

# Self-calibration of the distortion of a zooming camera by matching points at different resolutions

Selim BENHIMANE and Ezio MALIS

I.N.R.I.A.

Sophia-Antipolis, FRANCE

**Abstract**—This paper presents a new method for the self-calibration of the lens distortion of a zooming camera which appears for short focal lengths. The proposed technique does not need any special calibration pattern nor any prior knowledge about the environment. The key idea is to match points between a distorted image and an undistorted image taken at different resolutions. A new method for automatically matching points in the two images is proposed. The scale factor between the images is not needed for the matching algorithm. Matched points are used to compute invariants to the pinhole camera parameters. Then, lens distortion parameters are estimated in order to obtain the same invariants in both images. This approach is well suited to autonomous robotic vision applications. In fact, the self-calibration of the camera is done before moving the robot. Experiment with ground truth and tests on real images provide good results.

## I. INTRODUCTION

An important problem when considering autonomous robotic vision applications is the on-line self-calibration of the cameras. Consider, for example, the case of a mobile robot with a zooming camera mounted on-board. Self-calibration of the lens distortion for short focal length is necessary to extract geometric information and to control the movements of the robot. If the robot explores an unknown environment, standard calibration methods using known calibration patterns [9] [10] cannot be used. Similarly, if the robot navigates in an unstructured environment we cannot use methods that require the presence of straight lines in the scene [5] [1]. In that cases, a possible solution, proposed in [8], is to self-calibrate the distortion from point correspondences in multiple views. There are two major difficulties with this approach when considering autonomous robotic vision applications. The first problem is to obtain multiple views of the same scene. When a robot autonomously navigates in an unknown environment some obstacles can limit its movements. Thus, it would be preferable to self-calibrate the camera before moving the robot. The second problem is to obtain point correspondences in multiple views. Matching points between distorted images can be extremely difficult in an unstructured environment especially if geometric matching constraints (fundamental matrix, trilinear tensor) [11] cannot be used due to the presence of the distortion. In this paper, we propose a different solution to the self-calibration of the distortion which is specially adapted to the case of a zooming camera moving in unknown and unstructured environments. The technique proposed in the paper does not need any special calibration pattern nor the presence of structure in the scene

(like straight lines, for example). It is based on matching points between only two images at different resolutions. In this paper, a new matching method is proposed. The distorted image is taken with an unknown set of camera parameters, while the second one is taken from the same position with different camera parameters and it is supposed to be undistorted. One of the advantage of this method, is that the camera is stationary in the scene and we do not need to take a risk by moving the robot. Once point correspondences have been obtained, we use a new distortion measure based on the computation of invariants to camera parameters of the standard pinhole camera model. In the absence of lens distortion, the invariants computed from the two images should be the same. A difference in the invariants is observed in the presence of lens distortion. Thus, we estimate the best distortion parameters such that the distorted image can be corrected in order to verify a perspective projection model. The proposed self-calibration method has been tested on real images providing good results.

## II. THEORETICAL BACKGROUND

### A. Pinhole Camera Model

Let us suppose that the absolute frame coincides with the camera frame  $\mathcal{F}$ . A 3D point  $\mathcal{X}_j$  is projected to a virtual point  $\mathbf{m}_j = (x_j, y_j, 1)$  on a plane perpendicular to the optical axis of the camera:

$$\mathbf{m}_j \propto \begin{bmatrix} \mathbf{I}_{3 \times 3} & \mathbf{0}_{3 \times 1} \end{bmatrix} \mathcal{X}_j \quad (1)$$

The information given by a pinhole camera (which performs a perspective projection of 3D points in an image  $i$ ) is not the virtual point  $\mathbf{m}_j$  but an image point  $\mathbf{p}_{ij} = (u_{ij}, v_{ij}, 1)$ :

$$\mathbf{p}_{ij} = \mathbf{K}_i \mathbf{m}_j \quad (2)$$

where matrix  $\mathbf{K}_i$  contains the pinhole camera parameters:

$$\mathbf{K}_i = \begin{bmatrix} f_i & f_i s_i & u_{0i} \\ 0 & f_i r_i & v_{0i} \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

$f_i$  is the focal length in pixel,  $s_i$  is the skew,  $r_i$  is the aspect ratio and  $(u_{0i}, v_{0i})$  are the coordinates of the principal point (in pixels). Note that both intrinsic and extrinsic camera parameters vary when the camera is zooming [3]. However, we suppose that the pinhole model is valid for each setting of  $\mathbf{K}_i$  and that the scene is far enough to neglect the variation of the extrinsic parameters.

