Journal of
Electronic Imaging

SPIEDigitalLibrary.org/jei

Direct fitting of center collinear circles for equidistant fish-eye camera calibration

Feng Yue
Bin Li
Ming Yu

SPIE

IS&T
imaging.org

Direct fitting of center collinear circles for equidistant fish-eye camera calibration

Feng Yue
Bin Li
Beijing Academy of Science and Technology
Beijing Institute of New Technology Applications
Beijing, China
E-mail: csfyue@gmail.com

Ming Yu
Hebei University of Technology
Department of Computer Science and Engineering
Tianjin, China

Abstract. *An equidistant fish-eye camera can be calibrated using a single image of two sets of parallel lines in the scene, which is very convenient for practical applications. The core of the calibration procedure is the fitting of center collinear circles that intersect at two vanishing points. A fast and accurate fitting method that can be used for equidistant fish-eye camera calibration is presented. First, the fitting problem is formulated as a nonlinear least square problem and solved using Levenberg-Marquardt optimization. The objective function is also derived, as well as the Jacobian required by the Levenberg-Marquardt algorithm. It is also demonstrated how to determine the initial estimations. Experimental results on synthetic data have demonstrated the superiority of our method to two existing approaches in terms of speed and accuracy. Results on real data have also demonstrated its effectiveness for fish-eye camera calibration.* *[DOI: 10.1117/1.JEI.22.1.013004]*

1 Introduction

Because of its large field of view (FOV), the fish-eye camera has been widely used in applications such as robot navigation, visual surveillance and three-dimensional (3-D) reconstruction.[1] However, fish-eye lenses often introduce a large amount of radial distortion, which causes straight lines in the scene mapping to conics in the image. As a result, a calibration procedure is often required for distortion correction.[2]

The calibration procedure typically consists of two components, model selection and parameter estimation. In the last decades a number of models have been developed for fish-eye lenses, such as the polynomial model,[3] the division model,[4] the rational function model,[5] and so on. In order to estimate the model parameters, various methods using points, lines and conics have also been proposed in Refs. 6 and 7.

The ideal fish-eye lenses are constructed with the aim of complying with the equidistant projection function.[8,9] It has been demonstrated that under this model, parallel lines are projected onto circular arcs and all the circles intersect at two vanishing points, thus the centers of the circles are collinear.[10] This property is of great importance because it allows fish-eye camera to be calibrated using single image of two sets of parallel lines in the scene, which is very convenient for practical applications. As a result, fitting of center collinear circles lies at the core of fish-eye camera calibration.

Based on the center colinearity property, Geyer et al.[11] proposed a two-step fitting approach. They first fitted the circles separately and then refined the centers of the circles using the colinearity constraint. However, its accuracy is limited due to the fact that the estimated center collinear circles do not necessarily intersect at two points. Recently, Hughes et al.[10] proposed an iterative approach to solve the fitting problem. Instead of fitting the circles separately, they utilized the available data points to fit all the circles simultaneously. On each iteration, they fixed one vanishing point and updated the other one, and enforced all the circles to intersect at the two vanishing points. Obviously this approach is more robust against noise thus can greatly improve the accuracy.

In this paper we propose a direct approach for fitting of center collinear circles. First we build a coordinate system for the circles and give the objective function for nonlinear optimization. Then we use Levenberg-Marquardt (LM) algorithm[12] to solve the optimization problem. The main advantage of our approach is that two vanishing points are updated simultaneously. Experimental results show that the accuracy of our approach is the same as reported earlier,[10] while its speed is much faster.

The remainder of this paper is organized as follows. Section 2 briefly describes the equidistant fish-eye projection. Section 3 presents three fitting approaches, including two existing approaches and the proposed direct approach. Section 4 gives our experimental results on both synthetic and real data, and Sec. 5 offers our conclusion.

2 Equidistant Fish-Eye Projection

Ideal fish-eye cameras are manufactured to follow the equidistance mapping function such that the distance between a projected point and the optical center of the image is

Paper 12427 received Oct. 12, 2012; accepted for publication Dec. 6, 2012; published online Jan. 7, 2013.

