and straight line restrictions. Lens distortion is calibrated mapping the distorted point positions to their right positions.

together with the coordinates of the set of points. This initial value is computed from the detected (thus distorted) image points. If the image is seriously distorted, initial estimation of this ideal line, on which undistorted points should lie, could not be a good initial guess and the iterative non-linear optimization algorithm may end up with a false solution in a local minima. To investigate the effect of the image distortion in the estimation of the ideal line initial guess and how far is this initial guess to the final one, Figs. 2(d)–(f) plots the initial guesses of the lines parameters as a function of the radial distortion k_1. The initial guesses are computed for 14 synthetic lines represented with the points of Fig. 2(b), distributed over a 640×480 image which has been distorted with values of k_1 from 0 to 3×10^{-5} and corrupted by zero-mean Gaussian noise of standard deviation 0.1 pixels. Fig. 2(b) shows the points with $k_1 = 0$ and Fig. 2(c) with $k_1 = 3 \times 10^{-5}$. Distortion centre has been assumed in the centre of the image. True values of the parameters for straight lines are the ones with $k_1 = 0$ showed in Figs. 2(d)–(f) and these are the ones which the minimization algorithm should obtain. The variation of the initial guess of the parameters a_l, and b_l, when the image is seriously distorted is shown Fig. 2(d) and (e) and it is about 0.001 for the lines located close to the border of the image where the distortion is higher. For the parameter d_l, the variation is about 50 pixels for the border lines. These values could be consider a good initial guesses also for seriously distorted images since the differences with the final values of parameters a_l, and b_l are small and with a smooth variation of a_l, and b_l, the parameter d_l will change to the right value.

3.3. Correcting points in the image

To correct the position of the points in the image, an equation must be minimized which includes all constrains introduced in Sections 3.1 and 3.2. On the one hand, the cross-ratio of all sets of four linear points in the horizontal and vertical lines. On the other hand the constrain of the set of points fitting in straight lines. It is represented in the following expression:

$$J_{CP} = \sum_{l=1}^{n} \Big(\sum_{i=1}^{m} (a_l \cdot u_i + b_l \cdot v_i + d_l)^2 + \sum_{k=1}^{m-3} (CR(q_{k,l}, q_{k+1,l}, q_{k+2,l}, q_{k+3,l}) - CR(p_1, p_2, p_3, p_4))^2 \Big) \text{with constrain } a_l^2 + b_l^2 = 1 \quad (12)$$

It must be taken into account that horizontal and vertical straight lines exist in the template and then, the previous equation must be minimized for both directions simultaneously. This optimization step can be resolved using the Levenberg–Marquardt algorithm for non-linear least squares problems. $CR(\boldsymbol{p}_1,\boldsymbol{p}_2,\boldsymbol{p}_3,\boldsymbol{p}_4)$ is known since it represents the cross ratio value for the template points. $CR(\boldsymbol{q}_{k,l},\boldsymbol{q}_{k+1,l},\boldsymbol{q}_{k+2,l},\boldsymbol{q}_{k+3,l})$ is computed with the points of the image which belongs to line l. It starts with the detected coordinates of the points in the image $\boldsymbol{q}_{id}=(u_{id}, v_{id})$, and when the non-linear searching ends, they will be the undistorted ones $\boldsymbol{q}_{ip}=(u_{ip}, v_{ip})$. For the straight lines component of the Eq. (12), the non-linear searching starts with the estimation of the detected (thus distorted) points in the image. Iteratively, the points are corrected and also the estimated parameters a_l, b_l and d_l. When the non-linear searching ends, a set of points $\boldsymbol{q}_{ip}=(u_{ip}, v_{ip})$ are computed which fits the straight lines perfectly.

In addition, this equation adds two terms: straightness and cross-ratio. These are two different quantities which have different meanings and it could therefore be expected that a weight must be included when mixing these quantities. We initially considered weighting each magnitude, but this was finally not necessary, since the minimization of the objective function converges to a point where both quantities are completely satisfied. If the minimization had been converged to a point where one part of the equation were satisfied and the other not, an equation with both weighted quantities would have been needed. Results for a badly distorted image are shown in Fig. 3.

3.4. Are the corrected set of points the unique right solution?

Minimizing Eq. (12) a set of corrected points is computed. They are called "correct set of points" since they satisfy all constraints which are true in the template such as straight lines and distances between them measured with the cross ratio. But, from the infinite sets of points which satisfy the template constraints, are the computed set of points the unique right solution? A lot of sets of points exist which satisfy all the template constraints and there is not any decisive factor which defines the computed set of points as the unique right solution. Since the distance between points is measured with the cross ratio, any set of points arranged in straight lines could satisfy the cross ratio and they could be a solution for Eq. (12). Taking the points of the template and projecting them using a random homography, they will result in a set of points which satisfy template constraints and therefore they could be the solution for Eq. (12). To make the solution unique, it will be necessary to know some of the parameters of the camera, but this is not the case.

The corrected set of points cannot be defined as the unique right solution but, they can be defined as the best undistorted approach to distorted points in the image. It can be tested that corrected points are closer to the distorted points if they are compared with existing undistortion methods. Fig. 4(a) shows the image of the template using a low-cost smart camera VCM50 from vision components. This image has been undistorted using two existing undistortion methods and results have been compared

with the metric distortion correction described in this paper. Fig. 4(b) shows the corrected image if the pin-hole and lens distortion model are computed together using the method described by Zhang in [6] which is free available (FCCM). Fig. 4(c) shows the result if the lens distortion is calibrated alone using the non-metric lens distortion correction based in straight lines presented by Ahmed in [24] (NMLDC). Fig. 4(d) shows the undistorted image corrected with the metric method described here (MLDC). The computed lens distortion model is represented in Table 1. If camera lens distortion is computed together with the pin-hole model, since both models are hard coupled, results do not represent the camera performance correctly. Undistorted points in the image are far away of the original distorted points. If camera lens distortion is computed alone using the NMLDC method, the computed model must be reduced to avoid instabilities as shown in [24]. In this case, the distortion centre defined as, $\boldsymbol{c}=(u_0, v_0)$ is computed with the pin-hole model and it is not improved with the lens distortion calibration since an instability occurs. Using FCCM or NMLDC methods, undistorted points do not represent the distorted image correctly since they are far away of the distorted points. In case of metric lens distortion calibration, the corrected image represents the distorted image accurately since undistorted points are closer to the distorted ones, if they are compared with FCCM or NMLDC. This means that corrected points represent image distortion accurately. There is not any criterion to define the corrected points as the unique right solution, but they are the undistorted points closer to the right ones, compared with existing methods. Undistorted points are always closer to the distorted ones, since the Levenberg–Marquardt algorithm starts with the distorted points trying to satisfy all template constrains. Eq. (12) has infinite solutions, but the non-linear searching ends in a local minimum closer to initial searching points. This fact makes that the Levenberg–Marquardt with the starting solution gets in the most cases an appropriate solution.