### B. Lens Distortion Model

In this paper, we consider the standard radial distortion model [7], but any other parametric distortion model could be used. Let  $\mathbf{p}_{ci} = (u_{ci}, v_{ci}, 1)$  be the unknown center of radial distortion of the image  $i$  (not necessarily equal to the principal point), and let  $\rho_{ij}$  be the distance of a distorted point  $\mathbf{p}_{ij}^d = (u_{ij}^d, v_{ij}^d, 1)$  from  $\mathbf{p}_{ci}$ :

$$\rho_{ij} = \|\mathbf{p}_{ij}^d - \mathbf{p}_{ci}\| = \sqrt{(u_{ij}^d - u_{ci})^2 + (v_{ij}^d - v_{ci})^2}$$

The undistorted coordinates  $\mathbf{p}_{ij}$  of an image point can be computed as:

$$\mathbf{p}_{ij} = \mathbf{p}_{ij}^d + (\mathbf{p}_{ij}^d - \mathbf{p}_{ci})(d_{1i}\rho_{ij}^2 + d_{2i}\rho_{ij}^4 + \dots) \quad (4)$$

The camera distortion parameters of image  $i$  are the center of radial distortion  $\mathbf{p}_{ci}$  and the vector  $\mathbf{d}_i = (d_{1i}, d_{2i}, \dots)$ .

### III. INVARIANTS TO PINHOLE PARAMETERS

We are interested in the calibration of the camera distortion parameters. Once the lens distortion has been corrected, the pinhole camera parameters can be computed with any self-calibration technique. For this reason, we use measures that are invariants with respect to pinhole camera parameters  $\mathbf{K}_i$  under the assumption of perspective projection [4]. Suppose that at least  $n > 3$  points are present in the scene. The 3D points project to the virtual points  $\mathbf{m}_j$  and to the corresponding observed image points  $\mathbf{p}_{ij}$ . Consider now the following  $(3 \times 3)$  matrices:

$$\mathbf{S}_{pi} = \frac{1}{n} \sum_{j=1}^n \mathbf{p}_{ij} \mathbf{p}_{ij}^\top \quad \text{and} \quad \mathbf{S}_m = \frac{1}{n} \sum_{j=1}^n \mathbf{m}_j \mathbf{m}_j^\top$$

Matrix  $\mathbf{S}_{pi}$  is a positive symmetric matrix which can be computed from the image points while matrix  $\mathbf{S}_m$  is also a positive symmetric matrix but it is not directly measurable. These positive symmetric matrices can be written using the Cholesky decomposition:

$$\mathbf{S}_{pi} = \mathbf{T}_{pi} \mathbf{T}_{pi}^\top \quad \text{and} \quad \mathbf{S}_m = \mathbf{T}_m \mathbf{T}_m^\top$$

where  $\mathbf{T}_m$  and  $\mathbf{T}_{pi}$  are non-singular upper triangular matrices. From equation (2), it follows that the two triangular matrices are related by:

$$\mathbf{T}_{pi} = \mathbf{K}_i \mathbf{T}_m \quad (5)$$

where  $\mathbf{T}_m$  is a triangular matrix not depending of the pinhole camera parameters  $\mathbf{K}_i$ . The non-singular  $(3 \times 3)$  matrix  $\mathbf{T}_{pi}$  can be used to define the following projective transformation:

$$\mathbf{q}_{ij} = \mathbf{T}_{pi}^{-1} \mathbf{p}_{ij} \quad (6)$$

where the transformed points  $\mathbf{q}_{ij}$  are invariants to pinhole camera parameters. Indeed, from equations (2), (5) and (6), we obtain that the pinhole camera parameters  $\mathbf{K}_i$  are factored out:

$$\mathbf{q}_{ij} = \mathbf{T}_m^{-1} \mathbf{K}_i^{-1} \mathbf{K}_i \mathbf{m}_j = \mathbf{T}_m^{-1} \mathbf{m}_j$$

As an example, consider the two images in Figure 1 taken by a stationary zooming camera but with two different pinhole camera parameters  $\mathbf{K}_1$  and  $\mathbf{K}_2$ . These images

will be used as a ground truth in the paper, while real unstructured images will be used for the experimental results in section VI-A. In this example, 16 points are selected in each image. The points  $\mathbf{p}_{1j}$  in the first image are represented by the green circles in Figure 1(a). The points  $\mathbf{p}_{2j}$  in the second image are represented by red crosses in the Figure 1(b). Note that we do not need to find correspondences between the points in order to compute the invariants but we only need the same set of points in both images. From the two set of points, we compute the triangular matrices  $\mathbf{T}_{p1}^{-1}$  and  $\mathbf{T}_{p2}^{-1}$ . Thus, the invariants for the first image are:

$$\mathbf{q}_{1j} = \mathbf{T}_{p1}^{-1} \mathbf{p}_{1j} = \mathbf{T}_m^{-1} \mathbf{K}_1^{-1} \mathbf{K}_1 \mathbf{m}_j = \mathbf{T}_m^{-1} \mathbf{m}_j$$

while the invariants for the second image are:

$$\mathbf{q}_{2j} = \mathbf{T}_{p2}^{-1} \mathbf{p}_{2j} = \mathbf{T}_m^{-1} \mathbf{K}_2^{-1} \mathbf{K}_2 \mathbf{m}_j = \mathbf{T}_m^{-1} \mathbf{m}_j$$