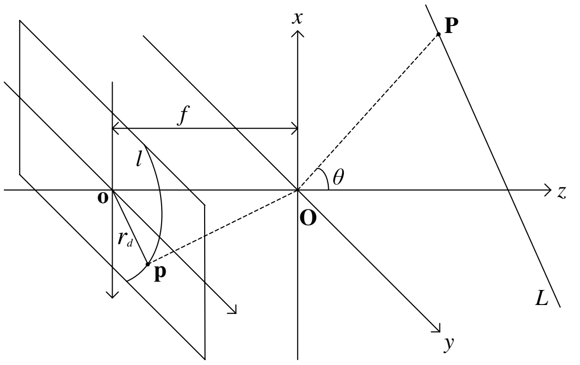

Fig. 1 Equidistant projection.

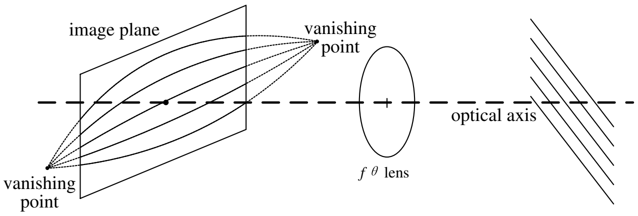

Fig. 2 Projection of parallel lines in an equidistant fish-eye camera system.

proportional to the incident angle of the projected ray, scaled only by the equidistance parameter f, as described by the projection equation:[13,14]

$$r_d = f\theta, \tag{1}$$

where r_d is the fish-eye radial distance of a projected point from the center, and θ is the incident angle of a ray from the 3-D point being projected to the image plane.[15] Figure 1 illustrates the equidistant projection for a simple model of an equidistant camera system.

Due to the radial distortion introduced by fish-eye lenses, straight lines in the scene will map to conics in the image. It has been proved that under this model, parallel lines are projected onto circular arcs and all the circles intersect at two vanishing points,[10] as shown in Fig. 2. It has also been demonstrated that if the parameters of two sets of circles are known, all the intrinsic parameters required for fish-eye camera calibration can be determined.[10] As a result, estimating the parameters of the circles lies at the core of equidistant fish-eye camera calibration. In the next section, we will present three fitting approaches for parameters estimation, including two existing approaches and the proposed direct approach.

3 Fitting of Center Collinear Circles

Fitting of single circle using a set of data points is a well-investigated nonlinear least square problem.[16] However, little attention has been paid to multiple circles fitting. In this paper, we will focus on the fitting of center collinear circles, i.e., a set of circles whose centers are collinear, as illustrated in Fig. 3. The most important property of center collinear circles is that all of them intersect at two points. Thus fitting of the circles is always coupled with the estimation of the positions of the two points. In this section, we will first briefly describe two existing methods in the literature, and then introduce the proposed fitting approach.

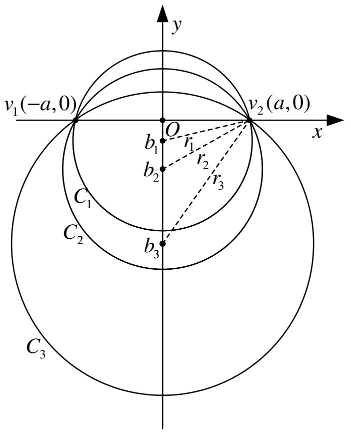

Fig. 3 The coordinate system for the proposed direct approach.

3.1 *Two-Step Approach*

Geyer et al.[11] proposed a two-step approach to estimate the parameters of center collinear circles for catadioptric camera calibration. The approach is briefly described as follows.

1. Separating the data points into several sets of points. Fitting one circle to each set of points.
2. Fitting a line to the centers of all the circles. Updating the centers by projecting them onto this line.

This approach first estimates the parameters of each circle separately, and then enforces the centers of the circles to be collinear. Note that it does not update the radii of the circles, and therefore, it cannot guarantee that all the circles intersect at two points.

3.2 *Iterative Approach*

Recently, Hughes et al. proposed an iterative approach to alternately update one point by fixing the other.[10] This approach consists of the following steps.