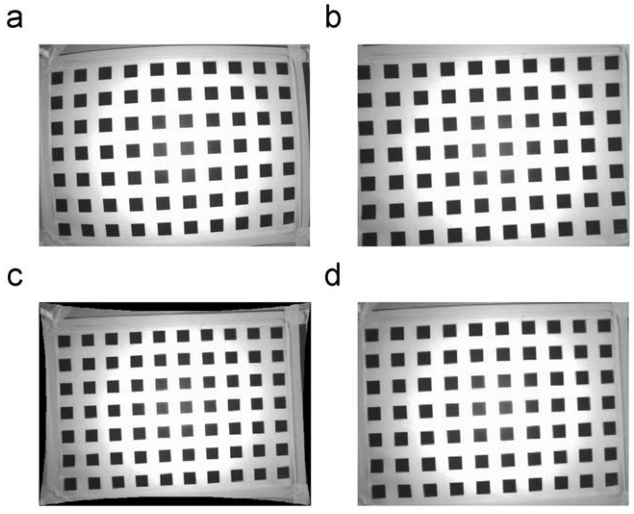

Fig. 4. (a) Image of the template using a low-cost smart camera VCM50 from Vision Components. (b) Undistorted image if the pin-hole and lens distortion model are computed together using the method described by Zhang in [6] (c) Corrected image if the lens distortion is calibrated alone using the non-metric lens distortion correction based in straight lines presented by Ahmed in [24] (d) Undistorted image corrected with the metric method described in this paper. Undistorted image is closer to the original distorted image if the metric lens distortion calibration is used.

4. Adjusting the camera lens distortion model

In the previous section, the method to correct distorted points in the image $\boldsymbol{q_{i,d}}$ and to find their right positions $\boldsymbol{q_{i,p}}$ has been described. This correction has come true using constrains that these sets of points must comply with since they are an image of a structure with known constrains. Next steps consist of computing the parameters of the lens distortion which models the mapping from distorted $\boldsymbol{q_{i,d}}$ points to undistorted ones $\boldsymbol{q_{i,p}}$. The polynomial distortion model was defined in (2). It contains radial, tangential and prism distortion. We only consider the second order polynomial distortion model, but it is straightforward to extend the result if higher order terms are used. Assuming second order terms for radial, tangential and prism distortions, the non-linear lens distortion model is parameterized with k_1, k_2, p_1, p_2, s_1, s_2, and u_0, v_0, where k_i models radial distortion, p_i represents tangential distortion, s_i is the prism distortion and u_0, v_0 represent the distortion centre. If second order polynomial distortion model is considered, the mapping between the observed distorted points $\boldsymbol{q_{i,d}}$ and the corrected computed ones $\boldsymbol{q_{i,p}}$ is given by the following expression:

$$\begin{aligned}\delta_{u,i} &= \Delta u_{i,d}\cdot(k_1\cdot r_{i,d}^2+k_2\cdot r_{i,d}^4)+p_1(3\Delta u_{i,d}^2+\Delta v_{i,d}^2)+2p_2\cdot\Delta u_{i,d}\cdot\Delta v_{i,d}+s_1\cdot r_{i,d}^2\\ \delta_{v,i} &= \Delta v_{i,d}\cdot(k_1\cdot r_{i,d}^2+k_2\cdot r_{i,d}^4)+2p_1\cdot\Delta u_{i,d}\cdot\Delta v_{i,d}+p_2(\Delta u_{i,d}^2+3\Delta v_{i,d}^2)+s_2\cdot r_{i,d}^2\end{aligned} \tag{13}$$

where $r_{i,d}^2=\Delta u_{i,d}^2+\Delta v_{i,d}^2$ is the distance of the point $\boldsymbol{q_{i,d}}=(u_{i,d}, v_{i,d})$ to the distortion centre defined as, $\boldsymbol{c}=(u_0, v_0)$, $\Delta u_{i,d}=u_{i,d}-u_0$, $\Delta v_{i,d}=v_{i,d}-u_0$, $\delta_{u,i}=u_{i,d}-u_{i,p}$, $\delta_{v,i}=v_{i,d}-v_{i,p}$. To solve the polynomial non-linear camera lens distortion model first a closed-form solution is given and afterwards it is improved with a non-linear optimization stage. For the closed-form solution step, the centre of the distortion $\boldsymbol{c}=(u_0, v_0)$ is the same as the principal point, considered in the centre of the image. From (13) we have two

Table 1
Parameters of non-linear polynomial distortion model computed with different methods.

Camera parameters	VCM 50			Basler		
	FCCM	NMLDC	MLDC	FCCM	NMLDC	MLDC
c_x (pixels)	310.46	329.07	315.24	325.48	327.97	323.16
c_y (pixels)	252.25	249.60	248.85	235.64	234.23	237.95
k_1 (mm^{-2})	−0.5680	0.008974	−0.07854	0.2356	0.06578	0.04589
k_2 (mm^{-4})	0.0987	−0.000256	0.00951	0.05689	0.000587	0.00101
p_1 (mm^{-2})	–	0.00947	0.00897	–	0.00458	0.00328
p_2 (mm^{-2})	–	0.00012	0.000985	–	0.000759	0.000476
Training error	0.957	2.071	1.547	0.789	1.464	1.112
Testing error	3.651	2.924	1.784	2.587	1.958	1.425

equations for each point in the image:

$$\begin{bmatrix} \Delta u_{i,d}\cdot r_{i,d}^2 & \Delta u_{i,d}\cdot r_{i,d}^4 & 3\Delta u_{i,d}^2+\Delta v_{i,d}^2 & 2\cdot\Delta u_{i,d}\cdot\Delta v_{i,d} & r_{i,d}^2 & 0 \\ \Delta v_{i,d}\cdot r_{i,d}^2 & \Delta v_{i,d}\cdot r_{i,d}^4 & 2\cdot\Delta u_{i,d}\cdot\Delta v_{i,d} & \Delta u_{i,d}^2+3\Delta v_{i,d}^2 & 0 & r_{i,d}^2 \end{bmatrix} \cdot \begin{bmatrix} k_1 \\ k_2 \\ p_1 \\ p_2 \\ s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} \delta_{u,i} \\ \delta_{v,i} \end{bmatrix} \tag{14}$$

