Camera Calibration Using Two Concentric Circles

Francisco Abad*, Emilio Camahort, and Roberto Vivó

Universidad Politécnica de Valencia, Camino de Vera s/n, Valencia 46021, Spain {fjabad, camahort, rvivo}@dsic.upv.es, WWW home page: http://www.sig.upv.es

Abstract. We present a simple calibration method for computing the extrinsic parameters (pose) and intrinsic parameters (focal length and principal point) of a camera by imaging a pattern of known geometry. Usually, the patterns used in calibration algorithms are complex to build (three orthogonal planes) or need a lot of features (checkerboard-like pattern). We propose using just two concentric circles that, when projected onto the image, become two ellipses. With a simple mark close to the outer circle, our algorithm can recover the full pose of the camera. Under the perfect pinhole camera assumption, the pose and the focal length can be recovered from just one image. If the principal point of the camera has to be computed as well, two images are required. We present several results, using both synthetic and real images, that show the robustness of our method.

1 Introduction

In the past two decades, several methods have been proposed for calibrating a camera by taking images of a pattern with known geometry. First in photogrammetry and then in computer vision, researchers have developed methods to recover a camera's extrinsic parameters (position and orientation) and intrinsic parameters (focal length and principal point). Those methods usually require expensive laboratory settings, or use complex fiducials [1].

In order to take the computer vision from the laboratory to the home user, robust, inexpensive and effective techniques are needed. In this paper, we present an algorithm that easily recovers the pose and the focal length of a camera by taking a single photo of a simple calibration pattern. We use a pattern made of two concentric circles of known radii, usually printed on a sheet of paper. We show how this pattern can be used in a simple setup to recover the camera parameters. Our method can be applied to camera tracking and related problems like robotics, entertainment and augmented reality.

This paper is organized as follows. The next section presents previous work in the field of camera calibration with circular markers. Section 3 presents the

^{*} This work was partially funded by the Programa de Incentivo a la Investigación of the Polytechnic University of Valencia, and by project TIC2002-04166-C03-01 of Spanish Ministry of Science and Technology

theoretical model and mathematical foundations of our work. In the following section we present some results of our method, and discuss the tests we run with both synthetic and real data. Our paper finishes with some conclusions and directions for future work.

2 Previous Work

Early work that used conics for computer vision applications was reported in [2–4]. Circular markers have been extensively used in tracking applications due to their robustness properties [6,7]. Kim et al. [8,9] proposed a calibration method using two concentric circles. Their algorithm requires some initial information about the camera to get an initial value for the intrinsic matrix. They define a cost function on the calibration parameters and minimize it. This method only recovers the normal of the marker's supporting plane.

Another method that recovers the supporting plane of the circles was proposed in [10]. The method computes the plane's normal and a point on it expressed in camera coordinates. The method assumes that the principal point is at the center of the image.

Unlike the previous methods, our algorithm does not require any a priori information about the camera parameters to calibrate it. Furthermore, we recover the pose (the full rotation matrix and the translation vector) using a simple marker. Finally, we also compute the position of the principal point.

3 Calibrating the Camera

3.1 Detecting the Marker

Our marker is composed of two concentric circles or radii r_1 and r_2 , and an exterior mark that intersects with a circle of radius r_3 (see Fig. 1).

The ellipses can be automatically recovered from an image by applying standard methods in Computer Vision. Pixel chains are extracted from the image and ellipses are fitted with, e.g., Fitzgibbon's algorithm [5]. See for example [7] for an explanation of an automatic extraction algorithm. To find the X axis mark, a circle of radius r_3 has to be projected using the same (unknown) camera as the other two. In an Appendix we explain how to project a circle of arbitrary radius concentric to two circles whose projections are known.