1. Separating the data points into several sets of points. Fitting one circle to each set of points.
2. Calculating the intersect points of the circles and determining the initial positions of the two points.
3. Fixing one point and updating the other using LM optimization.
4. Repeating step (3) until convergence is observed or the maximum number of iterations is reached.

It is obvious that when the algorithm terminates, we could obtain the positions of the two points as well as the parameters of the circles. Note that this approach enforces the circles to intersect at two points, thus it is expected to be more accurate than the two-step approach. However, since each time the approach updates only one point, it would take a number of iterations for convergence.

3.3 *Direct Approach*

In this subsection, we present a direct approach for fitting of center collinear circles. We first build a coordinate system to facilitate derivations, and then give the equations of the center collinear circles using the vanishing points constraint. Next we propose to solve the fitting problem using LM optimization, and give the objective function as well as its Jacobian. Finally we discuss some issues related to its implementation.

Assume the center collinear circles intersect at two vanishing points v_1 and v_2. After translation $\mathbf{T} = \mathbf{T}(x, y)$ and rotation $\mathbf{R} = \mathbf{R}(\theta)$, we define the midpoint of line segment $\overline{v_1 v_2}$ as the origin, and the line determined by v_1 and v_2 as the X-axis, as shown in Fig. 3. In this coordinate system, the center of the circle C_i is $(0, b_i)$, and the coordinates of v_1 and v_2 are $(-a, 0)$ and $(a, 0)$, respectively. The equation of circle C_i can be expressed as

$$x^2 + (y - b_i)^2 = r_i^2, \tag{2}$$

where the radius

$$r_i = \sqrt{a^2 + b_i^2}. \tag{3}$$

In order to fit N circles, we should estimate $N + 4$ parameters, i.e., x, y, θ, a, b_1, $\cdots$, b_N.

After translation and rotation, a data point p is transformed from (m, n) to (m', n'), which satisfies

$$\begin{aligned} \begin{bmatrix} m' \\ n' \end{bmatrix} &= \mathbf{R}(\mathbf{P} - \mathbf{T}) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} m - x \\ n - y \end{bmatrix} \\ &= \begin{bmatrix} (m - x)\cos\theta - (n - y)\sin\theta \\ (m - x)\sin\theta + (n - y)\cos\theta \end{bmatrix}. \end{aligned} \tag{4}$$

If there are N_i data points $p_{i,1}, \ldots, p_{i,N_i}$ belonging to circle C_i, and the distance between the k-th transformed point $p'_{i,k}$ and its corresponding center is

$$d_{i,k} = \|p'_{i,k} - C_i\| = \sqrt{(m'_{i,k})^2 + (n'_{i,k} - b_i)^2}, \tag{5}$$

then the fitting error of the points belonging to circle C_i is given by

$$\begin{aligned} f_i &= \sum_{k=1}^{N_i} (d_{i,k} - r_i)^2 \\ &= \sum_{k=1}^{N_i} \left[\sqrt{(m'_{i,k})^2 + (n'_{i,k} - b_i)^2} - \sqrt{a^2 + b_i^2} \right]^2, \end{aligned} \tag{6}$$

and the fitting error of all the data points is

$$F = \sum_{i=1}^{N} f_i = \sum_{i=1}^{N} \sum_{k=1}^{N_i} (d_{i,k} - r_i)^2. \tag{7}$$

Substituting Eqs. (4) and (6) into Eq. (7), we get

$$F = \sum_{i=1}^{N} \sum_{k=1}^{N_i} \left[\sqrt{(m_{i,k} - x)^2 + (n_{i,k} - y)^2 + b_i^2 - 2b_i(m_{i,k} - x)\sin\theta - 2b_i(n_{i,k} - y)\cos\theta} - \sqrt{a^2 + b_i^2} \right]^2. \tag{8}$$

Thus the fitting problem is formulated as a nonlinear least square problem whose objective is finding the minima of Eq. (8).

