Front. Optoelectron. 2015, 8(3): 319–328
DOI 10.1007/s12200-015-0453-7

RESEARCH ARTICLE

Lens distortion correction based on one chessboard pattern image

Yubin WU, Shixiong JIANG, Zhenkun XU, Song ZHU, Danhua CAO (✉)

School of Optical and Electronic Information, Huazhong University of Science and Technology, Wuhan 430074, China

Abstract This paper proposes a detection method of chessboard corner to correct camera distortions –including radial distortion, decentering distortion and prism distortion. This proposed method could achieve high corner detection rate. Then we used iterative procedure to optimize distortion parameter to minimize distortion residual. In this method, first, non-distortion points are evaluated by four points near image center; secondly, Levenberg-Marquardt nonlinear optimization algorithm was adopted to calculate distortion parameters, and then to correct image by these parameters; thirdly, we calculated corner points on the corrected image, and repeated previous two steps until distortion parameters converge. Results showed the proposed method by iterative procedure can make the impact of slight distortion around image center negligible and the average of distortion residual of one line is almost 0.3 pixels.

Keywords computer vision, camera distortion, distortion correction, corner detection

1 Introduction

Precise calibration for a camera is fundamental in computer vision and vision metric. A camera is often modeled as an ideal pin-hole one, which images on the focus plane without distortion. However, lens distortion will make the image distorted from ideal one.

There are several distortion models depending on the type of lens. In this paper, radial, decentering and prism distortion models proposed by Brown [1,2] were to be considered.

To identify each distortion model, several methods have been proposed. The first method uses the known object 3D world coordinates [3], with which the result could be inaccurate when both camera intrinsic and extrinsic parameters are estimated at the same time. The second method uses point correspondences in different views [4]. This method is based on fundament matrix without the intrinsic and extrinsic explicitly. However, it is difficult to find correct point correspondences in multi-views. The third method, which estimates the distortion without camera parameters, is based on projective invariants [5–7]. Nowadays, more and more researches depend on this method. This method estimates the distortion without camera parameters. The most widely used projective invariant is straight line, which remains straight from different view if there is no lens distortion based on pin-hole camera. This method needs to know the scene in advance, as it is important to find out the straight line or other invariant objects in distortion image. The forth method is also based on point correspondences between image points and world points [8], and it uses planar invariant instead. Planar calibration pattern points and non-distortion image points are mapped by homography matrix. This method calculates the distortion model parameters to make the correspondence most fit.

There are two different ways to compute the distortion models, backward mapping and forward mapping. Zhang [4] used forward mapping in both model definition and computing, while Hartley and Kang [9] used backward mapping.

In this paper, a new camera calibration method is presented, we used only one chessboard pattern image to solve the distortion models, which includes three main types of lens distortion–radial, decentering and prism distortions. It has four steps. First, image feature points are detected by chessboard point detecting algorithm. Second, the non-distortion feature point coordinate is calculated from detected feature point on distortion image. Third, nonlinear optimization is done to get the distortion model.

Received June 13, 2014; accepted September 5, 2014

E-mail: dhcao@mail.hust.edu.cn

Last, iterate former the three steps until converge. This method is based on the idea that the distortion near the optic axis is tiny, and we can use iterative procedure to reduce influence of this distortion. Therefore, non-distortion feature points can be calculated using the near center feature points on distortion image.

The paper is organized as follows. Section 2 describes the camera lens distortion model. Section 3 proposes the method of distortion model calculating. In Section 4, several experimental results are reported on both real and synthetic data. The paper ends with some concluding remarks.

2 Lens distortion model

Lens distortion makes the actual image different from ideal pin-hole image. Three forms of lens distortion, namely radial distortion, decentering distortion and thin prism distortion are taken into account in this paper [8]. The distortion can be formed as follow:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x^* \\ y^* \end{bmatrix} + \begin{bmatrix} \Delta_{x^*} \\ \Delta_{y^*} \end{bmatrix}. \tag{1}$$

$[x,y]^{\mathrm{T}}$ is the ideal image without distortion, $[x^*,y^*]^{\mathrm{T}}$ is the actual image with distortion, $[\Delta_{x^*},\Delta_{y^*}]^{\mathrm{T}}$ is the amount of distortion. Different forms of lens distortion result in the different part of distortion factor.

2.1 Radial distortion

Radial distortion is caused by inconsistent magnification in the field of view. This distortion is the significant part of lens distortion. Many researches only dealt with radial distortion [5,7,10].

