Planar pattern for automatic camera calibration

Beiwei Zhang
Y. F. Li
City University of Hong Kong
Department of Manufacturing Engineering
and Engineering Management
Kowloon, Hong Kong

Fu-Chao Wu
Institute of Automation
Chinese Academy of Science
National Laboratory of Pattern Recognition
Beijing, China

Abstract. We present a method for automatic camera calibration that requires only that a planar pattern be visible by the camera from a few (at least three) different views. A major issue is how to obtain the projection of an absolute point using the planar pattern. In the calibration, we can move either the camera or the planar pattern and no knowledge about the motion is required. We consider and resolve both linear and nonlinear parameters. In testing the proposed methods, satisfactory results are achieved in the simulation and in real data experiments. © *2003 Society of Photo-Optical Instrumentation Engineers.* [DOI: 10.1117/1.1574037]

Subject terms: planar pattern; automatic calibration; plane at infinity; absolute point; vanishing point.

Paper 010437 received Nov. 29, 2001; revised manuscript received Apr. 22, 2002; accepted for publication Jun. 24, 2002.

1 Introduction

Camera calibration is the process of determining the intrinsic and extrinsic parameters of a camera and much work has been conducted in this area.[1–7] The work has included photogeometric calibration and self-calibration.[3–10] Zeller et al. described a calibration technique based on the Kruppa equation using image sequences.[3] Ma explored a method for camera calibration using two sequences of motion, each consisting of three orthogonal translations.[4] Zhang proposed a technique that requires the camera to view a planar pattern shown at a few (at least two) different poses.[7] He made use of the relationship between a 3-D point and its image projection to calibrate the camera. However, most of the existing methods require either special motion of the camera or measurement of the patterns on the calibration target, which causes difficulties in practical use.

To overcome these limitations, Li and Chen[9] and Chen and Li[10] investigated the use of a single view for automatic recalibration of active vision systems in their recent study. In this paper, we study the use of a planar pattern for automatic calibration of the intrinsic and extrinsic camera parameters. Our method does not require the pattern to be placed in known positions. The planar pattern contains only an equilateral dodecagon whose position does not need to be known. Therefore it is easy to make the planar pattern. Furthermore, no knowledge is required about the motion of the calibration target. In this paper, a linear method is used to calibrate the linear parameters. Nonlinear lens distortion is also considered. A nonlinear refinement method is used to obtain the nonlinear parameters. Both computer simulation and real data experiments were conducted to verify the proposed method and satisfactory results were achieved. The theoretical analysis and experimental results show that this method has high potential for applications in robot vision.

This paper is organized as follows. Section 2 introduces the concept of the absolute point. In Sec. 3, we give a proof that the projection of an absolute conic is still an absolute conic. Section 4 then presents the procedure for computing the projection of an absolute point from the image of the pattern. Nonlinear lens distortion is also considered in the study. Both computer simulation and real data experiments were conducted to verify the proposed methods in Sec. 5 and good results were achieved. Section 6 provides a summary of the study conducted.

2 Absolute Point in 3-D Space

2.1 *Notation*

For a 3-D point denoted by $\mathbf{x}=(x,y,z)^T \in R^3$, its homogenous coordinate is denoted by adding a scale as the last element, i.e., $\tilde{\mathbf{x}}=(tx,ty,tz,t)^T$ (where $t\neq 0$). A point at infinity is denoted by $\tilde{\mathbf{x}}_\infty=(x,y,z,0)^T$, where x, y, and z are not all zeros. All such points make a plane referred to as the plane at infinity, which is denoted by

$$\pi_\infty=\{\tilde{\mathbf{x}}_\infty=(x,y,z,0)^T | x\neq 0 \cup y\neq 0 \cup z\neq 0\}.$$

In 3-D projective space $\tilde{R}^3$, a plane can be denoted by $ax+by+cz+dw=0$ or $\tilde{\mathbf{n}}^T\tilde{\mathbf{x}}=0$, where $\tilde{\mathbf{n}}=(a,b,c,d)^T$. When $a=b=c=0$ and $d\neq 0$, the plane becomes a plane at infinity. An absolute conic[11] on a plane at infinity can be denoted by $\tilde{\omega}_\infty=\{\tilde{\mathbf{x}}_\infty | \tilde{\mathbf{x}}_\infty^T\tilde{\mathbf{x}}_\infty=0\}$. This indicates that the equation for $\tilde{\omega}_\infty$ is $\tilde{\mathbf{x}}_\infty^T\tilde{\mathbf{x}}_\infty=0$.