LM algorithm[12] is very efficient for the above optimization problem. In order to apply LM algorithm, we should derive the first partial derivative of the objective function, i.e., the Jacobian.[17] After some mathematical manipulation, we get the Jacobian

$$\begin{aligned} \frac{\partial f_i}{\partial x} &= \frac{-(m - x) + b_i \sin\theta}{d} \\ \frac{\partial f_i}{\partial y} &= \frac{-(n - y) + b_i \cos\theta}{d} \\ \frac{\partial f_i}{\partial \theta} &= \frac{b_i}{d}[-(m - x)\cos\,\theta + (n - y)\sin\,\theta] \\ \frac{\partial f_i}{\partial a} &= -\frac{a}{r_i} \\ \frac{\partial f_i}{\partial b_j} &= \begin{cases} \frac{b_i - (m - x)\sin\theta - (n - y)\cos\theta}{d} - \frac{b_i}{r_i} & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{9}$$

Setting the initial estimation properly is very crucial for LM optimization. Since the fitting error increases with the radius of circle, we propose to first fit each circle separately, and then use the intersect points of two smallest circles as the initial estimation of the vanishing points. The initial values used for LM optimization then can be determined by the two vanishing points and the parameters of the circles. After convergence of LM algorithm, we can compute the parameters of the circles. The complete approach is given as follows:

1. Separating the data points into several sets of points. Fitting one circle to each set of points.
2. Finding two smallest circles and calculating the intersect points of them.
3. Calculating the initial estimation of the parameters x, y, θ, a, b_1, $\cdots$, b_N.
4. Updating the parameters using LM optimization until convergence.
5. Calculating the parameters of the circles using the optimization result.

Note that our method is also an iterative approach because LM algorithm is used for parameter estimation and LM algorithm often requires a number of iterations. However, in the iterative approach presented in Sec. 3.2, the vanishing points are estimated sequentially, while in our approach they are estimated simultaneously. Thus our approach is expected to be faster than the iterative approach. In addition, our approach also enforces the circles to intersect at two points, thus it is expected to be more accurate than the two-step approach.

4 Experimental Results

We have evaluated our method on both synthetic and real data and have compared it with those proposed.[10,11] We use speed and accuracy as performance indicators for comparison. The platform is a PC with Intel CPU i5-2400 3.10 GHz and 8 G RAM. The software environment is Windows 7 Ultimate and Visual Studio 2005. All the testing programs are written in C++ language.

4.1 *Synthetic Data*

In order to stimulate the fish-eye camera calibration, we generate eight center collinear circles (denoted as $C1$ to $C8$), and use only arcs of them for parameter estimation, as shown in Fig. 4. The resolution of the synthetic image is 640 × 480, and the parameters of the circles are listed in Table 1.

On each arc we randomly choose 100 points. Gaussian noise with zero-mean and σ standard deviation is added to the points. The noise levels σ are 0, 1, 2, 3, 4, 5, respectively. For each noise level, we perform 100 independent trials, and the mean values and standard deviations of these recovered parameters are computed over each run. The errors compared to the ground truth are defined as

$$\mathrm{error}_{C_x} = \frac{1}{T}\sum_{t=1}^{T}|C_x - C^*_{x,t}|,$$

$$\mathrm{error}_{C_y} = \frac{1}{T}\sum_{t=1}^{T}|C_y - C^*_{y,t}|, \qquad \mathrm{error}_{r} = \frac{1}{T}\sum_{t=1}^{T}\frac{|r - r^*_t|}{r}.$$

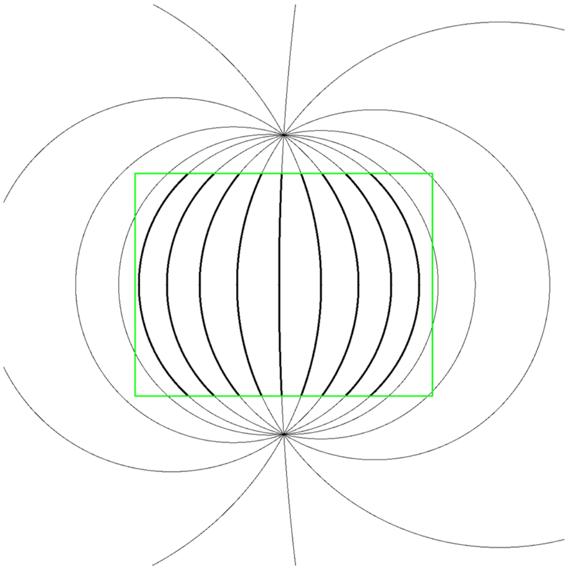

Fig. 4 The synthetic image of center collinear circles.