Thus, it is evident that  $\mathbf{q}_{1j} = \mathbf{q}_{2j}$  ( $\forall j$ ). We can also use matrices  $\mathbf{T}_{p1}$  and  $\mathbf{T}_{p2}$  to reproject the points of one image to the scale of the other. For example, matrix  $\mathbf{T}_{p1} \mathbf{T}_{p2}^{-1} = \mathbf{K}_1 \mathbf{K}_2^{-1}$  can be used to reproject all points from image 2 into the scale of image 1:

$$\tilde{\mathbf{p}}_1 = \mathbf{T}_{p1} \mathbf{T}_{p2}^{-1} \mathbf{p}_2 = \mathbf{K}_1 \mathbf{K}_2^{-1} \mathbf{p}_2$$

Figure 1(c) shows that the image 2 is perfectly reprojected into the scale of image 1, proving that the matrix  $\mathbf{T}_{p1} \mathbf{T}_{p2}^{-1}$  has been correctly estimated (i.e. the invariants  $\mathbf{q}_{1j}$  and  $\mathbf{q}_{2j}$  are also correctly estimated). The red crosses extracted in image 2 are reprojected into the green circles extracted in image 1. In the presence of lens distortion, the invariant points  $\mathbf{q}_{1j}$  and  $\mathbf{q}_{2j}$  will not be the same. This idea is the basis of our algorithm for the correction of the distortion.

### IV. MATCHING AT DIFFERENT RESOLUTIONS

The invariants to camera pinhole parameters can also be used in order to perform matching between two images taken with a stationary zooming camera but with two different sets of pinhole camera parameters. Contrarily to classical matching algorithms, the method proposed does not directly carry out a point-to-point matching by estimating intrinsic and extrinsic parameter variation. It makes use of the invariants to the camera parameters in order to discard points that are not present in both images. Once, a same set of points is extracted from the two images, the point-to-point matching becomes evident. Obviously, the lens distortion modifies the invariants and this must be taken into account in the matching algorithm.

#### A. Detecting Interest Points

In order to perform the automatic extraction of interests we use the Harris corner detector [2]. We briefly remind the bases of the detector since they will be used in the following. The Harris corner detector uses the following matrix  $\mathbf{M}(\mathbf{p}_{ij})$ :

$$\mathbf{M}(\mathbf{p}_{ij}) = \sum_{\mathbf{x} \in W(\mathbf{p}_{ij})} g(\mathbf{x}) \begin{bmatrix} I_u^2(\mathbf{x}) & I_u I_v(\mathbf{x}) \\ I_u I_v(\mathbf{x}) & I_v^2(\mathbf{x}) \end{bmatrix}$$

where  $W$  is a window centered on  $\mathbf{p}_{ij}$ ,  $I(\mathbf{p})$  is the intensity of the pixel  $\mathbf{p}$ ,  $I_u = \frac{\partial I}{\partial u}$ ,  $I_v = \frac{\partial I}{\partial v}$  and  $g$  is a Gaussian centered in  $\mathbf{p}_{ij}$ . If the eigen values of the matrix  $\mathbf{M}$  are close  $\lambda_1 \approx \lambda_2$  and both are higher than a certain threshold then  $\mathbf{p}_{ij}$  is an interest point. To avoid the calculus of the eigen values for each point, we use the following detection score:

$$s_2(\mathbf{p}_{ij}) = \frac{\det}{tr} \times \frac{\det}{tr^2} = \frac{(\lambda_1 \lambda_2)^2}{(\lambda_1 + \lambda_2)^3} \quad (7)$$

where  $\det$  and  $tr$  are determinant and the trace of the matrix  $\mathbf{M}$ . If the score of a point  $\mathbf{p}_{ij}$  is higher than a certain threshold (generally taken equal to 10% of the maximum of the highest score over all image points) then this point is detected as an interest point. The detector has selected 422 points in the first image (see Figure 2(a)) and 433 points in the second image (see Figure 2(b)). Obviously, we cannot use all extracted points to calculate the invariants defined in the previous section. Figure 2(c) shows that in this case, the reprojection of image 2 to the scale of image 1 is not possible. Indeed, we do not have the same points extracted from both images. However, many points are repeated in the two images. Consequently, pairs of correspondent points are closer to each other and points that are not present in both images are isolated. The next step of our matching algorithm is to eliminate unrepeated points and to find a set of points present in both images.