Given $n\cdot m$ points of one image, we can stack all equations together to obtain a total of $2\cdot n\cdot m$ equations or in matrix form as $\boldsymbol{W}\cdot\boldsymbol{x}=\boldsymbol{w}$, where $x=[k_1, k_2, p_1, p_2, s_1, s_2]^T$. The linear least square solution is given by

$$x=(W^T\cdot W)^{-1}\cdot W^T\cdot w \tag{15}$$

The above solution is obtained through minimizing an algebraic distance which is not physically meaningful. We can refine through maximum likelihood inference. It can be obtained by minimizing the following functional:

$$J_{NLPD}=\sum_{i=1}^{n\cdot m}\left(\begin{array}{l}\|\delta_{u,i}-\Delta u_{i,d}\cdot(k_1\cdot r_{i,d}^2+k_2\cdot r_{i,d}^4)+p_1(3\Delta u_{i,d}^2+\Delta v_{i,d}^2)+2p_2\cdot\Delta u_{i,d}\cdot\Delta v_{i,d}+s_1\cdot r_{i,d}^2\|+\\ +\|\delta_{v,i}-\Delta v_{i,d}\cdot(k_1\cdot r_{i,d}^2+k_2\cdot r_{i,d}^4)+2p_1\cdot\Delta u_{i,d}\cdot\Delta v_{i,d}+p_2(\Delta u_{i,d}^2+3\Delta v_{i,d}^2)+s_2\cdot r_{i,d}^2\|\end{array}\right) \tag{16}$$

Minimizing (16) is a non-linear minimization problem, which is solved with the Levenberg–Marquart algorithm. It requires an initial guess of non-linear camera lens distortion parameters. k_1, k_2, p_1, p_2, s_1, s_2 can be obtained using the closed-form solution and u_0, v_0 is initialized with the principal point. The non-linear searching always converges to a solution also solving the distortion centre. No instability occurs. If more images are taken of the template, more points can be used in the calibration process. First they have been corrected according to the process described in Section 3.

5. Experimental results

The introduced metric non-linear lens distortion calibration method is compared with existing calibrations methods of non-linear lens distortion. To compare them, the polynomial distortion model is adjusted using three different lens distortion calibration techniques. The first one is done with the full camera calibration method (FCCM) which computes external (rotation and translation) and internal camera parameters at the same time. The internal parameters are the pin-hole camera parameters and also the second order radial distortion parameters. The centre of the radial distortion is considered at the principal point. First a closed-form solution is solved to obtain the starting values of the non-linear search. This method was introduced by Zhang [6] and it is freely available.

The second method is the non-metric calibration of camera lens distortion (NMLDC) introduced by Ahmed [24] which is based on the idea than straight lines in the image have to be straight. First, a closed-form solution is solved to compute the first order of radial distortion k_1 and the tangential one p_1, assuming the distortion centre in the central point of the image. Second, the solution is improved computing an extended set of parameters together with the distortion centre using a non-linear optimization technique. In this case, the non-linear optimization step may lead to instability if the distortion centre and distortion coefficients are computed together [24].

Finally, the lens distortion model is adjusted using the metric non-linear lens distortion calibration method (MLDC) introduced in this paper. In this case, second order radial and tangential distortion parameters are computed together with the distortion centre (k_1, k_2, p_1, p_2, u_0, v_0) following the method described in Section 4.1. The comparison of lens distortions calibration methods has been tested on computer simulated data and real data.

For the experimental setup, since we can increase or decrease the number of data designing a new planar template with more interesting points or also, taking several images of the same template from different locations, the largest number of data as possible has been used to fit the model. Twenty images of a planar template with 28×40 corners have been used similar to the one shown in Fig. 1. Fifteen are used to compute the parameters and five to test the results. The best one will be the one which gives lowest calibration error defined in Section 5.1.

5.1. Calibration error

In our experiments the camera lens distortion is calibrated using training data and then validated with a neutral test set covering a wide distance range. For evaluating both training and testing accuracies several methods exits. Basically, calibration accuracy can be measured by computing the discrepancy between pixel coordinates, $^*\boldsymbol{q}_{i,d}=(^*u_{i,d}, {}^*v_{i,d})$ projected from measured world coordinates by the camera model and observed pixel coordinates $\boldsymbol{q}_{i,d}=(u_{i,d}, v_{i,d})$ from captured images.

$$e_d=\frac{1}{n}\sum_{i=1}^{n}\sqrt{(^*u_{i,d}-u_{i,d})^2+(^*v_{i,d}-v_{i,d})^2} \tag{17}$$

This discrepancy can be measured with distorted point or undistorted point coordinates in the image. Also, the calibration error can be defined with the distance with respect to the optical ray, between 3D points in camera coordinates $\boldsymbol{p}_i=(x_i, y_i, z_i)$ and the optical rays back-projected from the corresponding undistorted image points on the camera image plane $\boldsymbol{q}_{p,i}=(u_{p,i}, v_{p,i})$. All these measurements are intuitive but sensitive to the digital image resolution, camera field-of-view and object-to-camera distance.

The normalized calibration error (NCE) proposed by Weng et al. [5] overcomes this sensitivity by normalizing the discrepancy between estimated and observed 3D points with respect to the area each back-projected pixel covers at a given distance from the camera (see Fig. 6 of [5]). In this case the distance of the point to the image is used to normalize the discrepancy between the measured and the observed coordinates. The NCE is calculated as follows:

$$e_n=\frac{1}{n}\sum_{i=1}^{n}\sqrt{\frac{(^*x_i-x_i)^2+(^*y_i-y_i)^2}{z_i(\alpha^{-2}+\beta^{-2})/12}} \tag{18}$$

where $^*\boldsymbol{p}_i=(^*x_i, {}^*y_i, z_i)$ represents 3D camera coordinates as estimated by back-projection form 2D undistorted pixel coordinates to depth z_i and $\boldsymbol{p}_i=(x_i, y_i, z_i)$ represents observed 3D camera coordinates computed from measured 3D world coordinates. The values of α and β can be calculated using the intrinsic camera parameters. If NCE ≈ 1, it indicates a good calibration in which residual distortion is negligible compared with image digitization noise at this depth. NCE $\gg 1$ reveals a poor calibration and NCE $\ll 1$

corresponds to a calibration whose error is lower, on average, than the digitization noise of a pixel at this depth.

The depth z_i of the point to the image plane is computed depending on whether simulated or real data are used. With simulated data, the depth of each point to the image plane is known since the calibration scene is simulated and the camera parameters are known accurately. In the case of real data, the camera must be calibrated to obtain the back-projection of the undistorted pixel coordinates. As before, calibration of the real camera is done using Zhang algorithm [6] using a free available toolbox. Knowing the camera model and 3D point's coordinates of the grid corners of the chessboard pattern, it is possible to know the depth of each point. Calibration is done using undistorted 2D points in the image since distortion is corrected earlier.

5.2. Computer simulations

The simulated camera has an image scale factor $s_x=1.5$ and an effective focal length $f=8$ resulting in pixel focal lengths of $\alpha=750$, $\beta=750$. The image resolution is 640×480 and with the principal point of the image at (320,240) pixels. The skew factor is set to $\gamma=0$.