where C_x, C_y and r are the ground truth, $C^*_{x,t}$, $C^*_{y,t}$, and r^*_t are the estimated parameters of t-th trail, and T is the number of trails. Although the fitting error increases with the noise level, the comparison results of the three approaches are similar for different noise levels. For the sake of brevity, we only provide the fitting results when $\sigma = 3$, as shown in Tables 2–4. The three fitting approaches described in Sec. 3 are indicated by subscripts t, i and d, respectively.

From Tables 2–4, we can see that the fitting results are almost the same for the iterative and the direct approaches, and both of them are much more robust and accurate than the two-step approach. The error and standard derivation of the two-step approach are around 5 to 10 times larger than those of the other two approaches. Thus we can conclude that in terms of fitting accuracy both the iterative and the direct approaches are superior to the two-step approach.

Another observation from Tables 2–4 is that for all the three approaches, the error and standard derivation increase with the radius of the circle, which is consistent with previous studies.[18,19] The reason is that for a larger circle, the arc in the image corresponds to a smaller angle. As a result, the accuracy decreases due to a larger amount of occlusion of the circle. On the other hand, because the iterative and the direct approaches use all the data points for parameters estimation, the adverse effect of noise and occlusion can be greatly reduced by the vanishing points constraint. Hence these two approaches are more robust and accurate than the two-step approach.

The time requirements of the three approaches are reported in Table 5. It can be observed that the two-step approach is the fastest one, which only requires less than 16 ms on average when $\sigma = 3$. The other two approaches are much slower. The time required by the iterative and the direct approaches are about 700 and 30 times more than that of the two-step approach, respectively. It is reasonable because nonlinear optimization is adopted by the direct approach using all the data points and the center collinear constraint, which costs much more time than just estimating the parameters of each circle separately. The iterative approach often requires several tens of iterations for convergence (as listed in Table 6), and there is also a nonlinear optimization step on each iteration. As a result, the iterative approach is the most time-consuming of all. Our method is more than 20 times faster than the iterative approach, and its average running time is less than half a second, which makes it very promising for practical calibration procedures.

4.2 *Real Data*

In this expirment, we take an image using a fish-eye camera with FOV about 170 deg. A chess board is placed in front of the camera for calibration. The size of each block in the chess board is 15 × 15 mm^2, and the resolution of the image is

Table 1 Parameters of the synthetic circles.

	*C*1	*C*2	*C*3	*C*4	*C*5	*C*6	*C*7	*C*8
(C_x, C_y)	(31.55, 0)	(107.61, 0)	(240, 0)	(600, 0)	(−462, 0)	(−194.44, 0)	(−79.80, 0)	(−10.16, 0)
r	321.55	337.61	400	680	562	374.44	329.80	320.16

Note: (C_x, C_y) is the center of the circle. The center of the image (320, 240) is regarded as the origin. r is the radius of the circle.

Table 2 Fitting results of C_x when $\sigma = 3$.

	*C*1	*C*2	*C*3	*C*4	*C*5	*C*6	*C*7	*C*8
GT	31.55	107.61	240	600	−462	−194.44	−79.80	−10.16
mean$_t$	33.46	108.84	240.29	611.15	−462.04	−194.54	−81.28	−10.35
mean$_i$	31.10	107.11	240.34	601.35	−461.62	−195.56	−79.53	−10.41
mean$_d$	31.09	107.10	240.32	601.27	−461.64	−195.56	−79.53	−10.41
std$_t$	6.20	6.81	11.40	37.41	24.38	9.23	7.31	7.20
std$_i$	0.69	0.93	1.63	5.88	4.33	1.64	0.90	0.71
std$_d$	0.69	0.93	1.63	5.88	4.33	1.64	0.90	0.71
error$_t$	5.40	5.48	9.57	31.42	20.59	7.21	6.01	6.14
error$_i$	0.63	0.83	1.26	4.69	3.29	1.59	0.73	0.57
error$_d$	0.64	0.83	1.25	4.65	3.29	1.59	0.73	0.57

Table 3 Fitting results of C_y when $\sigma = 3$.