2.2 *Definition of Absolute Point*

Assume a spatial plane denoted by π. There is an orthogonal coordinate system $O-xyz$ in the space whose x and y axes lie on the plane while the z axis is perpendicular to this plane. Then the equation of the plane π is $z=0$. The line at infinity on the plane is the intersection of π with the plane at infinity, i.e.,

$$l_\infty=\{\tilde{x}_\infty=(x,y,0,0)^T | x\neq 0 \cup y\neq 0\}. \quad (1)$$

Let c be a circle in the plane π with $\mathbf{o}(x_0,y_0,0,1)$ as its center and let r be its radius. The equation of c is

 0091-3286/2003/$15.00 © 2003 Society of Photo-Optical Instrumentation Engineers

$$c=\{\tilde{x}=(x,y,0,w)^T|(x-x_0w)^T+(y-y_0w)^T-w^2r^2=0\}, \tag{2}$$

where (x,y,z,w) is the homogenous coordinate of a point on the circle. The intersection of the line at infinity with the circle is called an absolute point.

From Eqs. (1) and (2), we can have the following solutions for the coordinates of the absolute points:

$$\mathbf{I}=(1,i,0,0)^T,\quad \mathbf{J}=(1,-i,0,0)^T.$$

Obviously, the absolute points are a pair of conjugate points and are not related to the center or radius of the circle. It can also be inferred that

$$\mathbf{I},\mathbf{J}\in\omega_\infty. \tag{3}$$

2.3 *Constraints on the Intrinsic Parameters*

In this paper, a camera is modeled as a normal pinhole camera. The relationship between a 3-D point $\mathbf{x}=(x,y,z,w)^T$ and its image projection $\mathbf{m}=(u,v,s)^T$ is given by

$$\lambda\mathbf{m}=\mathbf{K}[\mathbf{R},\mathbf{t}]\mathbf{x},$$

where

$$\mathbf{K}=\begin{bmatrix} f_u & s & u_0 \\ 0 & f_\nu & \nu_0 \\ 0 & 0 & 1 \end{bmatrix}$$

is the matrix of intrinsic parameters of the camera, and $(\mathbf{R},\mathbf{t})$ represents the transformation by the camera.

Let $\mathbf{x}\in\tilde{\omega}_\infty$, i.e., $\mathbf{x}^T\mathbf{x}=0$. Then we have

$$\begin{aligned}\mathbf{m}^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{m}&=\lambda^{-2}\mathbf{x}^T[\mathbf{R},\mathbf{t}]^T\mathbf{K}^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{K}[\mathbf{R},\mathbf{t}]\mathbf{x}\\&=\lambda^{-2}\mathbf{x}^T[\mathbf{R},\mathbf{t}]^T[\mathbf{R},\mathbf{t}]\mathbf{x}=\lambda^{-2}\mathbf{x}^T\mathbf{x}=0,\end{aligned}$$

where $\mathbf{m}$ is the projection of $\mathbf{x}$. This proves that the image of an absolute conic is also an absolute conic, and its equation is

$$\mathbf{m}^T\mathbf{K}^{-T}\mathbf{K}^{-1}\mathbf{m}=0. \tag{4}$$

From Eq. (3), we know that the projections of absolute points satisfy Eq. (4), i.e.,

$$\mathbf{m}_I^T\mathbf{C}\mathbf{m}_I=0,\quad \mathbf{m}_J^T\mathbf{C}\mathbf{m}_J=0, \tag{5}$$

where $\mathbf{C}=\mathbf{K}^{-T}\mathbf{K}^{-1}$, and $\mathbf{m}_I$ and $\mathbf{m}_J$ are the projections of I and J.

Under projective transformation, $\mathbf{m}_I$ and $\mathbf{m}_J$ remain a pair of conjugate points. Thus Eq. (5) places only the following two constraints on the intrinsic parameters:

$$\mathrm{Re}(\mathbf{m}_I^T\mathbf{C}\mathbf{m}_I)=0,\quad \mathrm{Im}(\mathbf{m}_I^T\mathbf{C}\mathbf{m}_I)=0, \tag{6}$$

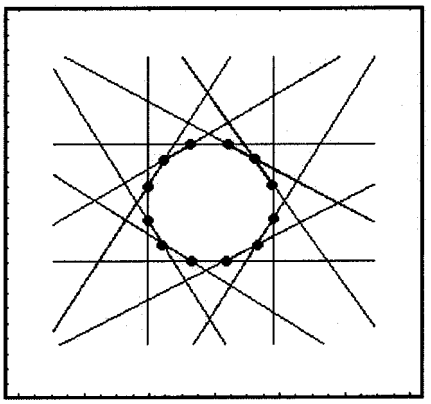


Fig. 1 Planar pattern.

where Re and Im denote the real and imaginary parts respectively. Because our camera model has five different intrinsic parameters, we require at least five equations to solve for them.

3 Implementation Algorithm Using the Planar Pattern

3.1 *Design of the Planar Pattern*

From Eq. (6), we know that the key step for camera calibration is the projection of the absolute points. How to design a planar pattern and achieve the projections of the absolute points from the pattern conveniently is of critical importance here. For this purpose, we designed a kind of planar pattern using equilateral polygons with $2n$ sides ($n>2$). In general, the larger the n, the more accurately this polygon can approximate a circle. However, a large n increases the computational load in the approximation. From our experiments, we found that an acceptable compromise is achieved when $n=6$. This results in a planar pattern of an equilateral dodecagon, as shown in Fig. 1. We extend each side of the dodecagon for the ease of subsequent uses.