### B. Eliminating Unrepeated Points

The goal of the elimination algorithm is to discard points that are not present in both images. Two criteria are used to discard points. First of all, if the selected points are present in both images, then the reprojection of the points from one image to the other should be the same (as shown in the previous section). Secondly, correspondent points should have similar photometric properties. To each point  $\mathbf{p}_{ij}$ , we associate 3 photometric descriptors:

- $s_1(\mathbf{p}_{ij})$  the mean of grey level values of a window centered on  $\mathbf{p}_{ij}$  (a  $(3 \times 3)$  window for example).
- $s_2(\mathbf{p}_{ij})$  the Harris detection score given in (7).
- $s_3(\mathbf{p}_{ij})$  a scale invariant descriptor.

We use a particular case of one of the scale invariant descriptors proposed in [6]. The descriptor can be computed using the gradient  $\nabla I$  and the Laplacian  $\Delta I$  of the image. In order to have a continuous function everywhere, we define the invariant descriptor as follows:

$$\begin{aligned} s_3(\mathbf{p}_{ij}) &= 0 & \text{if } \|\nabla I(\mathbf{p}_{ij})\|^2 = 0 \\ s_3(\mathbf{p}_{ij}) &= \frac{\|\nabla I(\mathbf{p}_{ij})\|^2}{|\Delta I(\mathbf{p}_{ij})|} & \text{if } \|\nabla I(\mathbf{p}_{ij})\|^2 < |\Delta I(\mathbf{p}_{ij})| \\ s_3(\mathbf{p}_{ij}) &= \frac{|\Delta I(\mathbf{p}_{ij})|}{\|\nabla I(\mathbf{p}_{ij})\|^2} & \text{if } \|\nabla I(\mathbf{p}_{ij})\|^2 > |\Delta I(\mathbf{p}_{ij})| \end{aligned}$$

Let us suppose that the Harris detector selected  $n$  points  $\mathbf{p}_{1k}$  ( $k \in \{1, \dots, n\}$ ) in the image 1 and  $m$  points

$\mathbf{p}_{2j}$  ( $j \in \{1, \dots, m\}$ ) in the image 2. If many points are present in both images, the reprojections of points  $\mathbf{p}_{2j}$  into image 1 are close to their corresponding points as shown in image 2(c). Isolated points are less likely to have a corresponding point and can be eliminated as follows:

*Step 1* : Compute the matrix  $\mathbf{T}_{p1}$  using the set of points  $\mathbf{p}_{1k}$  and the matrix  $\mathbf{T}_{p2}$  using the set of points  $\mathbf{p}_{2j}$ . Then, compute the reprojection of the points  $\mathbf{p}_{2j}$  into the image 1:  $\tilde{\mathbf{p}}_{1j} = \mathbf{T}_{p1} \mathbf{T}_{p2}^{-1} \mathbf{p}_{2j}$ .

*Step 2* : Eliminate  $\mathbf{p}_{1k}$  ( $\forall k \in \{1, \dots, n\}$ ) if  $\nexists j \in \{1, \dots, m\}$  such that:

- 1)  $\|\mathbf{p}_{1k} - \tilde{\mathbf{p}}_{1j}\| < \nu$  ;
- 2)  $|s_1(\mathbf{p}_{1k}) - s_1(\mathbf{p}_{2j})| < \tau_1$ .
- 3)  $\frac{\min\{s_2(\mathbf{p}_{1k}), s_2(\mathbf{p}_{2j})\}}{\max\{s_2(\mathbf{p}_{2j}), s_2(\mathbf{p}_{1k})\}} > \tau_2$  ;
- 4)  $\frac{\min\{s_3(\mathbf{p}_{1k}), s_3(\mathbf{p}_{2j})\}}{\max\{s_3(\mathbf{p}_{2j}), s_3(\mathbf{p}_{1k})\}} > \tau_3$  ;

where  $\nu$  is a variable threshold which defines the distance from which the point  $\mathbf{p}_{1k}$  is considered as an isolated point (at the first iteration of the algorithm  $\nu$  is fixed to  $\nu_{max}$ ), and  $\tau_1$ ,  $\tau_2$  and  $\tau_3$  are thresholds fixed such that  $\tau_1 \in [0, 255]$ ,  $\tau_2 \in [0, 1]$  and  $\tau_3 \in [0, 1]$ . The threshold  $\tau_1$  fixes a maximum difference between the grey level values of two corresponding points. The threshold  $\tau_2$  sets the closeness of the scores of the Harris detector. The threshold  $\tau_3$  sets the closeness of the scale invariant descriptor. Similarly,  $\forall j \in \{1, \dots, m\}$ , eliminate  $\mathbf{p}_{2j}$  if  $\nexists k \in \{1, \dots, n\}$  such that the conditions (1), (2), (3) and (4) are verified. If at least one point is eliminated then go to *Step 2*. Else, continue.