	*C*1	*C*2	*C*3	*C*4	*C*5	*C*6	*C*7	*C*8
GT	0	0	0	0	0	0	0	0
mean$_t$	0.013	−0.086	−0.253	−0.769	0.684	0.324	0.172	0.072
mean$_i$	0.003	−0.008	−0.028	−0.080	0.075	0.036	0.019	0.009
mean$_d$	0.003	−0.008	−0.028	−0.080	0.075	0.036	0.019	0.009
std$_t$	0.91	1.04	1.39	2.68	1.91	1.09	0.90	0.89
std$_i$	0.16	0.19	0.27	0.55	0.41	0.22	0.17	0.16
std$_d$	0.16	0.19	0.27	0.55	0.41	0.22	0.16	0.16
error$_t$	0.74	0.85	1.14	2.19	1.56	0.88	0.73	0.72
error$_i$	0.13	0.15	0.21	0.42	0.33	0.18	0.14	0.13
error$_d$	0.13	0.15	0.21	0.42	0.33	0.18	0.14	0.13

640 × 480, as depicted in Fig. 5. Curve extracting is accomplished by a software package developed by our lab. On both horizontal and vertical directions, only six arcs with the largest lengths are selected for parameters estimation. After the fitting of center collinear circles, we can obtain the positions of horizontal vanishing points v_{x1} and v_{x2} and the vertical vanishing points v_{y1} and v_{y2}. If the line joins v_{x1} and v_{x2} is denoted as l_x, and that joins v_{y1} and v_{y2} is denoted as l_y, then the optical center can be determined as the intersect point of l_x and l_y, and the focal lengths on horizontal and vertical directions are given by

$$f_x = \frac{|v_{x1} - v_{x2}|}{\pi} \tag{10}$$

and

$$f_y = \frac{|v_{y1} - v_{y2}|}{\pi}. \tag{11}$$

Then we can use Eq. (1) for distortion correction. The details about distortion correction are available.[15]

Due to lack of the ground truth of parameters of the fish-eye camera, we just check the undistorted image to evaluate the calibration results and the proposed approach. The undistorted image is shown in Fig. 6. As can be seen in this figure, straight lines in the scene now map to straight lines in the image. We then extract the 8 × 8 corners of the chess board[20] and compute the reconstruction error. The average difference

Table 4 Fitting results of r when $\sigma = 3$.

	***C*1**	***C*2**	***C*3**	***C*4**	***C*5**	***C*6**	***C*7**	***C*8**
GT	321.55	337.61	400	680	562	374.44	329.80	320.16
mean_t	323.73	338.87	400.29	690.83	562.18	374.34	331.33	320.21
mean_i	321.58	337.54	400.31	681.31	561.66	375.03	329.77	320.23
mean_d	321.57	337.53	400.29	681.22	561.67	375.03	329.77	320.22
std_t	4.94	5.49	9.83	35.97	22.90	7.89	5.92	5.64
std_i	0.59	0.79	1.48	5.74	4.19	1.45	0.78	0.56
std_d	0.59	0.79	1.47	5.73	4.19	1.45	0.78	0.56
error_t	1.34e-2	1.30e-2	2.04e-2	4.43e-2	3.43e-2	1.65e-2	1.48e-2	1.48e-2
error_i	1.39e-3	1.82e-3	2.83e-3	6.75e-3	5.67e-3	3.24e-3	1.81e-3	1.35e-3
error_d	1.39e-3	1.82e-3	2.82e-3	6.70e-3	5.67e-3	3.24e-3	1.81e-3	1.34e-3

Table 5 The average time (ms) required by the three approaches.

σ	0	1	2	3	4	5
Two-step approach	8.27	11.55	13.01	14.35	14.88	15.73
Iterative approach	5490	7440	8845	9563	11364	11467
Direct approach	285.7	349.5	386.6	403.9	424.4	438.4

Table 6 The average iterations required by the iterative approach for convergence.