The amount of radial distortion is circularly symmetric about optical center as shown in Fig. 1, which means there is no distortion at the optical center. In general, the farther the image point is away from the optical axis, the greater the distortion is. Radial distortion can be expressed as follows:

$$\begin{cases} \Delta_{rx}(x^*,y^*) = k_1 x^* \left((x^*)^2+(y^*)^2\right) + k_2 x^* \left((x^*)^2+(y^*)^2\right)^2, \\ \Delta_{ry}(x^*,y^*) = k_1 y^* \left((x^*)^2+(y^*)^2\right) + k_2 y^* \left((x^*)^2+(y^*)^2\right)^2, \end{cases} \tag{2}$$

where k_1, k_2 are distortion factors of radial distortion model.

2.2 Decentering distortion

Decentering distortion is due to non-strict collineation of the optical centers of lens elements, as shown in Fig. 2. The decentering distortion model is formed as Eq. (3).

$$\begin{cases} \Delta_{dx}(x^*,y^*) = p_1\left(3(x^*)^2 + (y^*)^2\right) + 2p_2 x^* y^*, \\ \Delta_{dy}(x^*,y^*) = p_2\left((x^*)^2 + 3(y^*)^2\right) + 2p_1 x^* y^*, \end{cases} \tag{3}$$

where p_1, p_2 are distortion factors of decentering distortion model.

2.3 Thin prism distortion

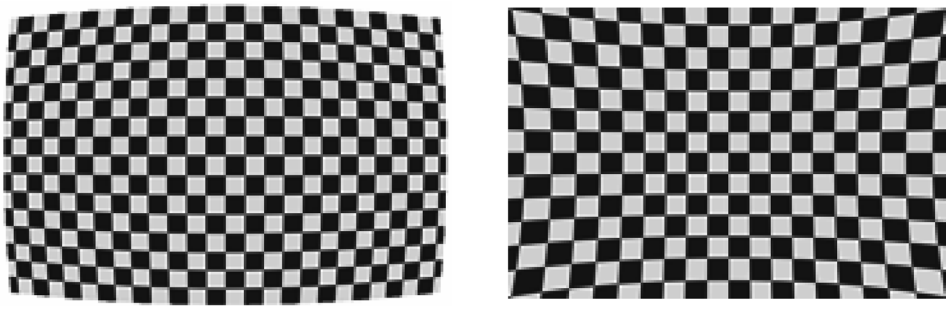
Thin prism distortion arises from the imperfection in lens design and manufacturing as well as camera, as shown in Fig. 3.The thin prism distortion model is described in Eq. (4).

$$\begin{cases} \Delta_{px}(x^*,y^*) = s_1\left((x^*)^2 + (y^*)^2\right), \\ \Delta_{py}(x^*,y^*) = s_2\left((x^*)^2 + (y^*)^2\right), \end{cases} \tag{4}$$

where s_1, s_2 are distortion factors of prism distortion model.

In this paper, three forms of distortion are considered. Then, the total amount of distortion is the sum of three distortions.

$$\begin{cases} \Delta_{x^*}(x^*,y^*) = \Delta_{rx}(x^*,y^*) + \Delta_{dx}(x^*,y^*) + \Delta_{px}(x^*,y^*), \\ \Delta_{y^*}(x^*,y^*) = \Delta_{ry}(x^*,y^*) + \Delta_{dy}(x^*,y^*) + \Delta_{py}(x^*,y^*). \end{cases} \tag{5}$$



(a) (b)

Fig. 1 (a) Barrel distortion image; (b) pincushion distortion image

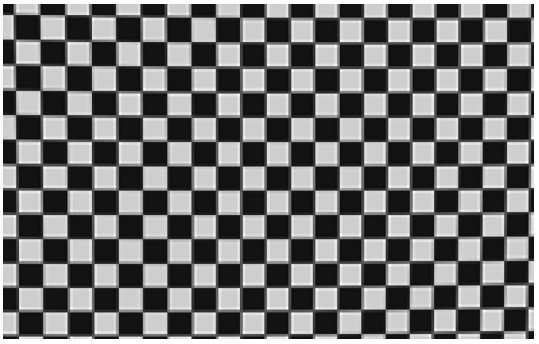


Fig. 2 Image with decentering distortion

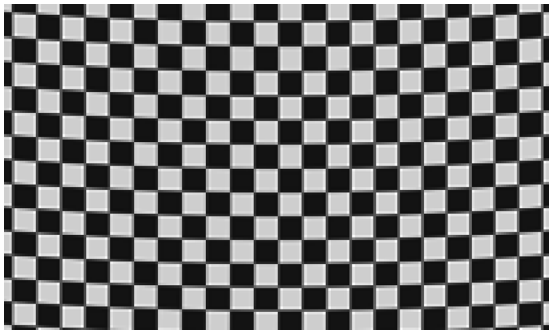


Fig. 3 Image with thin prism distortion

3 Distortion correct with one chessboard pattern image

The proposed method consists following steps: 1) preprocess chessboard image and pick up the coordinate of corner points; 2) reconstruct non-distortion corner coordinate from distortion image; 3) initialize distortion model parameters using Levenberg-Marquardt nonlinear algorithm (LM); 4) use iterative procedure to minimize distortion residual; 5) use forward mapping method to correct image.