*Step 3* : If  $\nu < \nu_{min}$ , then stop. Else, reduce  $\nu$  with a factor  $\gamma$  ( $0 < \gamma < 1$ ) and go to *Step 1*.

At each iteration, some isolated points are discarded. Thus, we obtain better estimates of matrices  $\mathbf{T}_{p1}$  and  $\mathbf{T}_{p2}$  and the reprojections of points  $\mathbf{p}_{2j}$  into image 1 become closer to their corresponding points. The thresholds used to find the matching in Figure 3 are:  $\nu_{max} = 100$ ,  $\nu_{min} = 25$ ,  $\tau_1 = 64$ ,  $\tau_2 = \tau_3 = 0.5$ ,  $\gamma = 0.9$ .

### C. Final Matching

At the end of the elimination algorithm, we use only points that do not present any ambiguity in order to obtain the final best transformations  $\mathbf{T}_{p1}$  and  $\mathbf{T}_{p2}$ . With these two matrices, we are able to compute the best reprojection  $\tilde{\mathbf{p}}_{1j}$ . Finally, a point  $\mathbf{p}_{1k}$  is matched to  $\mathbf{p}_{2j}$ , if  $\mathbf{p}_{1k}$  is the closest point to  $\tilde{\mathbf{p}}_{1j}$ . Figure 3 shows the results obtained with our matching algorithm. In Figure 3(a) and (b) are plotted the 289 points matched in the two images, while Figure 3(c) shows that the second image can be perfectly reprojected into the first one. If we compare Figure 2(c) and

Figure 3(c), we can notice that several isolated points have been eliminated.

#### D. Matching in presence of distortion

In the presence of lens distortion, the matching algorithm still provides good results. However, less strict thresholds must be used since it is not possible any more to compute exact invariants to pinhole camera parameters. As a consequence less corresponding points are found. Figure 4(a) shows an artificial distortion of the image in Figure 3(a). This distorted image will be used as ground truth in the experimental results. The 247 correspondences found by the algorithm are plotted in Figure 4(a) and (b). As expected, due to the distortion the reprojections plotted in Figure 4(c) are not as accurate as the reprojections plotted in Figure 3(c). On the other hand, the correspondences are correct and can be used to self-calibrate the lens distortion parameters as proposed in the next section.

### V. CORRECTING THE DISTORTION

If the projection model is perfectly perspective, the invariants computed in the first and the second images should be exactly the same  $\mathbf{q}_{2j} = \mathbf{q}_{1j}$ . Due to image distortion in one of the images, there is a difference between the invariants. We suppose that the lens distortion of a zooming camera appears only for short focal lengths and can be neglected for long focal lengths. We will consider in our examples that only one image is distorted. The undistorted image coordinates can be computed from a parametric model of the distortion. For simplicity, we consider here only the radial distortion model described in section II-B. Let  $\mathbf{p}_{c1}$  and  $\mathbf{d}_1$  be respectively the center and the coefficients of distortion for the image 1. The undistorted points in image 1 are a function of  $\mathbf{p}_{c1}$  and  $\mathbf{d}_1$  (see equation (4)):

$$\mathbf{p}_{1j} = f(\mathbf{p}_{1j}^d, \mathbf{p}_{c1}, \mathbf{d}_1)$$

After computing the matrices  $\mathbf{T}_{p1}$  and  $\mathbf{T}_{p2}$ , we obtain the invariants to pinhole camera parameters:

$$\mathbf{q}_{1j} = \mathbf{T}_{p1}^{-1} \mathbf{p}_{1j} = \mathbf{q}_{1j}(\mathbf{p}_{c1}, \mathbf{d}_1)$$

and:

$$\mathbf{q}_{2j} = \mathbf{T}_{p2}^{-1} \mathbf{p}_{2j}$$

Thus, we can define a cost function which depends on the difference between the invariants computed from the two images:

$$\mathcal{E}(\mathbf{p}_{c1}, \mathbf{d}_1) = \sum_{i=1}^n \|\mathbf{q}_{2j} - \mathbf{q}_{1j}(\mathbf{p}_{c1}, \mathbf{d}_1)\|^2$$

The distortion parameters can be found with a nonlinear optimization method by minimizing the cost function:

$$\min \mathcal{E}(\mathbf{p}_{c1}, \mathbf{d}_1)$$

Experimental results show that the proposed cost function has a large convergence range. In particular, very rough guesses of the center of distortion are needed.