3.2 Pinhole Projection of a Circle

The pinhole camera configuration (assuming zero skew and square pixels) is usually described using an intrinsic parameter matrix (A), that describes the focal length and principal point (see Fig. 2), and an extrinsic parameter matrix (M), that establishes the camera pose (position and orientation) from a given global coordinate system:



Fig. 1. Design of our fiducial

Fig. 2. Pinhole camera in the scene

$$A = \begin{bmatrix} f \ 0 \ u_0 \\ 0 \ f \ v_0 \\ 0 \ 0 \ 1 \end{bmatrix} \qquad M = \begin{bmatrix} R_{11} \ R_{12} \ R_{13} \ T_X \\ R_{21} \ R_{22} \ R_{23} \ T_Y \\ R_{31} \ R_{32} \ R_{33} \ T_Z \end{bmatrix} . \tag{1}$$

In Fig. 2, the world coordinate system (WCS) has its origin at the center of the concentric circles. Those circles are in the $X_W Y_W$ plane of the WCS, so the $Z_{\rm W}$ axis is perpendicular to them. The projection operator (P), that computes the image pixel coordinates that corresponds to a 3D point in WCS is P = AM.

Given a point X in WCS, equation $\lambda x = PX$ computes its homogeneous coordinates in the image coordinate system. The two circles are located on the plane $Z_{\rm W} = 0$, so we can write:

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X_{\mathrm{W}} \\ Y_{\mathrm{W}} \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11}X_{\mathrm{W}} + p_{12}Y_{\mathrm{W}} + p_{14} \\ p_{21}X_{\mathrm{W}} + p_{22}Y_{\mathrm{W}} + p_{24} \\ p_{31}X_{\mathrm{W}} + p_{32}Y_{\mathrm{W}} + p_{34} \end{bmatrix} .$$

If we assume that the image coordinate system is centered at the principal point of the image, then $u_0 = 0$ and $v_0 = 0$ in (1) and we can write (see [10]):

$$X_{\rm W} = \frac{\boldsymbol{x}^t \left(\boldsymbol{R_2} \times \boldsymbol{T} \right)}{\boldsymbol{x}^t \boldsymbol{R_3}} \quad \text{and} \quad Y_{\rm W} = \frac{\boldsymbol{x}^t \left(\boldsymbol{T} \times \boldsymbol{R_1} \right)}{\boldsymbol{x}^t \boldsymbol{R_3}} \quad , \tag{2}$$

where $\boldsymbol{x} = \begin{bmatrix} u \ v \ f \end{bmatrix}^t$, and $M = \begin{bmatrix} \boldsymbol{R_1} \ \boldsymbol{R_2} \ \boldsymbol{R_3} \ \boldsymbol{T} \end{bmatrix}$. In the WCS, the exterior circle of radius r_2 has coordinates $C(X_{\rm W}, Y_{\rm W}) = X_{\rm W}^2 + Y_{\rm W}^2 - r_2^2 = 0$. Substituting (2) in this equation and factoring, we can express the exterior circle in terms of the image coordinate system as follows:

$$C'(x,y) = Ax^{2} + Bxy + Cy^{2} + Dx + Ey + F = 0 .$$
(3)

3.3 Recovering the Circles' Projected Center

Under perspective projection any conic is transformed into another conic. Specifically, circles are transformed into elipses when imaged by a camera. The projected center of the original circle, however, does not generally coincide with the center of the ellipse in the image. The projected center of the circles has to be computed in order to recover the normal to the supporting plane. The direction of the $Z_{\rm W}$ axis in the camera coordinate system (or R_3) can be computed as follows [2, 3]:

$$\begin{bmatrix} R_{13} \\ R_{23} \\ R_{33} \end{bmatrix} = \pm N \left\{ Q \begin{bmatrix} x_c \\ y_c \\ f \end{bmatrix} \right\} \quad , \tag{4}$$

where (x_c, y_c) are the coordinates of the projected circle center in the image coordinate system (see Fig. 2), N represents the normalization to a unit vector, and Q is the matrix that describes the ellipse, as defined in [3]:

$$Q = \begin{bmatrix} A & B/2 & D/2f \\ B/2 & C & E/2f \\ D/2f & E/2f & F/f^2 \end{bmatrix} .$$
 (5)

Parameters A to F are those defined in (3) and f is the focal length. Two methods to recover the projected center of two concentric circles can be found in [8] and [9]. In the Appendix we present our own original method.