3.1 Corner points detection on distortion image

The performance of distortion correction is affected by the accuracy of the corner point coordinate. In this paper, a new corners detecting method of chessboard pattern image is proposed. This method has high corner detection rate.

The proposed method detects corner points according to the dissimilar distribution of pixels gray-scale between corner region and non-corner region. It defines three operators to calculate the dissimilarity distribution on different directions as follows. S is centrosymmetric operator. V is vertical symmetry operator. H is horizontal symmetry operator.

$$\begin{cases} s(x_0,y_0) = \sum\limits_{(x,y)\in neighborhood(x_0,y_0)} \left| p_{(x,y)} - p_{(2x_0-x,2y_0-y)} \right| \\ \qquad /card\left(neighborhood(x_0,y_0)\right), \\ v(x_0,y_0) = \sum\limits_{(x,y)\in neighborhood(x_0,y_0)} \left| p_{(x,y)} - p_{(x,2y_0-y)} \right| \\ \qquad /card\left(neighborhood(x_0,y_0)\right), \\ h(x_0,y_0) = \sum\limits_{(x,y)\in neighborhood(x_0,y_0)} \left| p_{(x,y)} - p_{(2x_0-x,y)} \right| \\ \qquad /card\left(neighborhood(x_0,y_0)\right), \end{cases} \tag{6}$$

where $neighborhood(x_0,y_0)$ is 8×8 rectangle region with point (x_0,y_0) located at the center, *card* ( ) returns the pixel number of the region.

Define M as pixels response, whose value is associated with three operators. The response value of corner point is calculated by the following steps:

1) Compute the value of pixels by the three operators;

2) If all of the three operator results are between the predefined ranges, the final response value is the result by operator S. Otherwise, M is set to zero.

$$M=\{s|s\in[s_{\min},s_{\max}],v\in[v_{\min},v_{\max}],h\in[h_{\min},h_{\max}]\}$$
$$\cup\{0|s\notin[s_{\min},s_{\max}] \text{ or } v\notin[v_{\min},v_{\max}] \text{ or } h\notin[h_{\min},h_{\max}]\}. \tag{7}$$

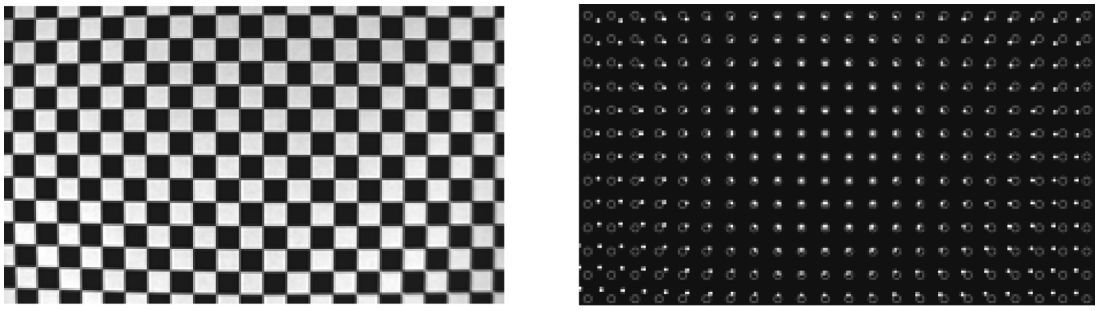
Median filtering algorithm is used to remove isolated detective point which is fault detection caused by noise. The corner coordinate can be computed by calculating the connected region centroid of response M, described as follows:

$$centroid = \frac{\sum\limits_{connected} m*position_m}{\sum\limits_{connected} m}, \tag{8}$$

where $m \in M$, connected region is detected by 8-adjacent connection, $position_m$ is the coordinate of the point with value m.