## VI. EXPERIMENTAL RESULTS

### A. Experiment with Ground Truth

Consider the images in Figure 5 (b) and Figure 5 (c). In order to have a ground truth, we have artificially distorted the image 2 in Figure 5 (c) and obtained the image in Figure 5 (a). The center of distortion is the center of the image  $\mathbf{p}_c = (384, 288, 1)$ . It does not necessarily correspond to the principal point. We distort the image such that the distorted image can be corrected using only one coefficient  $\mathbf{d}_1 = (d_{11})$  where  $d_{11} = +6 \cdot 10^{-7}$ . The corresponding distorted image is given in Figure 5 (a). The green circles in Figure 5 (a) and the red crosses in Figure 5 (b) are the correspondences obtained by our automatic matching algorithm. Once the correspondences have been

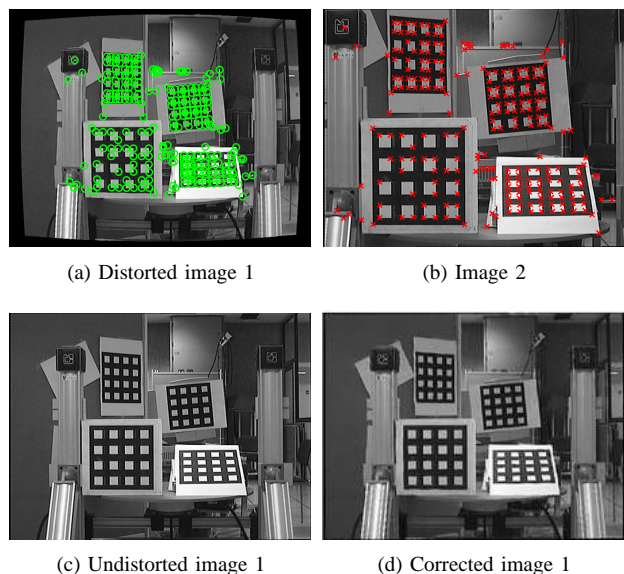


Fig. 5. Correction of lens distortion with ground truth.

obtained, the distortion parameters can be estimated as described in section V. In this experiment, the unknown distortion parameters are the center of radial distortion  $\mathbf{p}_c = (u_c, v_c, 1)$  and the distortion coefficient  $d_{11}$ . Theoretically, we could consider more than one distortion coefficient, but we would obtain only small improvements in the correction of the distortion. The starting points for the optimization algorithm are  $\mathbf{p}_c(0) = (434, 208, 1)$  and  $d_{11} = 0$ .

	true values	measured values
$u_c$	384	385.5
$v_c$	288	289.0
$d_{11}$	$+6.00 \cdot 10^{-7}$	$+5.25 \cdot 10^{-7}$

TABLE I  
SELF-CALIBRATION RESULTS FOR THE LENS DISTORTION  
PARAMETERS ARTIFICIALLY ADDED TO THE IMAGE.

Despite the initial center of distortion is guessed very far from the true one, Table I shows that the distortion parameters are accurately estimated. In fact, the measured

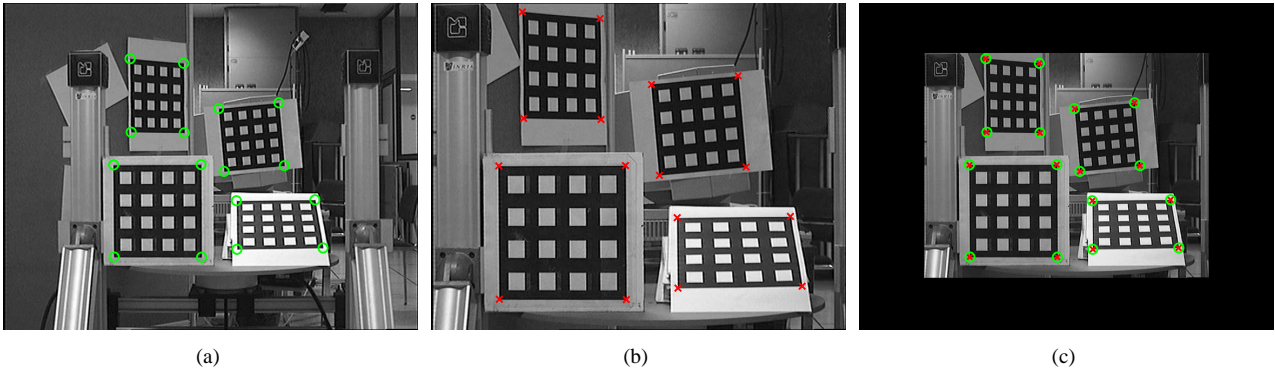


Fig. 1. Interest points manually extracted from two images with different resolutions taken by a zooming camera.