3.2 *Solving the Projection of Absolute Points*

3.2.1 *Solving the vanishing line*

A vanishing point is the projection of a point at infinity. A vanishing line is the projection of a line at infinity. There are six groups of parallel lines in the equilateral dodecagon and they intersect at six points on the line at infinity. According to the principle of projection, their projections also intersect at six points on the vanishing line. Thus we can solve for the vanishing line with the projection of the equilateral dodecagon (see Fig. 3 in Sec. 3.2.2). The procedure is given as follows:

1. For a pair of parallel lines AB and HG, obtain their projections: $l_{ab}=0$ and $l_{hg}=0$.
2. Solve the two equations $l_{ab}=0$ and $l_{hg}=0$, and obtain their intersection point p_1, which is a vanishing point.

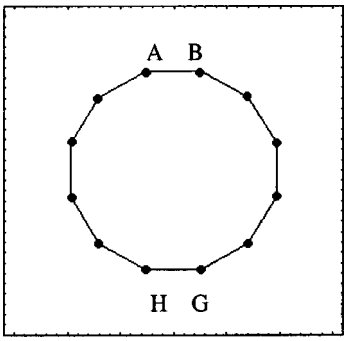


Fig. 2 Equilateral dodecagon in the planar pattern.

3. Repeat steps 1 and 2. We can then obtain the other five vanishing points, giving p_i, $i=1,2,\ldots,6$.
4. Solve the following equations (with a, b, and c as the unknowns) using $p_i=(u_i,v_i,w_i)^T$:
   $u_i a+v_i b+w_i c=0 \quad i=1,2,\ldots,6.$

We can then obtain the equation of the vanishing line $l_s: au+bv+cw=0$.

3.2.2 *Solution of the projection of a circle*

The equilateral dodecagon has a circumcircle whose projection is an ellipse. According to the principle of projection, the projections of the vertices of the equilateral dodecagon are all on the ellipse. Thus we can solve the projection of the circle (see Figs. 2 and 3) by the following procedure: First, let the equation of the ellipse be

$$E(u,v,w)=(u,v,w)\begin{bmatrix} a & d & e \\ d & b & f \\ e & f & c \end{bmatrix}\begin{pmatrix} u \\ v \\ w \end{pmatrix}=0,$$

where a, b, c, d, e, and f are the unknowns.

Second, solve the following equations with $m_i=(u_i,v_i,w_i)^T$:

$$\mathrm{A}x=0,$$

where

$$\mathrm{A}=\begin{bmatrix} u_1^2 & v_1^2 & w_1^2 & 2u_1v_1 & 2u_1w_1 & 2v_1w_1 \\ u_2^2 & v_2^2 & w_2^2 & 2u_2v_2 & 2u_2w_2 & 2v_2w_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_{12}^2 & v_{12}^2 & w_{12}^2 & 2u_{12}v_{12} & 2u_{12}w_{12} & 2v_{12}w_{12} \end{bmatrix},$$

$x=(a\ \ b\ \ c\ \ d\ \ e\ \ f)^T$.

We can get the equation of the ellipse: $E_c=0$.

3.2.3 *Solution of the projection of an absolute point*

An absolute point is the intersection of a line at infinity with a circle. According to the properties of projection transformation, its projection is the intersection of the vanishing line with the ellipse. So we can solve this problem with the equations of the vanishing line and the ellipse. That is solving the group of equations: $l_s=0$ and $E_c=0$.

3.3 *Solution of the Linear Parameters of the Camera*

Based on the preceding elaboration, we can have the following algorithm for the automatic camera calibration:

1. Move either the planar pattern or the camera to take a few (at least three) images of the planar pattern in different poses.
2. Calculate the absolute point for each image.
3. Solve the group of constraints derived from Eq. (6) and obtain the solution $\mathbf{C}^*$.
4. Derive the intrinsic parameter matrix $\mathbf{K}$ of the camera from $\mathbf{C}^*$.

4 Solution for Nonlinear Camera Parameters

4.1 *Nonlinear Camera Parameters*

A real camera does not follow the pinhole model strictly because of many factors. The camera lens is always subject to some distortions. The parameters describing the distortions are nonlinear and can be described by the following equations[12,13]:

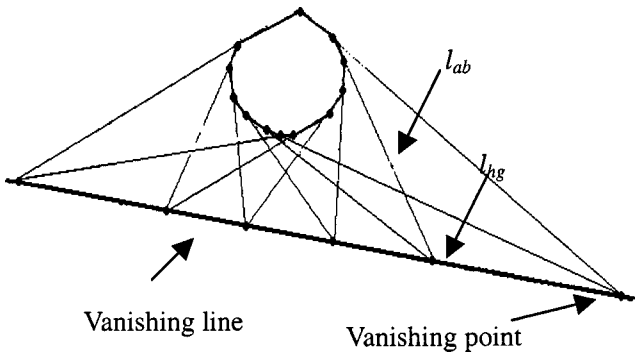


Fig. 3 Vanishing line.