The training model plane is a checkerboard with 1120 corners points (28×40) similar to Fig. 1 of 210 mm $\times$ 297 mm. The images are taken from 20 different orientations in front of the virtual camera. Images of the checker are taken according to the pin-hole model and then they are distorted using the non-linear polynomial distortion model (NLPD). Two levels of distortion are applied simulating soft and fish-eye distortion. In case of fish eye distortion a third-order radial distortion is simulated with the coefficients $k_1=-0.3\,\mathrm{mm}^{-2}$, $k_2=-0.045\,\mathrm{mm}^{-2}$. Second order tangential distortion is added with $p_1=-0.005\,\mathrm{mm}^{-2}$ and $p_2=-0.00025\,\mathrm{mm}^{-2}$. Prism distortion is discarded and the distortion centre is simulated at $u_0=305$, $v_0=257$ pixels. Soft distortion is simulated with $k_1=-0.2\,\mathrm{mm}^{-2}$, $k_2=0$, $p_1=p_2=0$ and $u_0=317$, $v_0=244$ pixels. 15 images are used to compute the parameters and 5 to test the results. Gaussian noise of mean 0 and standard deviation σ is added to the point's coordinates of the images. σ is added to the image in front of the camera when the model plane is parallel with the image plane and depending on the angle between the model and the image plane, σ is increased until 2σ with 70°. It simulates the influence of the point's detection in the image depending on the angle between the image plane and the model plane. No noise is added to test data.

Calibration accuracy vs. pixel coordinate noise: Figs. 5 and 6 show the decrease of accuracy (normalized calibration error) if pixel noise in training and test data increases. Fig. 5 shows the simulation results of soft lens distortion and Fig. 6 shows hard lens distortion. Non-metric calibration of camera lens distortion is more sensitive to pixel noise. This is because the reduced set of distortion parameters does not represent the lens distortions and errors cannot be compensated with the camera model. In case of full camera calibration method (FCCM) errors in distortion parameters are compensated by the camera pin-hole parameters. However, this compensation is only true for the camera location of the calibration process. Lens distortion and camera parameters satisfy the set of training data. When data change, the computed model does not satisfy them. Similar results were obtained by Sun et al. [30]. Finally, using the metric calibration method (MLDC), non-linear lens distortion is well represented by the computed model and when the data change the normalized error is similar. In this case, calibration of the camera pin-hole model is independent of calibration of lens distortion and it does not depend on the training data.

Calibration accuracy vs. number of points: Figs. 7 and 8 show the effect of the number of data in the calibration process. In absence of noise, a small number of training points can yield 100% accuracy. As some existing corner detection algorithms claim an accuracy of 0.1 pixels, Gaussian noise of zero mean and $\sigma=0.1$ is

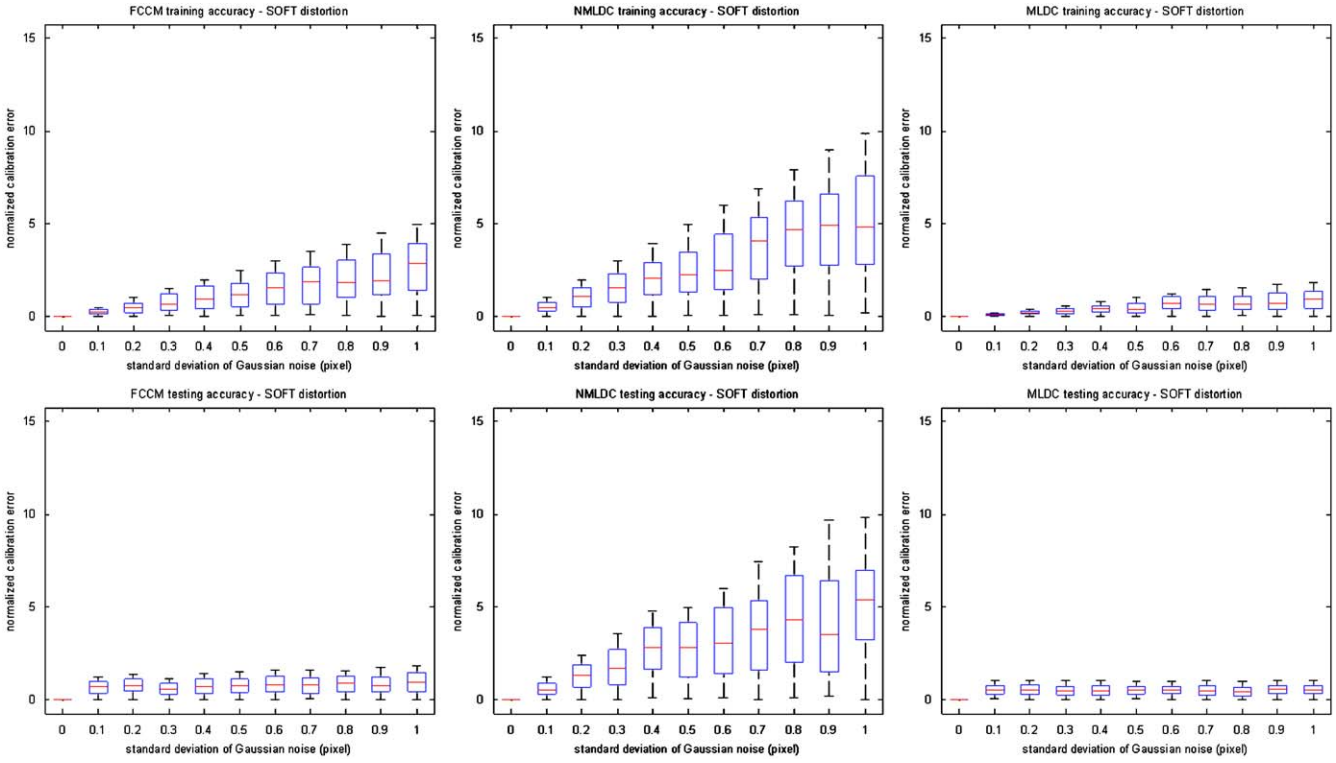

Fig. 5. Effects of pixel coordinate noise on calibration accuracy. Gaussian noise is added to training data but not to testing data. Soft lens distortion is simulated.

added to the pixel coordinates of the training data. The average results of 25 trials are illustrated in the Figs. 7 and 8. Again, the non-metric calibration of camera lens distortion shows higher errors.