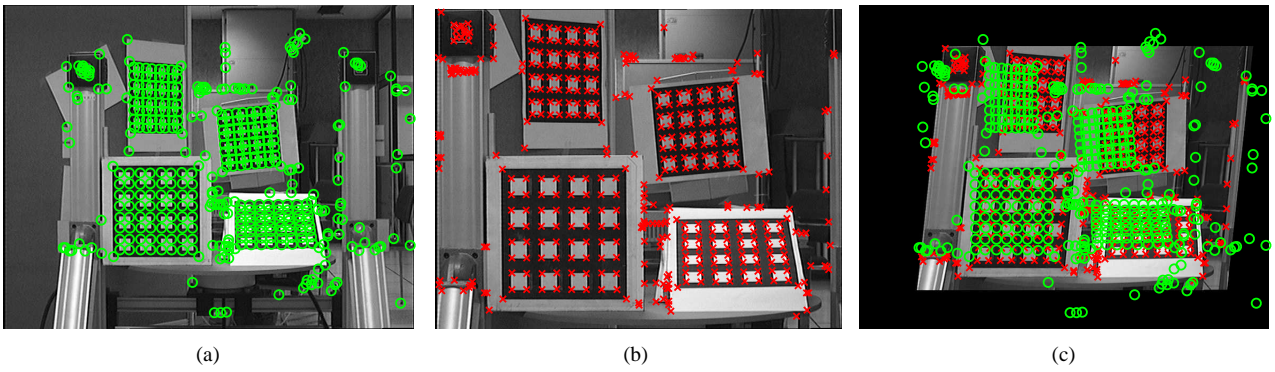


Fig. 2. Two different set of points are extracted using a Harris corner detector. The second image cannot be reprojected to the scale of the first one.

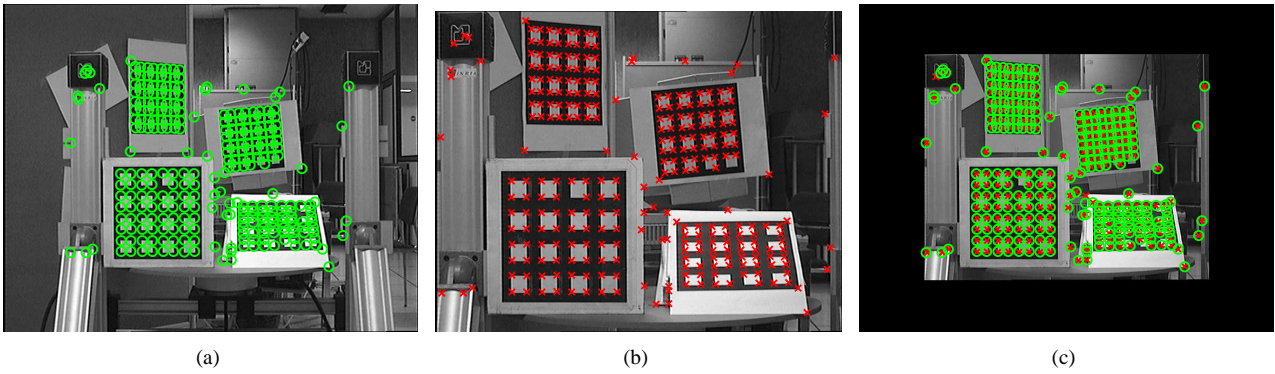


Fig. 3. The final set of points matched between the two images.

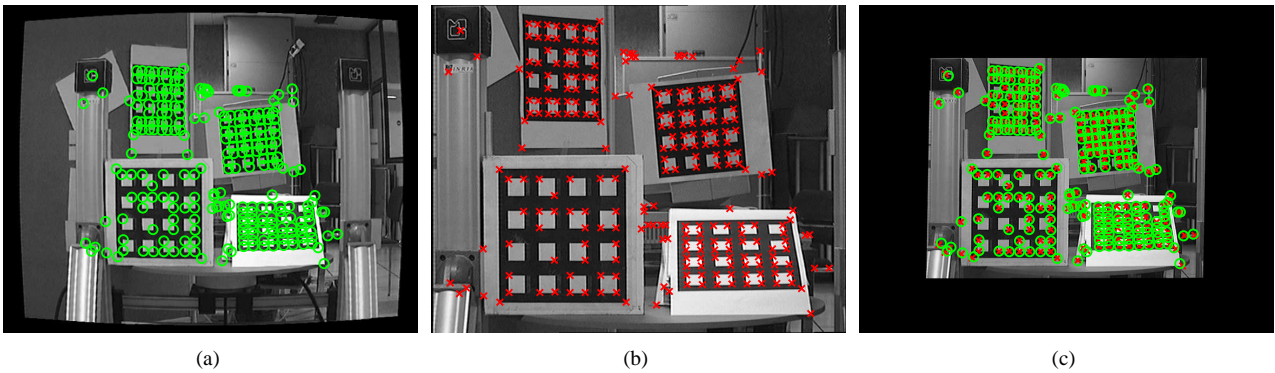


Fig. 4. Matching in the presence of distortion.