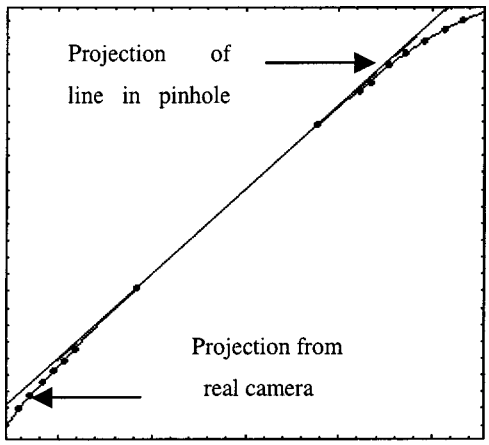
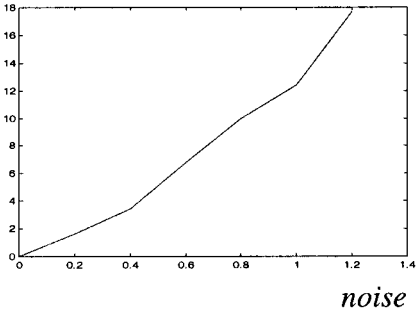


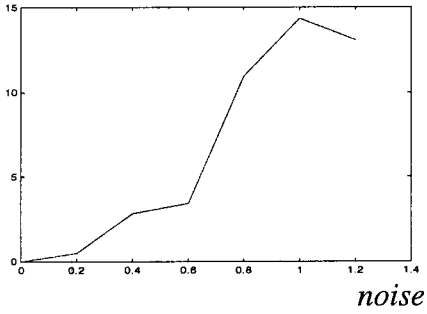
Fig. 4 Comparison of two kinds of projections.

$\nabla \mid fu \mid$



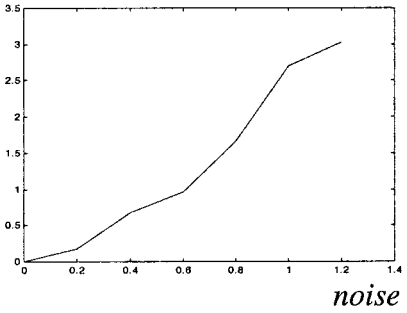
(a) Average error of f_u

$\nabla \mid fv \mid$



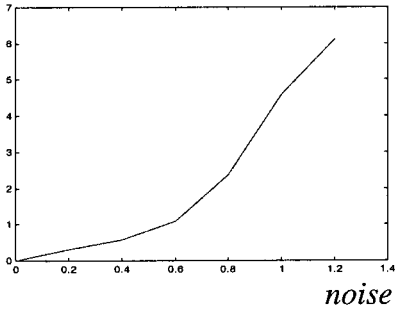
(b) Average error of f_v

$\nabla \mid u0 \mid$



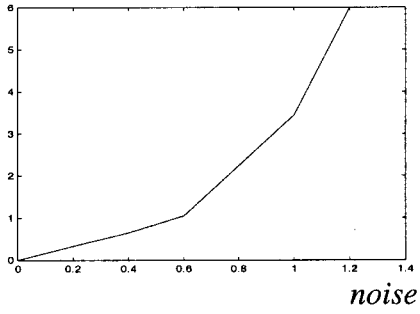
(c) Average error of u_0

$\nabla \mid v0 \mid$



(d) Average error of v_0

$\nabla \mid s \mid$



(e) Average error of s

Fig. 5 Experimental results of linear parameters.