3.4 Recovering the Pose

Each parameter of the ellipse in (3) can be expressed in terms of f^2 , f^2/T_Z^2 and a constant term by substituting (5) in (4) [10]. This derivation uses the properties of the rotation matrices and the following relations derived from the pinhole camera model in Fig. 2:

$$T_{\rm X} = \frac{T_{\rm Z} x_{\rm C}}{f}$$
 and $T_{\rm Y} = \frac{T_{\rm Z} y_{\rm C}}{f}$. (6)

The result is

$$Q' = \begin{bmatrix} \alpha_1^2 & -\alpha_1^2 r_2^2 & (\alpha_1^2 + \alpha_2^2) y_{\rm C}^2 + 2\alpha_2 \alpha_3 y_{\rm C} + \alpha_3^2 \\ 2\alpha_1 \alpha_2 & -2\alpha_1 \alpha_2 r_2^2 - 2(\alpha_2 \alpha_3 x_{\rm C} + (\alpha_1^2 + \alpha_2^2) x_{\rm C} y_{\rm C} + \alpha_1 \alpha_3 y_{\rm C}) \\ \alpha_2^2 & -\alpha_2^2 r_2^2 & (\alpha_1^2 + \alpha_2^2) x_{\rm C}^2 + 2\alpha_1 \alpha_3 x_{\rm C} + \alpha_3^2 \\ -2\alpha_1 (\alpha_1 x_{\rm C} + \alpha_2 y_{\rm C}) - 2\alpha_1 \alpha_3 r_2^2 & -2\alpha_3 (\alpha_3 x_{\rm C} + \alpha_2 x_{\rm C} y_{\rm C} - \alpha_1 y_{\rm C}^2) \\ -2\alpha_2 (\alpha_1 x_{\rm C} + \alpha_2 y_{\rm C}) - 2\alpha_2 \alpha_3 r_2^2 & -2\alpha_3 (-\alpha_2 x_{\rm C}^2 + \alpha_1 x_{\rm C} y_{\rm C} + \alpha_3 y_{\rm C}) \\ (\alpha_1 x_{\rm C} + \alpha_2 y_{\rm C})^2 & -\alpha_3^2 r_2^2 & \alpha_3^2 (x_{\rm C}^2 + y_{\rm C}^2) \end{bmatrix}$$
(7)

where:

$$\alpha_1 = \frac{2Ax_{\rm C} + By_{\rm C} + D}{2}$$
 $\alpha_2 = \frac{Bx_{\rm C} + 2Cy_{\rm C} + E}{2}$ $\alpha_3 = \frac{Dx_{\rm C} + Ey_{\rm C} + 2F}{2}$

Therefore, (3) can be expressed as:

$$C'(x,y) = \left(\begin{array}{c} f^2 \\ f^2/Tz^2 \\ 1 \end{array} \right)^t G = 0 \quad , \tag{8}$$

where $G = \begin{bmatrix} x^2 & xy & y^2 & xy & 1 \end{bmatrix}^t$. The unknowns to be computed are f^2 and f^2/T_Z^2 , so we rearrange (8) to leave the constant terms on the right-hand side of the expression:

$$\left(\begin{bmatrix} q_{11} \ q_{21} \ q_{31} \ q_{41} \ q_{51} \ q_{61} \\ q_{12} \ q_{22} \ q_{32} \ q_{42} \ q_{52} \ q_{62} \end{bmatrix} G \right)^t \begin{bmatrix} f^2 \\ f^2 \\ T^2_Z \end{bmatrix} = - \begin{bmatrix} q_{13} \ q_{23} \ q_{33} \ q_{43} \ q_{53} \ q_{63} \end{bmatrix} G \quad , \quad (9)$$