5.3. Real data

We have used two hardware set-ups to test the accuracy of the distortion calibration depending on the distortion level. The first

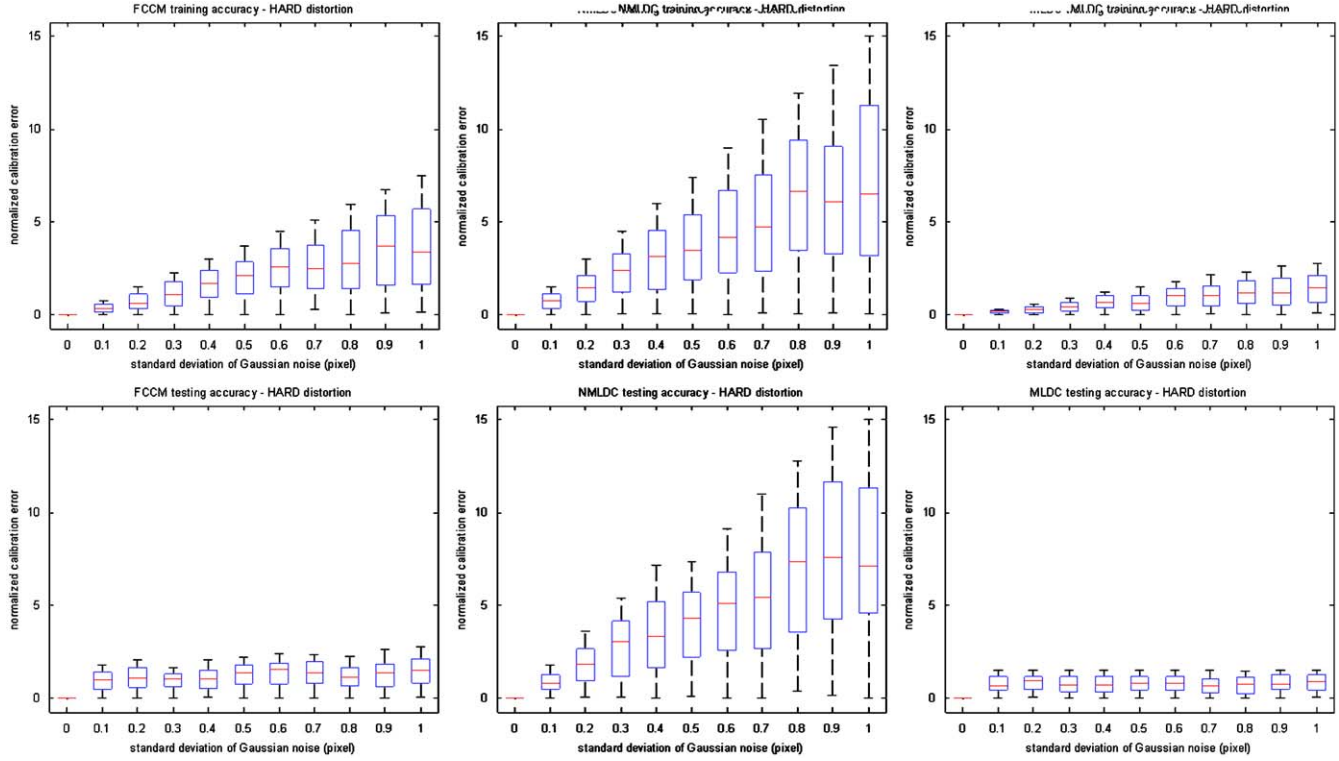

Fig. 6. Effects of pixel coordinates noise on calibration accuracy. Gaussian noise is added to training data but not to testing data. Hard lens distortion is simulated.

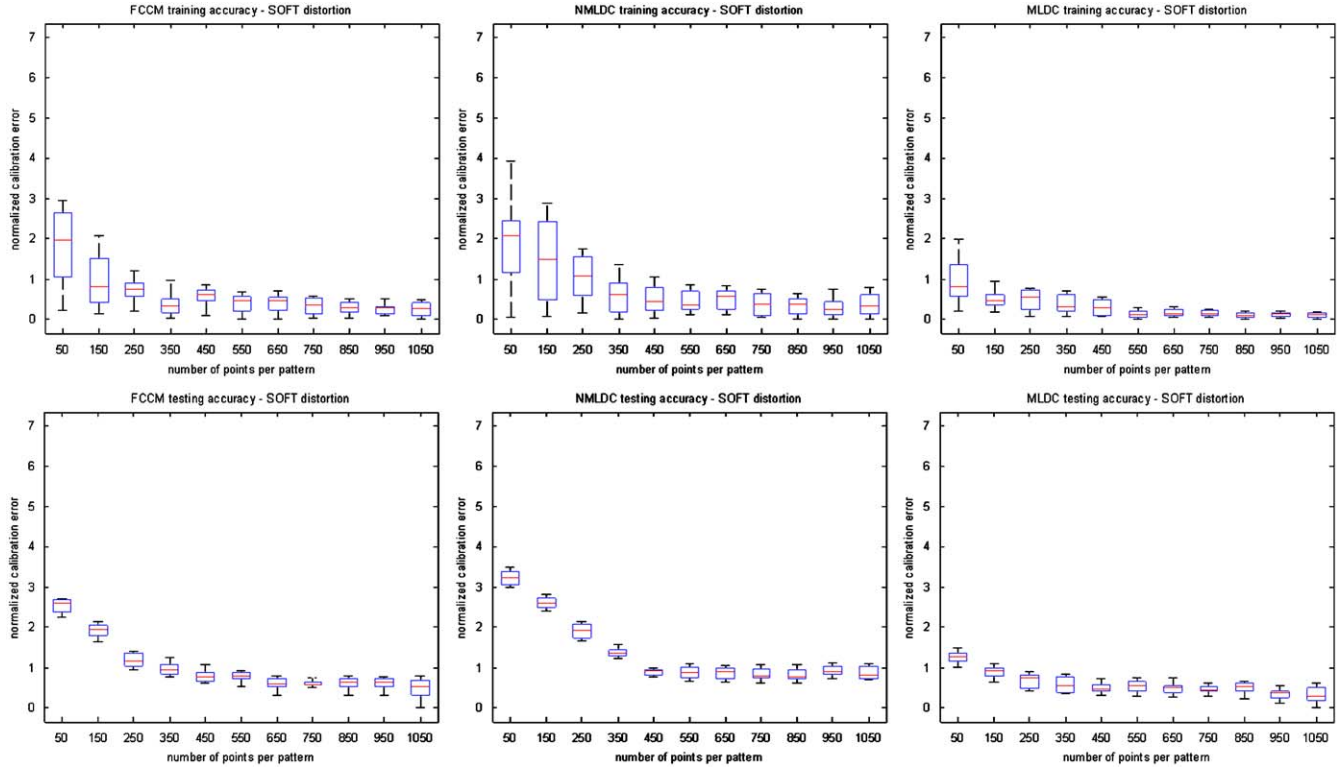

Fig. 7. Effects of training data quantity on calibration accuracy. Gaussian noise of $\sigma = 0.1$ is added to training data but not to testing data. Soft lens distortion is simulated.

one is a low-cost smart camera VCM50 from Vision Components which is not designed for either accuracy or quality applications. The acquired image is 640×480 pixels with much distortion caused by the cheap lens included in the camera. Other images have been acquired with a firewire Basler camera. In this case a 16 mm lens has been mounted and images of 640×480 pixels have been captured to compare the calibration of lens distortion with the same image resolution in both cameras. The optimization