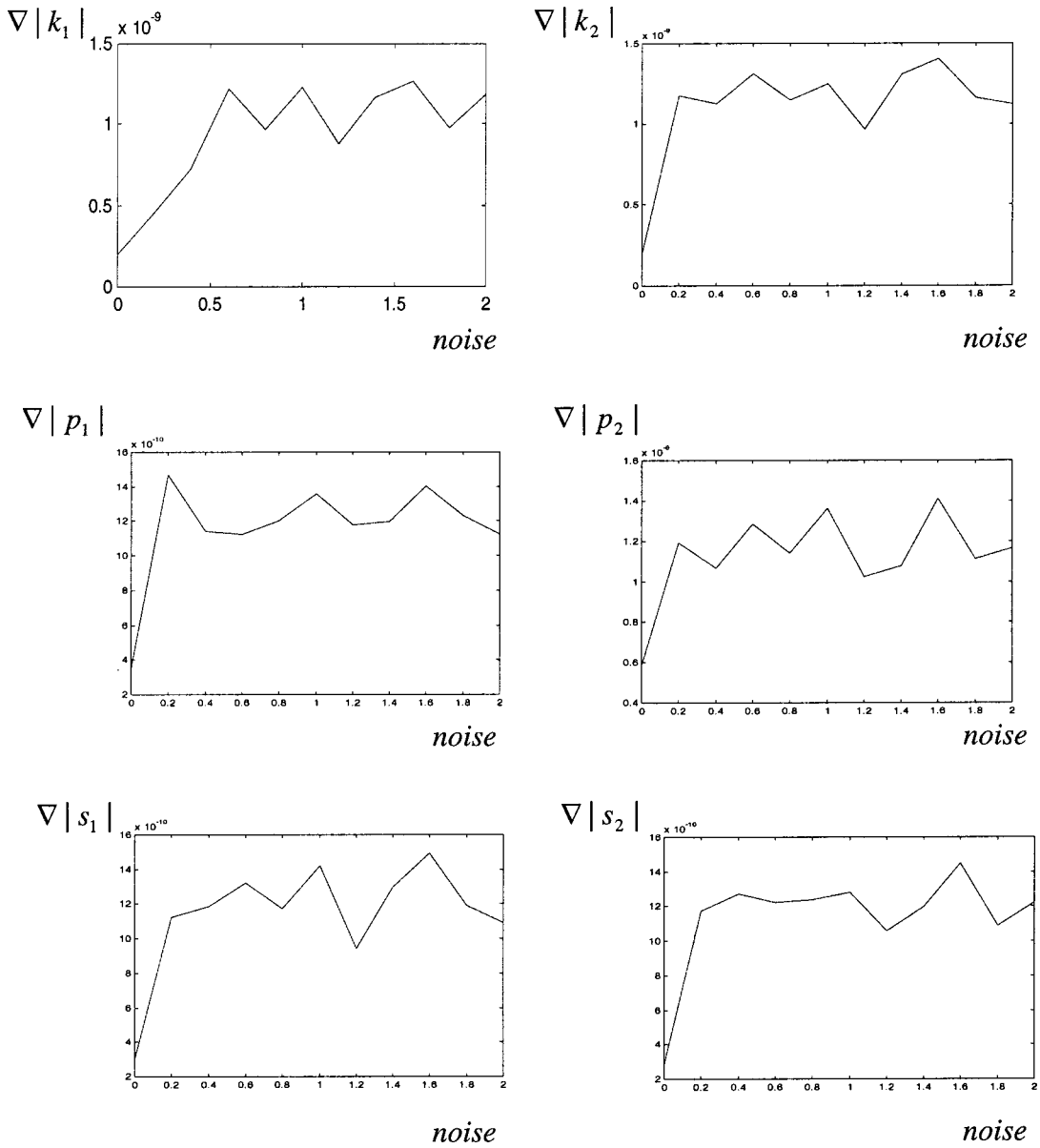


Fig. 6 Experimental results of nonlinear parameters.

$$\begin{cases} \overline{x} = x + \delta_x(x,y) \\ \overline{y} = y + \delta_y(x,y) \end{cases}$$

where $(\overline{x},\overline{y})$ is the projection by the linear camera model, and (x,y) is the projection in real camera. Here (δ_x, δ_y) represents the nonlinear distortion and can be given as follows[11,14]:

$$\begin{cases} \delta_x = k_1 x(x^2+y^2) + [p_1(3x^2+y^2) + 2p_2xy] + s_1(x^2+y^2), \\ \delta_x = k_2 y(x^2+y^2) + [p_2(3x^2+y^2) + 2p_1xy] + s_2(x^2+y^2), \end{cases}$$

where the first term is the radial distortion, the second represents the decentering effect, and the third represents thin prism effect. Here k_1, k_2, p_1, p_2, s_1, and s_2 are the nonlinear camera parameters.

The characteristics of distortions are such that their effects can be ignored in the center area and the degree of distortion increases toward the edge, as shown in Fig. 4.

4.2 *Solution of the Nonlinear Parameters*

With the knowledge of the characteristics of the distortions and those of the projective transformation, we can solve for the nonlinear parameters via the following steps:

1. Choose a line **L** in the given planar pattern and obtain its projection **S**. Select several pixels in its center area and derive a line **l** by a nonlinear optimization technique.
2. Select several pixels in another area **S**. The projections of these pixels $P_i^S = (x_i^s, y_i^s)^T$ must be on **l** by the pinhole model. Thus, we can get the following

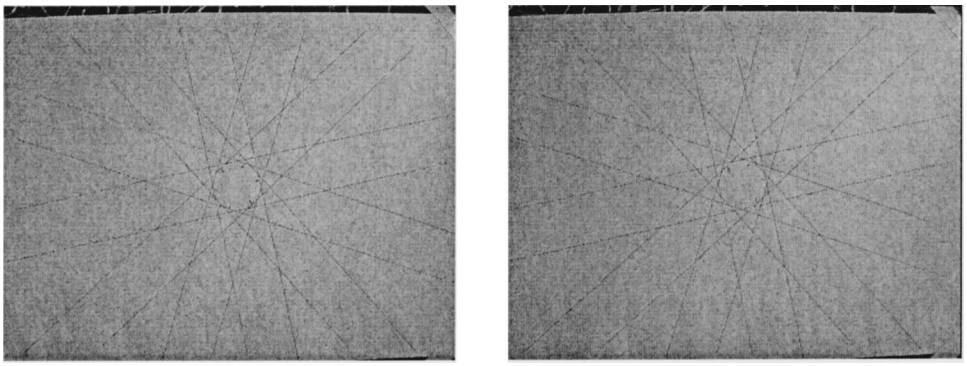


Fig. 7 Two pictures taken in the experiments.