where q_{ij} is the element of row *i*, column *j* of matrix Q' in (7). Given *N* points of the ellipse in the image we can build an *N*-degree over-determined system WX = B:

$$\begin{bmatrix} W_{11} & W_{12} \\ W_{21} & W_{22} \\ \vdots & \vdots \\ W_{N1} & W_{N2} \end{bmatrix} \begin{bmatrix} f^2 \\ \frac{f^2}{T_Z^2} \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_N \end{bmatrix} , \qquad (10)$$

where W_{i1} , W_{i2} and B_i are computed using (9) with (x, y) replaced by the coordinates (x_i, y_i) of the *i*-th point on the ellipse.

This system can be solved using the least square pseudo-inverse technique:

$$\begin{bmatrix} f^2\\ \frac{f^2}{T_Z^2} \end{bmatrix} = \left(W^t W\right)^{-1} W^t B \ .$$

Solving the system leads to f and T_Z . The components of R_3 can be computed by replacing f in (4). T_X and T_Y can be recovered from (6).

Following the previous steps we recover the normal to the plane that contains the circles (\mathbf{R}_3) and the position of the origin of the WCS in camera coordinates (\mathbf{T}) (see Fig. 2). Fremont [10] proposed a calibration pattern that uses three orthogonal planes to recover the other two axes $(\mathbf{X}_W \text{ and } \mathbf{Y}_W)$. Instead, we use a single mark on the exterior circle that defines the \mathbf{X}_W direction, an idea that has been used before in marker detection [7].

Given the pixel coordinates of the $X_{\rm W}$ axis mark in the image, we reproject it onto the plane of the concentric circles. That plane is completely defined by its normal (\mathbf{R}_3) and a point on it (\mathbf{T}) . Let the X axis mark position be $(r_3, 0, 0)$ in WCS, $(x_{\rm m}, y_{\rm m})$ in image coordinates, and $\mathbf{X}_{\rm m}$ in camera coordinates. Then $\mathbf{X}_{\rm m} = \mu [x_{\rm m} y_{\rm m} f]^t$ where

$$\mu = \frac{-D}{R_{13}x_{\rm m} + R_{23}y_{\rm m} + R_{33}f} \ ,$$

and $D = -\mathbf{R_3}^t \mathbf{T}$. Having the 3D coordinates of the \mathbf{X}_W axis mark given in camera coordinates, and the 3D coordinates of the origin of the WCS, given in camera coordinates as well, the \mathbf{X}_W axis (or $\mathbf{R_1}$) is defined by $\mathbf{X}_W = N \{\mathbf{X}_{\mathrm{III}} - \mathbf{T}\}$ where N is a normalization operator. Obviously, in a right-handed coordinate system, $\mathbf{Y}_W = \mathbf{Z}_W \times \mathbf{X}_W$, or $\mathbf{R_2} = \mathbf{R_3} \times \mathbf{R_1}$.

3.5 Recovering the Principal Point

So far we have assumed that the optical axis of the camera is perfectly centered at the image (i.e., the principal point is the center of the image). In this section we remove this assumption and compute the principal point using the results of the previous sections.

Due to the error in the estimation of the principal point, reprojecting the original circle using the parameters computed in the previous sections does not produce the ellipses in the image. This misalignment is proportional to the error incurred in the estimation of the position of the principal point. By minizating that error, the principal point can be recovered. When processing a video stream with multiple frames, the principal point can be recovered once and kept fixed for the remaining frames. This is true as long as the internal camera settings are not changed.

Once the parameters that define the projection have been recovered, we can reproject the circle of radius r_2 onto an ellipse in the image. By minimizing the error in the reprojection, a good approximation to the principal point can be computed. We have found that the error of reprojection can be defined as the distance between the center of the ellipse used for the calibration and the center of the reprojected ellipse. Alternatively, we can define the error in terms of the angle between the principal axes of those two ellipses. The algorithm would be:

- 1. Start with an initial guess of the principal point (i.e., the center of the image).
- 2. Define the ellipses and the X axis marker of the image with respect to that principal point.
- 3. Calibrate the camera.
- 4. Reproject the original circle (of radius r_2) using the parameters obtained in the previous step.
- 5. Compute the reprojection error and update the working principal point accordingly.