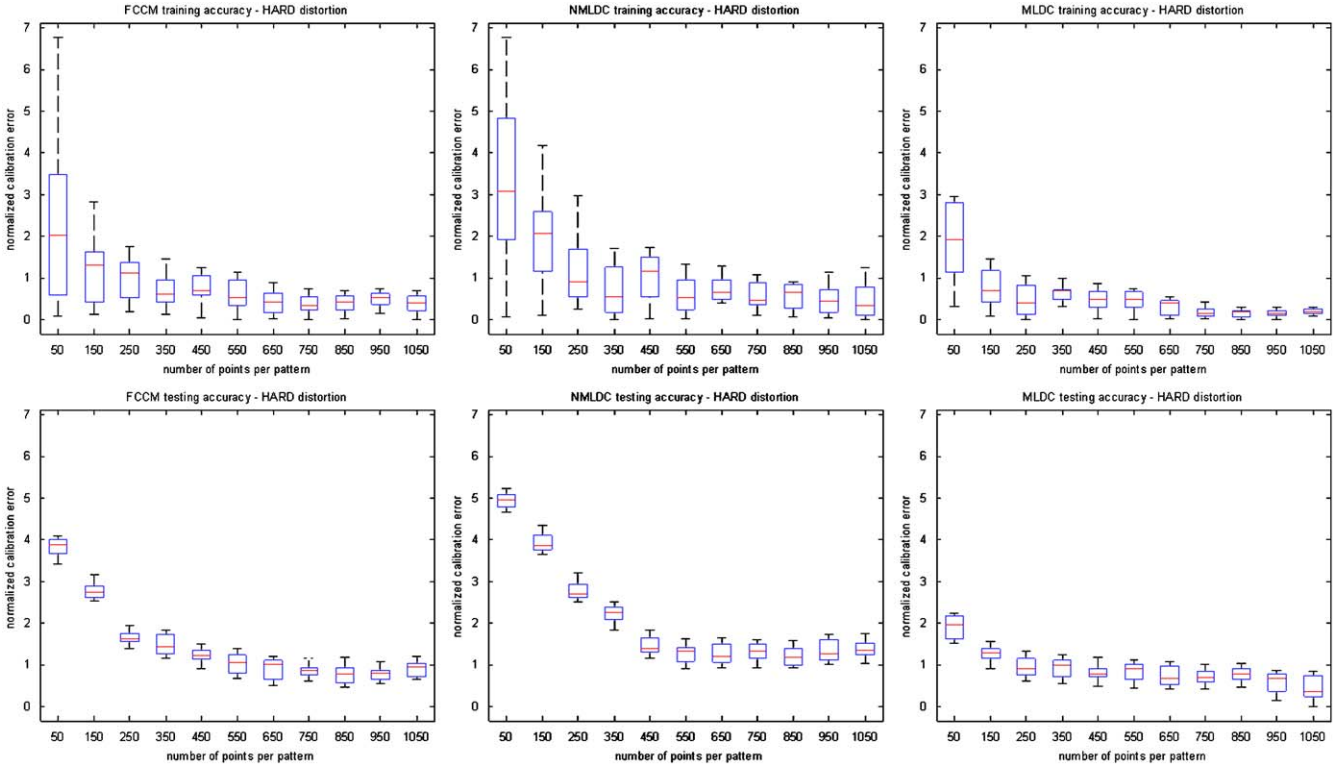

Fig. 8. Effects of training data quantity on calibration accuracy. Gaussian noise of $\sigma = 0.1$ is added to training data but not to testing data. Hard lens distortion is simulated.

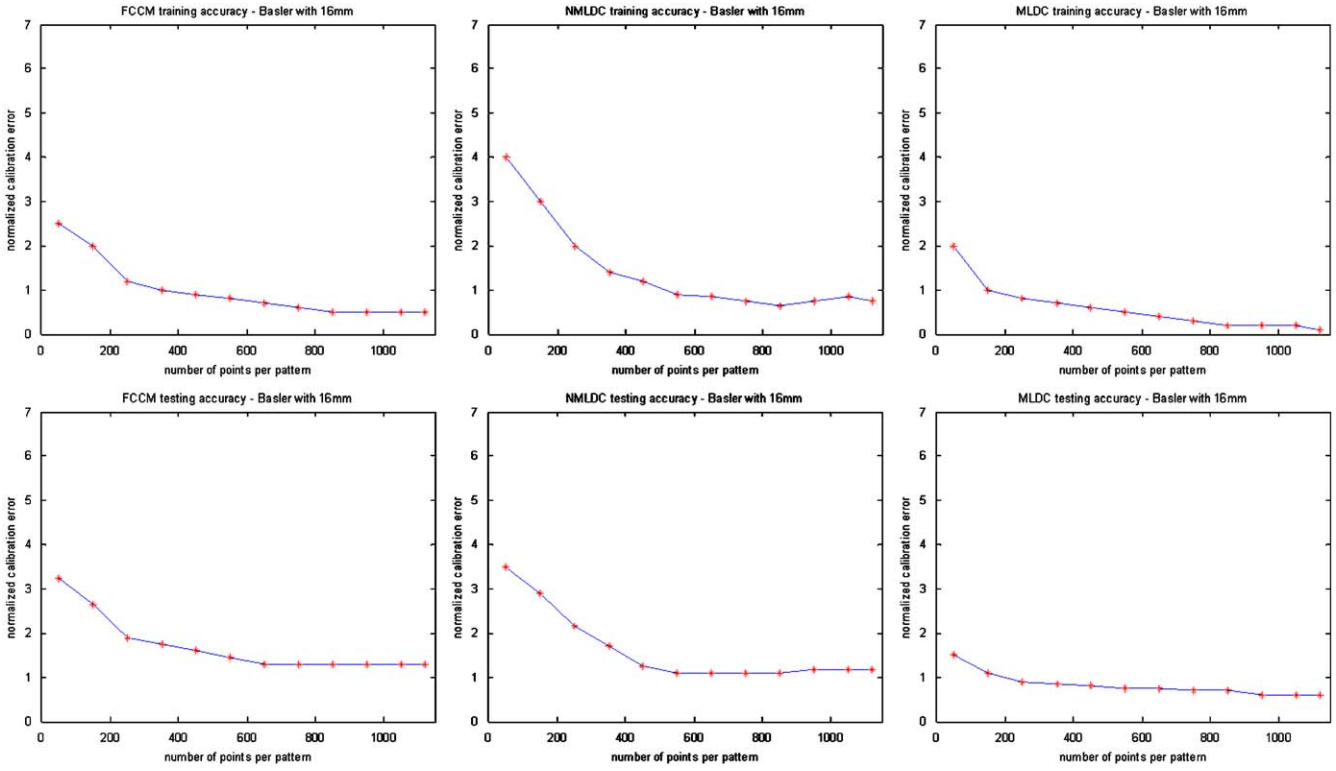

Fig. 9. Effects of training data quantity on calibration accuracy with a real camera Basler with 16 mm lens mounted.