σ	0	1	2	3	4	5
Avg. Iters	21.53	24.19	29.03	31.25	36.76	37.01

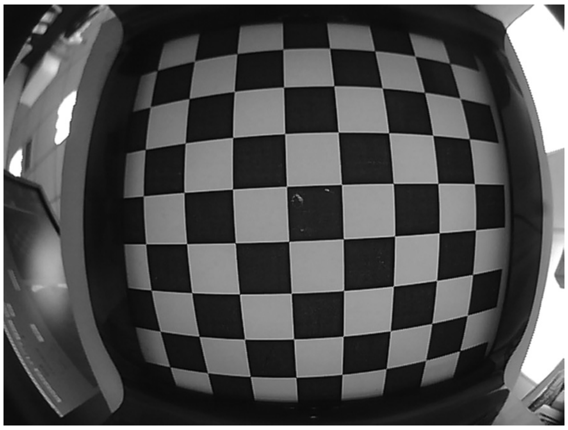

Fig. 5 An image captured using a fish-eye camera.

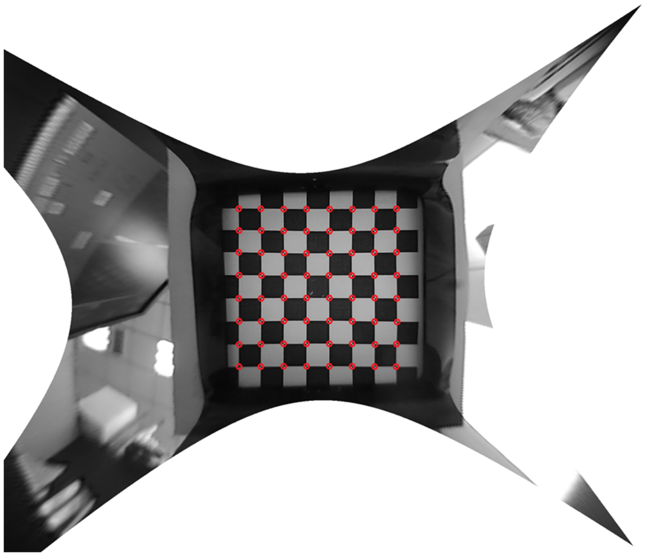

Fig. 6 The image after distortion correction.

between the position of each corner and their true position is 1.631 pixel. Because the resolution of each block in the undistorted image is 63.267×63.267, the reconstruction error can be computed as $1.631/63.267 \times 15 = 0.387$ mm. Thus we conclude that our method is effective for fish-eye camera calibration.

5 Conclusions

In this paper we propose a novel method for fitting of center collinear circles for fish-eye camera calibration. We formulate the fitting problem as a nonlinear least square problem and solve it using LM optimization. Experimental results on synthetic data show that the proposed method is much more accurate than the two-step approach, and it requires much less time than the iterative approach while keeping the fitting performance unspoiled. Results on real data demonstrate its effectiveness for fish-eye camera calibration. Hence we can conclude that the proposed method is very promising for practical applications.

Acknowledgments

This work is partially supported by Beijing Postdoctoral Research Foundation (2011ZZ-57), Postdoctoral Science Foundation of China (2011M500013), and Overseas Talents Attracting Plan of BJAST (OTP-2012-012).