Optimization methods like Levenberg-Marquardt [11] (implemented in MIN-PACK) can efficiently find the 2D position of the principal point that minimizes the error of reprojection.

4 Validating our Method

We have validated our method using both synthetic and real data. We use synthetic data to determine how robust is our method in the presence of noise.



Fig. 3. Relative errors in the estimations of T and R_3 (Z_W)

4.1 Robustness

To check the robustness of the algorithm, we project two concentric circles using a known synthetic camera configuration. Then, we perturb the points of the projected circles by adding random noise to their coordinates. We fit an ellipse to each set of perturbed points using Fitzgibbon's algorithm [5]. Finally, we compute the camera parameters using these two ellipses. Figure 3 shows the errors that the added noise produces in the recovered normal of the supporting plane (\mathbf{R}_3) and the translation vector (\mathbf{T}) . Note that the error incurred is relatively small.

We have found that the system is very robust in the presence of systematic errors, i.e., when both ellipses are affected by the same error (for instance, with a non-centered optical axis). On the other hand, if the parameters of the ellipses are perturbed beyond a certain limit, the accuracy of the results decreases dramatically.

4.2 Experimental Results

In order to validate the computed calibration with real images, we have applied our algorithm to several images taken with a camera. Figure 4 shows an example of the process. First, the ellipses were recovered from the image and the camera parameters were computed. By using those parameters, we can draw the WCS axes on the image. Furthermore, the marker has been reprojected using the same parameters. The marker seen in the image has the following properties $r_1 = 2.6$ cm, $r_2 = 5$ cm and $r_3 = 6.5$ cm.

5 Conclusions and Future Work

In this paper we introduce a camera calibration technique that uses a very simple pattern made of two circles. The algorithm obtains accurate intrinsic and extrinsic camera parameters. We show that our method behaves in a robust manner in the presence of different types of input errors. We also show that the algorithm



Fig. 4. Reprojecting the marker and the coordinate system in the images

works well with real world images as long as good ellipse extraction and fitting algorithms are used.

Our work has a lot of applications, particularly in camera tracking and related fields. Our marker is easy to build and use. This makes it particularly well suited for augmented reality and entertainment applications. We are currently working on applications in these two areas. We are also trying to extend our camera model to take into account skew and lense distortion, in order to better approximate the behavior of a real camera. We are exploring the working limits of our algorithm and we are studying techniques to make the results more stable in the presence of noise.

References

- Zhang, Z.: A Flexible New Technique for Camera Calibration. IEEE Trans. Patt. Anal. Machine Intell., vol. 22, no. 11, (2000) 1330–1334
- Forsyth, D., Mundy, et al.: Invariant Descriptors for 3-D Object Recognition and Pose. IEEE Trans. Patt. Anal. Machine Intell., vol. 13, no. 10, (1991) 971–991
- Kanatani, K., Liu, W.: 3D Interpretation of Conics and Orthogonality. CVGIP: Image Undestanding, Vol. 58, no. 3, (1993) 286–301
- Rothwell, C.A., Zisserman, A., et al.: Relative Motion and Pose from Arbitrary Plane Curves. Image and Vision Computing, vol. 10, no. 4, May (1992) 250–262
- Fitzgibbon, A.W., Pilu, M., Fisher, R.B.: Direct Least Squares Fitting of Ellipses. IEEE Trans. Patt. Anal. Machine Intell., vol. 21, no. 5, (1999) 476–480
- Ahn, S.J., Rauh, W., Kim, S.I.: Circular Coded Target for Automation of Optical 3D-Measurement and Camera Calibration. Int. Jour. Patt. Recog. Artificial Intell., vol. 15, no. 6, (2001) 905-919
- López de Ipiña, D., Mendonça, P.R.S., Hopper, A.: TRIP: a Low-Cost Vision-Based Location System for Ubiquitous Computing. Personal and Ubiquitous Computing Journal, Springer, Vol. 6, no. 3, (May 2002) 206–219
- Kim, J.S., Kweon, I.S.: A New Camera Calibration Method for Robotic Applications. Int. Conf. Intelligent Robots and Systems, Hawaii, (Oct 2001) 778–783
- Kim, J.S., Kim, H.W., Kweon, I.S.: A Camera Calibration Method using Concentric Circles for Vision Applications. 5th Asian Conf. Computer Vision (2002)
- Fremont, V., Chellali, R.: Direct Camera Calibration using Two Concentric Circles from a Single View. 12th Int. Conf. Artificial Reality and Telexistence (2002)
- 11. More, J.J.: The Levenberg Marquardt algorithm: implementation and theory. Numerical Analysis, G.A. Watson ed., Springer-Verlag (1977)