values of the distortion parameters are very close to the true values. Using these parameters to correct the distorted image, we obtain the image in Figure 5 (d).

### B. Experiment without Ground Truth

In the experiments without ground truth, we use images taken in an unknown and unstructured environment. An image of flowers is taken with a short focal length (see Figure 6(a)) and one image of the same scene is taken at higher resolution (see Figure 6(b)). It is not evident at all to see the effects of the lens distortion in the image in Figure 6(a) since there are not straight lines in it. In order to prove that the image is indeed distorted, we have taken another image with the same set of camera parameters. Figure 7(a) shows the effects of lens distortion on the borders of the poster (which should be straight lines). The matching algorithm proposed in section IV finds 109 correspondences which are plotted in Figure 6(a) and (b). The distortion parameters estimated with our algorithm using the images of the flowers are  $u_c = 319$ ,  $v_c = 242$  and  $d = 3.6 \cdot 10^{-7}$ . They are used to correct the image of the poster. The corrected image in Figure 7(b) proves that the distortion parameters were accurately estimated since the borders of the poster are now straight lines.

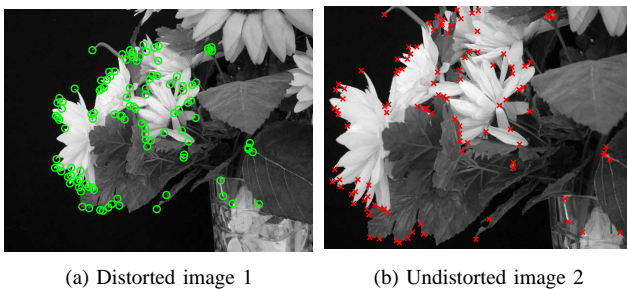


Fig. 6. Correction of the radial distortion using unstructured images.



Fig. 7. Correction of the radial distortion using unstructured images and verification using a structured image with straight lines.

## VII. CONCLUSIONS

A simple method for self-calibration has been proposed in the paper. The method does not need any special calibration pattern nor the presence of special structures (e.g. straight lines) in the environment. Our goal is to self-calibrate a zooming camera mounted on a robot exploring an unknown and unstructured environment. With our method, the robot does not need to move during the self-calibration of the distortion. Only two images of a stationary zooming camera are needed. By matching a distorted image with an undistorted one at different resolutions, it is possible to correct the distorted image. The matching and the self-calibration algorithm make use of invariants to pinhole camera parameters. The method could be improved by also using multiple views taken from different positions.

### ACKNOWLEDGMENT

This work has been funded by the project AEROB of ROBEA: “Robotique et Entités Artificielles”.

### REFERENCES

- [1] F. Devernay and O. Faugeras. Straight lines have to be straight. Automatic calibration and removal of distortion from scenes of structured environments. *Machine Vision and Applications*, 13(1), pp. 14–24, 2001.
- [2] C. Harris and M. Stephens. A combined corner and edge detector. *Proceedings of the 4th Alvey Vision Conference*, pp. 147–151, 1988.
- [3] M. Li and J.-M. Lavest. Some aspects of zoom lens camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(11), pp. 1105–1110, 1996.
- [4] E. Malis. A Unified Approach to Model-based and Model-free Visual Servoing. *European Conference on Computer Vision*, (4), pp. 433–447, 2002.
- [5] B. Prescott and G.F. McLean. Line-based correction of radial lens distortion. *Graphical Models and Image Processing*, 59(1), pp. 39–47, 1997.
- [6] C. Schmid and R. Mohr. Local grey value invariants for image retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(5), pp. 530–534, 1997.
- [7] C.C. Slama, C. Theurer, and S.W. Henriksen, editors. *Manual of photogrammetry*. American Society of Photogrammetry, 1980.
- [8] G.P. Stein. Lens distortion calibration using point correspondences. *IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 602–608, 1997.
- [9] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, (4), pp.323–344, 1987.
- [10] J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(10), pp. 965–980, 1992.
- [11] Z. Zhang, R. Deriche, O. Faugeras, and Q.-T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Technical Report 2273, INRIA, 1994.