equation for the nonlinear parameters k_1, k_2, p_1, p_2, s_1, and s_2:

$$f_i(k_1,k_2,p_1,p_2,s_1,s_2)=a(x_i^s+\delta_x(x_i^s,y_i^s)+b[y_i^s+\delta_x(x_i^s,y_i^s)]+c=0, \quad i=1,2,\ldots,n_s,$$

where

$$\begin{cases}\delta_x(x,y)=k_1x(x^2+y^2)+[p_1(3x^2+y^2)+2p_2xy]+s_1(x^2+y^2)\\ \delta_x(x,y)=k_2y(x^2+y^2)+[p_2(3x^2+y^2)+2p_1xy]^{\cdot}+s_2(x^2+y^2)\end{cases}$$

3. Repeating the preceding two steps, we can obtain a group of equations as follows:
   $F_{ij}(k_1,k_2,p_1,p_2,s_1,s_2)=0, \quad j=1,2,\ldots,1, 2,$
   $i=1,2,\ldots,n_j,$
   where j denotes the sequence of a line, and i denotes sequence of a point.
4. Solving the preceding equations, we can obtain k_1, k_2, p_1, p_2, s_1, and s_2.

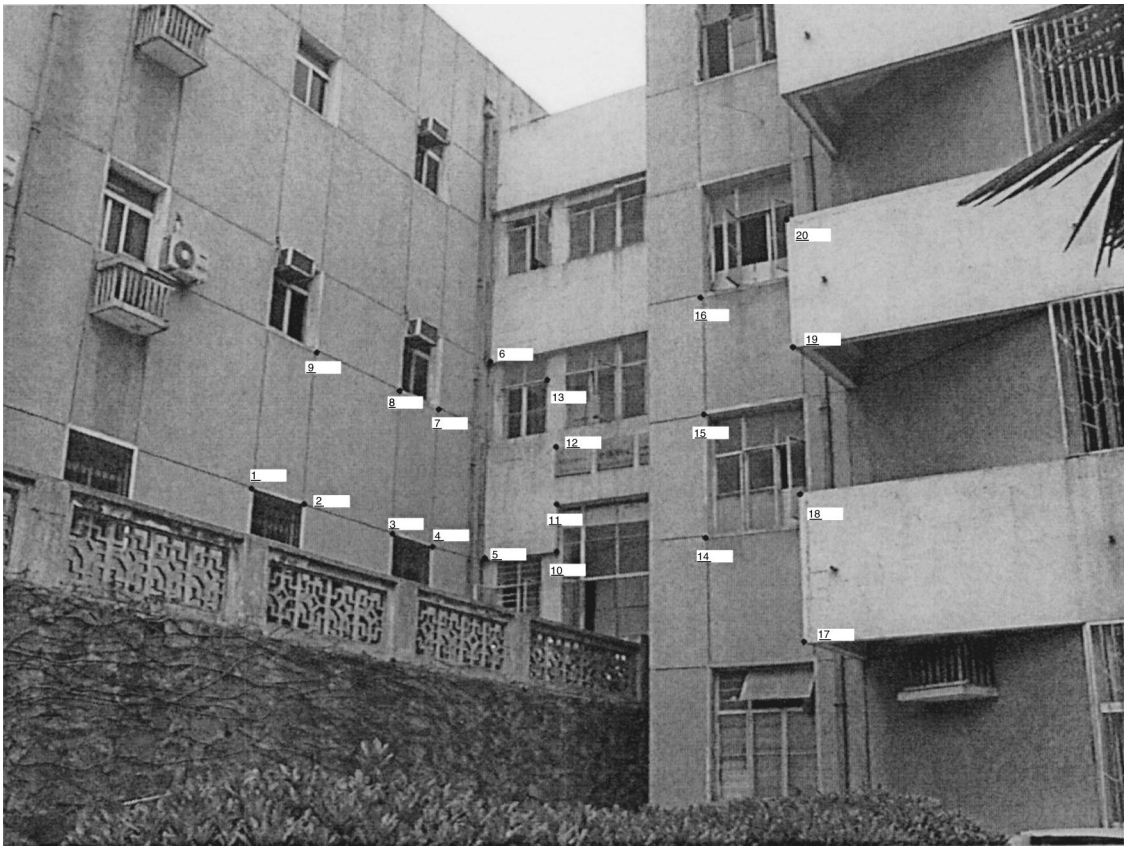


Fig. 8 Twenty points for reconstruction.