References

1. X. H. Ying, Z. Y. Hu, and H. B. Zha, "Fisheye lenses calibration using straight-line spherical perspective projection constraint," in *Proc. 8th Asian Conf. Comput. Vision (ACCV)*, pp. 61–70, Springer, Berlin, Germany (2006).
2. C. Burchardt and K. Voss, "A new algorithm to correct fish-eye and strong wide-angle-lens-distortion from single images," in *Proc. Int. Conf. Image Process. (ICIP)*, pp. 225–228, IEEE CS Press, Los Alamitos, California (2001).
3. S. Shah and J. K. Aggarwal, "Intrinsic parameter calibration procedure for a (high-distortion) fish-eye lens camera with distortion model and accuracy estimation," *Pattern Recognit.* **29**(11), 1775–1788 (1996).
4. A. W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp. 125–132, IEEE CS Press, Los Alamitos, California (2001).
5. D. Claus and A. W. Fitzgibbon, "A rational function lens distortion model for general cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pp 213–219, IEEE CS Press, Los Alamitos, California (2005).
6. J. Kannala and S. Brandt, "A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses," *IEEE Trans. Pattern Anal. Mach. Intell.* **28**(8), 1335–1340 (2006).
7. J. P. Barreto and H. Araujo, "Fitting conics to paracatadioptric projections of lines," *Comput. Vis. Image. Und.* **101**(3), 151–165 (2006).
8. D. B. Gennery, "Generalized camera calibration including fish-eye lenses," *Int. J. Comput. Vis.* **68**(3), 239–266 (2006).
9. C. Hughes et al., "Accuracy of fish-eye lens models," *Appl. Opt.* **49**(17), 3338–3347 (2010).
10. C. Hughes et al., "Equidistant ($f\theta$) fish-eye perspective with application in distortion centre estimation," *Image Vis. Comput.* **28**(3), 538–551 (2010).
11. C. Geyer and K. Daniilidis, "Catadioptric camera calibration," in *Proc. 7th Int. Conf. Comput. Vis. (ICCV)*, pp. 398–404, IEEE CS Press, Los Alamitos, CA (1999).
12. D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *SIAM J. Appl. Math.* **11**(2), 431–441 (1963).
13. K. Miyamoto, "Fish eye lens," *J. Opt. Soc. Am.* **54**(8), 1060–1061 (1964).
14. C. C. Slama, *Manual of Photogrammetry*, American Society for Photogrammetry and Remote Sensing, Bethesda, Maryland (1980).
15. C. Hughes et al., "Equidistant fish-eye calibration and rectification by vanishing point extraction," *IEEE Trans. Pattern Anal. Mach. Intell.* **32**(12), 2289–2296 (2010).
16. Z. Y. Zhang, "Parameter estimation techniques: a tutorial with application to conic fitting," *Image Vis. Comput.* **15**(1), 59–76 (1997).
17. K. Madsen, H. B. Nielsen, and O. Tingleff, *Methods for Non-Linear Least Squares Problems*, 2nd ed., Technical University of Denmark (2004).
18. X. Ying and Z. Hu, "Catadioptric camera calibration using geometric invariants," *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(10), 1260–1271 (2004).
19. A. Fitzgibbon and R. Fisher, "A buyer's guide to conic fitting," in *Proc. 6th British Machine Vision Conf.*, pp. 513–522, BMVA Press, Manchester, United Kingdom (1995).
20. C. Harris and M. Stephens, "A combined corner and edge detector," in *Alvey Vision Conference*, pp. 147–151, BMVA Press, Manchester, United Kingdom (1988).

Feng Yue received his BSc and PhD degrees in computer science from the Harbin Institute of Technology (HIT), in 2003 and 2010, respectively. He is now a postdoctoral researcher in the Pattern Recognition Research Center, Beijing Institute of New Technology Applications. His current research interests include computer vision and pattern recognition.

Bin Li received his MSc and PhD degrees in computer science from Harbin Institute of Technology (HIT), Harbin, China, in 2000 and 2006, respectively. From 2006 to 2008, he worked in School of Computer Science and Technology, HIT, as a lecturer. From 2008 to now, he has been working in Key Laboratory of Pattern Recognition of Beijing Academy of Science and Technology as a director, and in Beijing Institute of New Technology Applications as an associate research fellow and deputy director. So far, he has published over 15 papers and two books. His research interests include signal processing, pattern recognition and biometrics, etc.

Ming Yu received his BS degree from Beijing University of Post and Telecommunications (BUPT), Beijing, China in 1986, the MS degree from Hebei University of Technology (HUT), Tianjin, China in 1989, and his PhD degree in communication and information systems from Beijing Institute of Technology (BIT), Beijing, China in 1999, respectively. Since 1989, he has been with the Department of Computer Science and Engineering, Hebei University of Technology, where he is currently a professor. His current research interests include image/video understanding, intelligent media processing and pattern recognition.