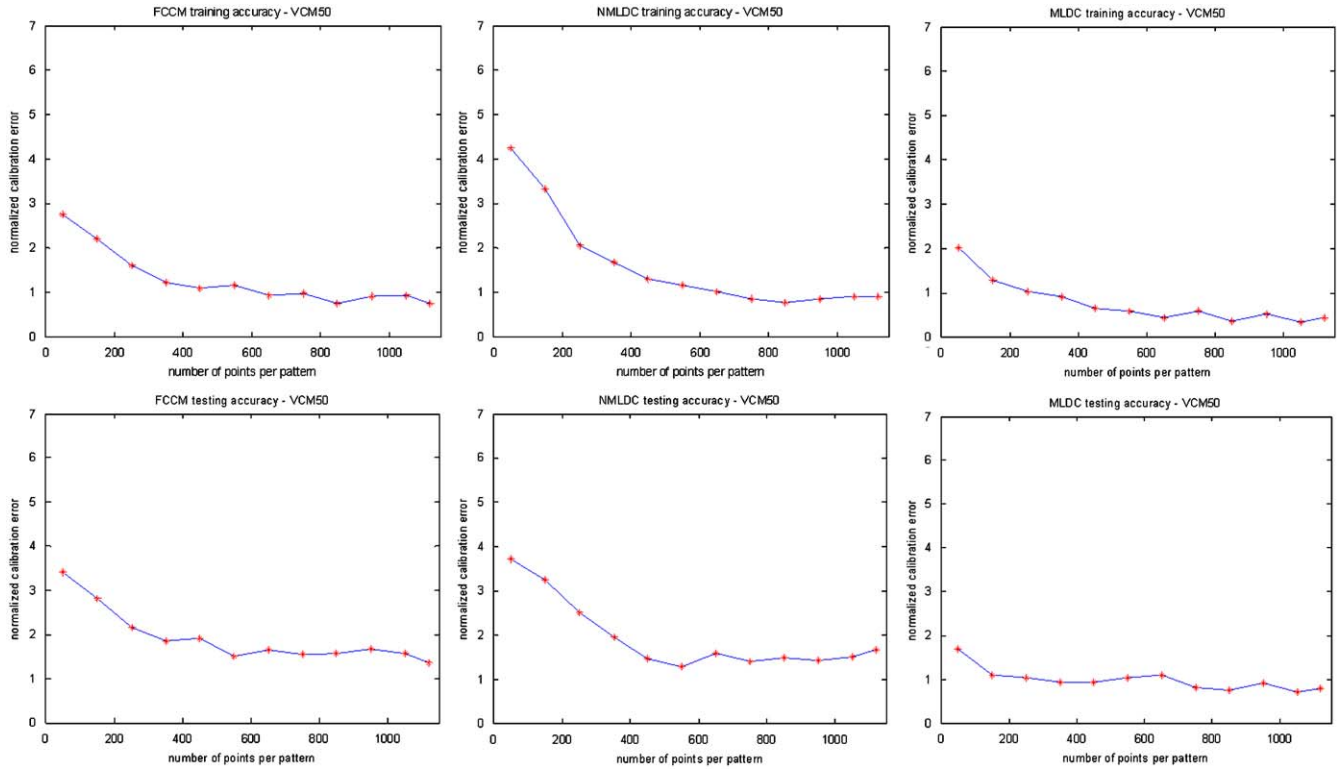

Fig. 10. Effects of training data quantity on calibration accuracy with the real camera VCM50.

step has been performed using the subroutine 'lsqnonlin' from MatLaboratory The training data are generated by printing a chessboard pattern of 28×40 grid corners onto letter-size sheets and viewed at 20 different orientations at roughly 45° with respect camera plane. This produced 20 data sets of 1120 points. Fifteen are used to compute the parameters and five to test the results. In this case, the calibration method and distortion model are compared.

Calibration technique: On the one hand, to compare the calibration methods, the non-linear polynomial model is computed with all of them. The computed parameters are presented in Table 1. The full camera calibration method obtains better results in the training step. This is because lens distortion calibration errors are compensated with the camera pin-hole parameters. After, when data changes with new images, computed parameters of the pin-hole and lens distortion models does not fit them. If lens distortion is calibrated isolated of the camera pin-hole model, training errors are similar to the testing errors. In this case, computed parameters represent the camera behaviour accurately. Comparing the isolated calibration methods, normalized error is smaller if the lens distortion is adjusted with the metric calibration method. Moreover, if the undistorted image is compared, better performance is computed with the metric lens distortion calibration as shown in Fig. 4. Undistorted image is closer to the original distorted image.

Calibration technique accuracy vs. number of points: Figs. 9 and 10 show the effect of the number of data in the calibration process changing the calibration technique. In this case one calibration is done with each set of points. Normalized error is shown in Fig. 10 for the VCM50 camera and Fig. 9 for the Basler. Results are quite similar to the simulation step. The full camera calibration method obtains low normalized calibration error if training data are used to test it. If testing data are used, the normalized calibration error increases. This is because with the full camera calibration method, parameters are adjusted to satisfy the set of training data and does not really represent the lens distortion model and camera model. When camera parameters are computed separately of lens distortion parameters they adjust testing and training data if a complete model of lens distortion is adjusted. This can be done by using the metric lens distortion calibration method. The non-metric one computes a reduced model which works well if low distortion lenses are used. Figs. 9 and 10 show this effect.

6. Conclusions

A metric camera lens distortion calibration method has been described which improves the performance of existing lens distortion calibration methods. Up to now, lens distortion was computed together with camera parameters or using some existing lens distortion non-metric or self-calibration methods. If lens distortion is computed together with camera parameters, both sets of parameters satisfy the training data since errors are compensated each others. However, when the camera location or scene changes, the computed model does not work as well as it worked in the calibration step. Moreover, the undistorted image suffers the coupling between the pin-hole and the lens distortion models. To improve it, the lens distortion parameters must be computed isolated from the camera parameters using some of the existing non-metric lens distortion calibration methods, which compute a reduced set of the distortion model parameter and not always under stable conditions. To obtain a better performance lens distortion model, a metric lens distortion calibration method has been proposed which computes a full set of parameters of the camera lens distortion model. This metric lens distortion calibration is a two step process where first, distorted points in the image are corrected to satisfy all geometric constrains of the calibration template, and second, using both set of distorted and undistorted points, lens distortion model is adjusted. The method does not need any extra calibration template since training data to resolve

this metric lens distortion calibration are a planar checker board template which can also be used afterwards for the camera calibration process.