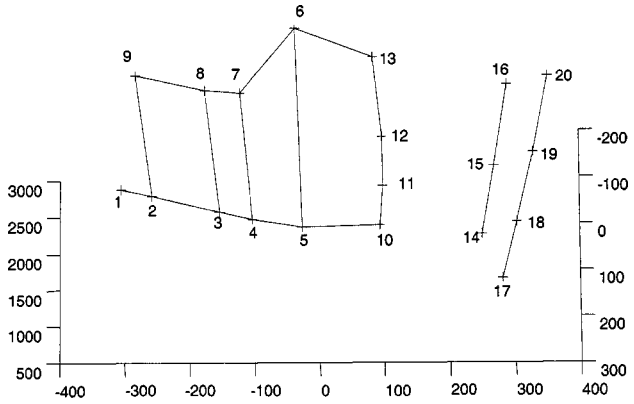


Fig. 9 Result of 3-D reconstruction.

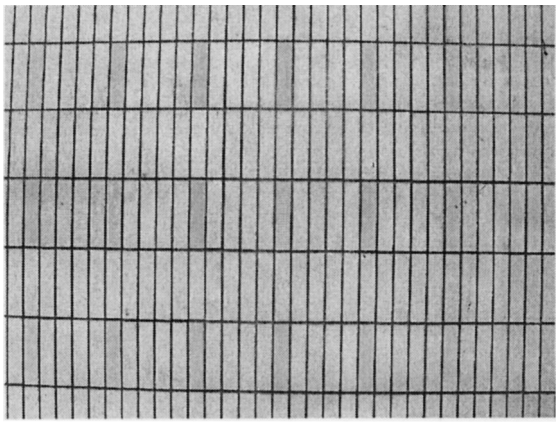


Fig. 10 Picture of the wall of a building.

5 Implementation Results

5.1 *Computer Simulations*

5.1.1 *Linear parameters*

The simulated camera has the following parameters: $f_u = 990$, $f_v = 990$, $s = 2$, $u_0 = 10$, and $v_0 = 10$. The image size used is 1280×960. Different levels of random noise with zero mean, i.e., $N(0,\sigma^2)$ are added to the projected image points. We vary the noise levels from 0.0 to 1.2 pixels. For each noise level, we performed 100 independent trials, and the results of the average errors are shown in Fig. 5.

5.1.2 *Nonlinear parameters*

The simulated camera has the following nonlinear parameters: $k_1 = k_2 = p_1 = p_2 = s_1 = s_2 = 1.0 \times 10^{-10}$. The results of the average errors are shown in Fig. 6. From these results, we can see that when the noise level is very low, say less than 0.3, and the average errors are within a reasonable range. The errors of the linear parameters tend to increase along with the noise levels, whereas those of the nonlinear parameters remain more or less stabilized as the noise level is further increased.

5.2 *Experiments with Real Data*

5.2.1 *Solution of the linear parameters*

Calibration of the linear parameters. In the experiment, we first took five pictures (two of which are shown in Fig. 7) of the planar pattern with our camera from different views. Applying the algorithm from Sec. 3.3, we obtained the following results:

$$\mathrm{K} = \begin{bmatrix} 1369.7 & -3.3 & -38.6 \\ 0 & 1376.8 & -16.9 \\ 0 & 0 & 1 \end{bmatrix}.$$

Verification of the results. We used the technique of 3-D reconstruction to verify the linear parameters in matrix **K**, which we obtained by the preceding calibration method. We selected 20 points in the real scene shown in Fig. 8 and reconstructed these points in 3-D space with the results shown in Fig. 9.

From the preceding experimental results, we observe that the reconstructed 3-D points match the real 3-D points of the building well, with such small matching errors that the differences are nearly invisible. This proves that our algorithm for the linear parameter calibration is valid in practice.

5.2.2 *Solution of the nonlinear parameters*

Calibration of the nonlinear parameters. At this stage, we selected one of the five pictures of the planar pattern and implemented the algorithm from Sec. 4.2. The following results were then obtained:

$$\mathrm{k}_1 = 2.63298182907667 \times 10^{-008},$$

$$\mathrm{k}_2 = -7.54119342616964 \times 10^{-010},$$

$$\mathrm{p}_1 = -7.46723672408062 \times 10^{-006},$$

$$\mathrm{p}_2 = -8.11026915720216 \times 10^{-007},$$

$$\mathrm{s}_1 = -7.37428432755732 \times 10^{-006},$$

$$\mathrm{s}_2 = 7.04970377098087 \times 10^{-006}.$$

Verification of the results. Next we conducted some other experiments to verify the results we had just obtained. Figure 10 is a picture taken from the wall of a building. Figure 11 is the rectified result using the obtained nonlinear parameters. Comparing these two figures, we can see that the nonlinear distortions, which are easily observable in the original image in Fig. 10 (especially the lines close to the boundaries), were corrected using the parameters obtained by our calibration method. The result in the removal of the distortions, as seen in Fig. 11, proves that our algorithm for the nonlinear parameters is a valid solution.