In this paper, distortion of the four corner points around the center is tiny for distortion image. Non-distortion points are reconstructed by the geometry feature (considering the same size of each block) of chessboard, which exploits these four corner points, as shown in Fig. 4, defined in Eq. (9).

$$\begin{cases} x_{(i,j)} = x_{(0,0)} + j*\left(x_{(0,1)} - x_{(0,0)}\right) + i*\left(x_{(1,0)} - x_{(0,0)}\right), \\ y_{(i,j)} = y_{(0,0)} + j*\left(y_{(0,1)} - y_{(0,0)}\right) + i*\left(y_{(1,0)} - y_{(0,0)}\right), \end{cases} \tag{9}$$



(a) (b)

Fig. 4 (a) Distortion image; (b) the dots are detected corners in image (a) and the circles are ideal points location

where (i,j) is the points array order, i and j could be negative integer, center four points are (0,0), (0,1), (1,0), (1,1).

To minimize the influence of the distortion of four center points, this paper introduces iterative process. The distortion of the four points would reduce after image correction, and then the corrected image is treated as an input distortion image for next iteration to reconstruct non-distortion points, iterate this procedure until four points unchanged after correction.

3.2 Distortion model calculate

Three forms of distortion indicate that the optical center region is always non-distortion.

As long as the distortion and non-distortion points are achieved, the total amount of distortion would be computed by Eq. (1).

Expansion Eq. (5) is the whole model formed by three distortions.

$$\begin{cases} \Delta_x(x^*,y^*) = k_1 x^*\left((x^*)^2+(y^*)^2\right) + k_2 x^*\left((x^*)^2+(y^*)^2\right)^2 \\ \qquad + \left[p_1\left(3(x^*)^2+(y^*)^2\right) + 2p_2 x^* y^*\right] \\ \qquad + s_1\left((x^*)^2+(y^*)^2\right), \\ \Delta_y(x^*,y^*) = k_1 y^*\left((x^*)^2+(y^*)^2\right) + k_2 y^*\left((x^*)^2+(y^*)^2\right)^2 \\ \qquad + \left[p_2\left((x^*)^2+3(y^*)^2\right) + 2p_1 x^* y^*\right] \\ \qquad + s_2\left((x^*)^2+(y^*)^2\right). \end{cases} \tag{10}$$

In this paper, the optical center is set to the image center. Different center only affects the distortion model values, which would not influence the result of distortion correction. Then, the linear equation is showed as follows

$$\boldsymbol{X}-\boldsymbol{X}^* = \boldsymbol{A}\boldsymbol{P}, \tag{11}$$

where

$$\begin{cases} \boldsymbol{A} = \begin{bmatrix} x^*\left((x^*)^2+(y^*)^2\right), x^*\left((x^*)^2+(y^*)^2\right)^2, \\ 3(x^*)^2+(y^*)^2, 2x^*y^*, (x^*)^2+(y^*)^2, 0 \\ y^*\left((x^*)^2+(y^*)^2\right), y^*\left((x^*)^2+(y^*)^2\right)^2, \\ 2x^*y^*, (x^*)^2+3(y^*)^2, 0, (x^*)^2+(y^*)^2 \end{bmatrix}, \\ \boldsymbol{P} = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad s_1 \quad s_2]^{\mathrm{T}}, \\ \boldsymbol{X} = [x \quad y]^{\mathrm{T}}, \\ \boldsymbol{X}^* = [x^* \quad y^*]^{\mathrm{T}}. \end{cases}$$

$\boldsymbol{P}$ is the required model parameters, $\boldsymbol{X}$ is the non-distortion position, $\boldsymbol{X}^*$ is the distortion position.

The model $\boldsymbol{P}$ can be calculated by LM nonlinear optimization [11]. The objective function is defined as follows:

$$\frac{1}{2}\left[\left\|\boldsymbol{X}_0-\boldsymbol{X}(\boldsymbol{P},\boldsymbol{X}_0^*)\right\|^2,\ldots,\left\|\boldsymbol{X}_j-\boldsymbol{X}(\boldsymbol{P},\boldsymbol{X}_j^*)\right\|^2,\ldots, \left\|\boldsymbol{X}_m-\boldsymbol{X}(\boldsymbol{P},\boldsymbol{X}_m^*)\right\|^2\right] = \boldsymbol{0}, \tag{12}$$

where $\boldsymbol{X}_j$ is non-distortion pixel coordinate, $\boldsymbol{X}(\boldsymbol{P}, \boldsymbol{X}_j^*)$ is the pixel coordinate transferred by Eq. (11) related to distortion image.

The Jacobian matrix is used for LM algorithm. From Eq. (12), the Jacobian matrix can be easy to get.