Acknowledgements

This work was financially supported by the Spanish government (CICYT Project number DPI2006-15320-C03-01) and European Community FEDER funds. We would like to thank the R&D&I Linguistic Assistance Office, Universidad Politecnica de Valencia (Spain), for granting financial support for the linguistic revision of this paper.

References

[1] S. Shih, Y. Hung, W. Lin, When should we consider lens distortion in camera calibration, Pattern Recognition 28 (1995) 447–461.
[2] D.C. Brown, Decentering distortion of lenses, Photogrammetric Engineering and Remote Sensing 24 (1966) 555–566.
[3] D.C. Brown, Close-range camera calibration, Photogrammetric Engineering and Remote Sensing 42 (1971) 855–866.
[4] R. Tsai, A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-self TV camera lenses, IEEE Journal of Robotics and Automation. RA-3 (1997) 323–344.
[5] J. Weng, P. Cohen, M. Herniou, Camera calibration with distortion models and accuracy evaluation, IEEE Transactions on Pattern Analysis and Machine Intelligence 14 (1992) 965–980.
[6] Z. Zhang, A flexible new technique for camera calibration, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2000) 1330–1334.
[7] F. Devernay, O. Faugeras, Straight lines have to be straight, Machine Vision Applications 13 (2001) 14–24.
[8] C. McGlone, E. Mikhail, J. Bethel, Manual of Photogrammetry, fifth ed., American Society of Photogrammetry and Remote Sensing, 2004.
[9] A. Basu, S. Licardie, Alternative models for fish-eye lenses, Pattern Recognition Letters 16 (1995) 433–441.
[10] S. Shah, J.K. Aggarwal, Intrinsic parameter calibration procedure for a (high distortion) fish-eye lens camera with distortion model and accuracy estimation, Pattern Recognition 29 (1996) 1775–1778.
[11] S.S. Beauchemin, R. Bajcsy, Modelling and removing radial and tangential distortions in spherical lenses, Multi Image Analysis, Lecture Notes in Computer Science, vol. 2023, 2001, pp. 1–21.
[12] L. Ma, Y.Q. Chen, K.L. Moore, Flexible camera calibration using a new analytical radial undistortion formula with application to mobile robot localization, IEEE International Symposium on Intelligent Control (2003).
[13] J. Mallon, P.F. Whelan, Precise radial un-distortion of images, in: Proceedings of the 17th International Conference on Pattern Recognition, 2004.
[14] P. Sturm, S. Ramalingam, A generic concept for camera calibration, in: Proceedings of the Fifth European Conference on Computer Vision, 2004.
[15] R. Hartley, S. Kang, Parameter-free radial distortion correction with centre of distortion estimation, in: Proceedings of the 10th International Conference on Computer Vision, 2005.
[16] W. Faig, Calibration of close-range photogrammetry systems: mathematical formulation, Photogrammetric Engineering and Remote Sensing 41 (1975) 1479–1486.
[17] G. Wei, S. de Ma, A complete two-plane camera calibration method and experimental comparison, in: Proceedings of Fourth International Conference on Computer Vision, 1993, pp. 439–446.
[18] Z. Zhang, IEEE Transactions on Pattern Analysis and Machine Intelligence 26 (2004) 892–899.
[19] O. Faugeras, T. Luong, S. Maybank, Camera self-calibration: theory and experiments, European Conference on Computer Vision (1992) 321–334.
[20] G. Stein, Accurate internal camera calibration using rotation with analysis of sources of error, in: Proceedings of the Fifth International Conference on Computer Vision, 1995.
[21] R. Hartley, Projective reconstruction and invariants from multiple images, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (1994) 1036–1040.
[22] B. Prescott, G. McLean, Line-based correction of radial lens distortion, Graphical Models and Image Processing 59 (1997) 39–47.
[23] R. Swaminathan, S. Nayar, Non-metric calibration of wide-angle lenses and polycameras, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (2002) 1172–1178.
[24] M. Ahmed, A. Farag, Non-metric calibration of camera lens distortion: differential methods and robust estimation, IEEE Transactions on image processing 14 (2005) 1215–1230.
[25] S. Becker, V. Bove, Semi-automatic 3D model extraction from uncalibrated 2D camera views, Proceedings of Visual Data Exploration and Analysis II 2 (1995) 447–461.
[26] M. Penna, Camera calibration: a quick and easy way to detection of scale factor, IEEE Transactions on Pattern Analysis and Machine Intelligence 12 (1991) 1240–1245.
[27] Z. Zhang, On the epipolar geometry between two images with lens distortion, Proceedings of International Conference on Computer Vision and Pattern Recognition 1 (1996) 407–411.
[28] G. Stein, Lens distortion calibration using point correspondences, Proceedings of the International Conference on Computer Vision and Pattern Recognition (1997) 602–608.
[29] A. Fitzgibbon, Simultaneous linear estimation of multiple view geometry and lens distortion, IEEE International Conference on Computer Vision and Pattern Recognition (2001).
[30] W. Sun, J. Cooperstock, An empirical evaluation of factors influencing camera calibration accuracy using three publicly available techniques, Machine Vision and Applications 17 (2006) 51–67.
[31] C. Slama, Manual of Photogrammetry, fourth ed., American Society of Photogrammetry, 1980.
[32] J. Wang, F. Shi, J. Zhang, Y. Liu, A new calibration model of camera lens distortion, Pattern Recognition 41 (2008) 607–615.
[33] J. Wei, S. de Ma, Implicit and explicit camera calibration: theory and experiments, IEEE Transactions on Pattern Analysis and Machine Intelligence 16 (1994) 15–20.
[34] G. Zhang, J. He, X. Yang, Calibrating camera radial distortion with cross-ratio invariability, Optics and Laser Technology 35 (2003) 457–461.
[35] O. Faugeras, Three Dimensional Computer Vision, MIT Press, London, 1993.

About the Author—CARLOS RICOLFE-VIALA worked for one year in the University of Ulm (Germany) as research engineer on computer vision in 1999. He received the M.S. degree in 2000 in Electronics and Automatic Control Engineering in the Universidad Politecnica de Valencia. In 2006, he obtained his Ph.D. degree on computer vision and its applications. Now he has worked as a lecturer and research scientist in this university since 2000. His research interests are in image processing and computer vision, especially in motion estimation, feature detection and matching, camera calibration, 3D computer vision and intelligent system robot. He also cooperates with the Institute of Industrial Informatics and Automatic Control of Valencia.

About the Author—ANTONIO-JOSE-SANCHEZ-SALMERON received the B.E. in Computer Science degree in 1992, M.Sc. in CAD/CAM/CIM degree in 1999 and Ph.D. degree in Computer Science in 2001. He is currently an Associate Professor at the "Universidad Politecnica de Valencia" (UPV) at the "Departamento de Ingenieria de Sistemas y Automatica" (DISA) and belongs to the Robotics Research Group at the "Instituto de Automatica e Informatica Industrial" (ai2). His research interests focus on the areas of artificial vision and robotics.