6 Conclusions

We presented a new planar pattern for automatic camera calibration. The method requires only that the camera view the pattern from a few (at least three) different poses. We can move either the camera or the target during the calibration. The motion does not need to be known for the cali-

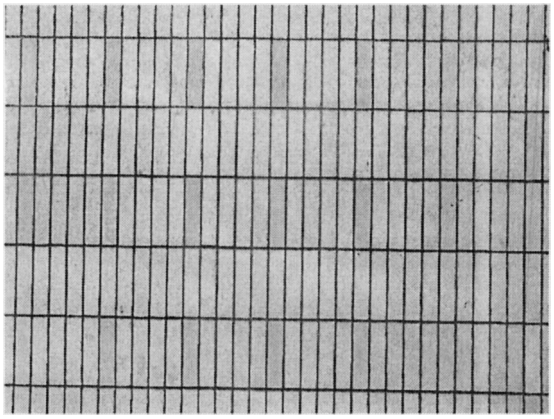


Fig. 11 Rectified picture.

bration. In addition to the normal parameters in a pinhole model, nonlinear lens distortion can also be taken into account by our calibration method. Both computer simulation and experiments on real data were conducted to verify the validity of the proposed method. The satisfactory results obtained demonstrated that our method provides a correct and practical solution for automatic camera calibration. Without the necessity of precise placement of the calibration target, this method offers the advantage of ease of use in practical applications. In a robotic application, this removed the required knowledge of the relative motions between the camera and the target, significantly facilitating on-line calibration of a robot vision system during the execution of a task.

Acknowledgments

The work described in this paper was partly support by a grant from the Research Grants Council of Hong Kong, Project No. CityU 1136/98E (9040368).

References

1. R. Hartley, ''Estimation of relative camera positions for uncalibrated cameras,'' in Proc. ECCV'92, Italy, pp. 579–387 (1992).
2. O. D. Faugeras, ''What can be seen in three dimensions with an uncalibrated stereo rig,'' in Proc. ECCV'92, Italy, pp. 563–578 (1992).
3. C. Zeller and O. D. Faugeras, ''Camera self-calibration from video sequences: the Kruppa equations revisited,'' Research Report 2793, INRIA (1996).
4. S. D. Ma, ''A self-calibration technique for active vision systems,'' *IEEE Trans. Rob. Autom.* **12**(1), 114–120 (1996).
5. F. Du and J. M. Brady, ''Self-calibration of the intrinsic parameters of cameras for active vision system,'' in Proc. CVPR'93, New York, pp. 477–482 (1993).
6. R. Y. Tsai, ''An efficient and accurate camera calibration technique for 3D machine vision,'' in Proc. CVPR'86, Miami (1986).
7. Z. Zhang, ''A flexible new technique for camera calibration,'' MSRTR-98-71, Microsoft Research (Dec. 1998).
8. J. Y. Luh and J. A. Klaasen, ''A three dimensional vision by off-shelf system with multi-cameras,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **7**(1), 35–45 (1985).
9. Y. F. Li and S. Chen, ''Automatic recalibration of an active structured light vision system,'' IEEE Trans. Robot. Automat. (in press).
10. S. Chen and Y. F. Li, ''Self-recalibration of a colour-encoded light system for automated 3-D measurements,'' Meas. Sci. Technol. (in press).
11. J. Weng, P. Cohen, and M. Herniou, ''Camera calibration with distortion models and accuracy evaluation,'' *IEEE Trans. Pattern Anal. Mach. Intell.* **14**(10), 965–980 (1992).
12. S. D. Ma and Z. Zhang, *Computer Vision—Basics of Theory and Algorithm*, Science Publishing Company, Beijing (1998).
13. K. B. Atkison, *Developments in Close Range Photogrammetry*, London (1980).
14. J. Weng, P. Cohen, and M. Herniou, ''Calibration of stereo cameras using a non-linear distortion model,'' in *Proc. Int. Conf. on Pattern Recognition*, Vol. 1, pp. 246–253 (1990).

Beiwei Zhang received his BS degree in mechanical and electrical science from Anhui Agriculture University in 1998 and his MS degree in computer science from Anhui University, China, in 2001. His current research interests include computer vision, image processing, and optical character recognition. He is currently a PhD student in the Department of Manufacturing Engineering Management, City University of Hong Kong.

Y. F. Li received his BS and MS degrees in electrical engineering from Harbin Institute of Technology, China. From 1989 to 1993 he was with the Robotics Research Group, Department of Engineering Science, the University of Oxford, United Kingdom. He received the PhD degree in engineering science from the University of Oxford in 1993. From 1993 to 1995 he was a postdoctoral research associate in the AI and Robotics Research Group, the Department of Computer Science, the University of Wales, Aberystwyth, United Kingdom. He is currently an associate professor in the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong. His research interests include robot vision and sensor-based control for robotics. He is a senior member of IEEE.

Fu-Chao Wu: Biography not available.