Appendix

Given two concentric circles whose projections are known, we show how to project a third circle of known radius using the same projection. A circle C of radius r centered at the origin and located in the plane Z = 0 is defined by:

$$X^{t}CX = \begin{bmatrix} X \ Y \ 1 \end{bmatrix} \begin{bmatrix} 1 \ 0 \ 0 \\ 0 \ 1 \ 0 \\ 0 \ 0 \ -r^{2} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

A projection matrix P projects a circle C onto an ellipse Q by $\lambda Q = P^{-t}CP^{-1}$. We compute the difference between the projection of a circle of radius $r + \alpha$ and the projection of a circle of radius r:

$$\lambda_1 Q_{r+\alpha} - \lambda_2 Q_r = P^{-t} (C_{r+\alpha} - C_r) P^{-1} = \alpha (\alpha + 2r) M \quad , \tag{11}$$

where $M = q^t q$ and q is the third row of matrix P^{-1} . Therefore, we can write:

$$\begin{cases} Q_3 = Q_1 - \alpha_1(\alpha_1 + 2r_1)M \\ Q_3 = Q_2 - \alpha_2(\alpha_2 + 2r_2)M \end{cases},$$

where $\alpha_1 = r_3 - r_1$ and $\alpha_2 = r_3 - r_2$. Using these two equations we can express Q_3 as:

$$Q_3 = kQ_2 - Q_1 \frac{r_3^2 - r_2^2}{r_3^2 - r_1^2} , \qquad (12)$$

where k is a scale correcting factor of Q_1 and Q_2 . That factor can be computed applying the rank 1 condition to the ellipses (note that as $C_{r+\alpha} - C_r$ in equation (11) has rank 1, $\lambda_1 Q_{r+\alpha} - \lambda_2 Q_r$ should have rank 1, too) [9]. Solving for k in $Q_1 - kQ_2$ to have a rank 1 matrix leads to a scale correcting factor. Therefore, equation (12) allows us to project a circle of any radius given the projection of two circles, all of them concentric.

This process can be used to find the projected center of the concentric circles as well. As the projected center of a circle is always enclosed in its projected ellipse, if we project circles of smaller and smaller radii, we will be reducing the space where the projected center can be. In the limit, a circle of radius zero should project onto the projected center of the circles. Applying equation (12) to a circle of radius $r_3 = 0$ results in an ellipse (of radius zero) whose center is at the projected center of the concentric circles $(x_{\rm C}, y_{\rm C})$. The center of an ellipse in matrix form is given by [8]:

$$x_{\rm C} = \frac{Q_{(2,2)}Q_{(1,3)} - Q_{(1,2)}Q_{(2,3)}}{\left(Q_{(1,2)}\right)^2 - Q_{(1,1)}Q_{(2,2)}} \quad \text{and} \quad y_{\rm C} = \frac{Q_{(2,3)}Q_{(1,1)} - Q_{(1,2)}Q_{(1,3)}}{\left(Q_{(1,2)}\right)^2 - Q_{(1,1)}Q_{(2,2)}}$$