Then, the procedure for calculating distortion parameters is:

1) use corner detection to obtain point coordinate;

2) calculate the non-distortion point location depending on near center point;

3) using LM method to optimize distortion model parameters;

4) calculate the distortion residual, repeat steps 1), 2) and 3) until distortion residual convergence;

The parameter of convergence is the best solution of proposed method.

3.3 Distortion image correction

Image is corrected by the distortion parameters obtained in last steps. This procedure contains two steps.

1) Coordinate transfer: transfer the pixel coordinate from distortion image to non-distortion image (forward-mapping, as shown in Fig. 5) or from non-distortion image to distortion image (backward-mapping). This paper uses forward-mapping method to rectify image.

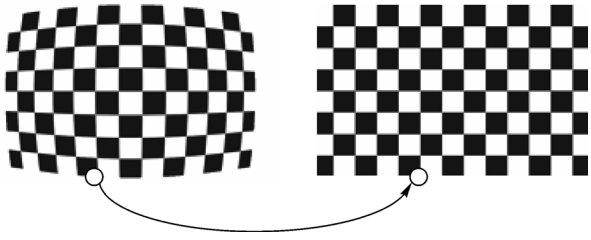


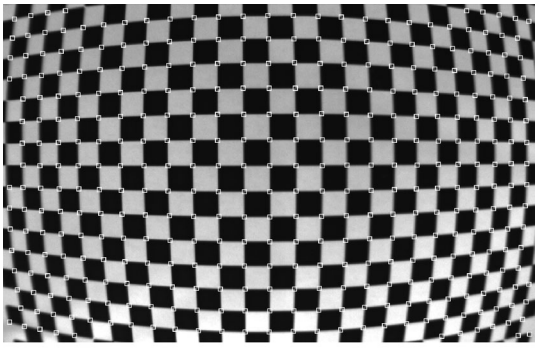
Fig. 5 Forward-mapping from distortion image to correction image

2) Gray-scale interpolation: in this paper, bilinear interpolation is used to reconstruct the correction image.

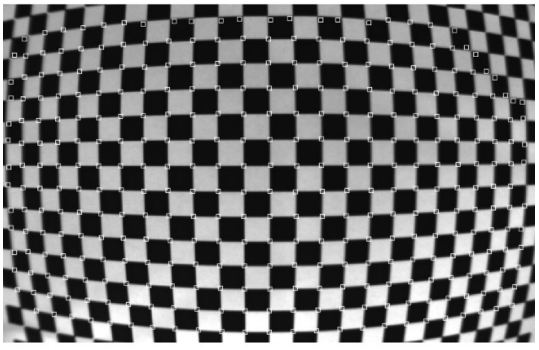
4 Experiments

4.1 Corner points detection algorithm

The performance of the proposed corner detective algorithm is tested by comparing with the chessboard corner detection algorithm in OpenCV. It is shown in Fig. 6 that the proposed method achieves higher detection rate and is more accurate than the OpenCV method.



(a)



(b)

Fig. 6 White rectangles are the detected corners: (a) the proposed method; (b) the OpenCV method

4.2 Synthetic data experiment

The performance of the proposed method is tested in several experiments. First, use the proposed method to estimate parameters of artificial distortion images. We distorted the origin image (Fig. 7) by adding all the three forms of distortion, and then corrected them with the proposed method.

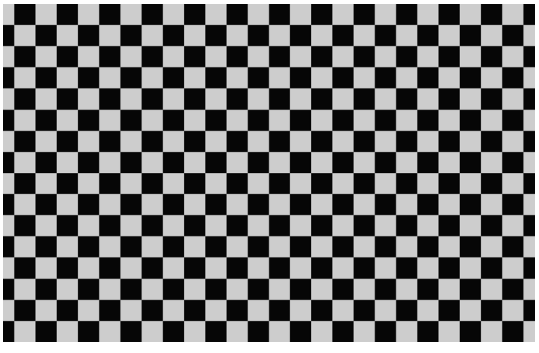


Fig. 7 Origin image without distortion

This method has a good estimation about parameters k_1, p_1, p_2, which are the main factors of distortion. The correction images are very similar to the origin image without distortion, shown in Fig. 8.

Synthetic data are used for further analysis for the performance of the proposed method. The synthetic data are setup as Gao and Yin did in Ref. [8], shown in Fig. 9(a), defining the optical axis as z-axis, the horizon direction on the image plane as x-axis, and vertical direction on the image plane as y-axis.

Figure 10 shows the three methods' performance versus noises. In this experiment, Gaussian noise with zero mean and σ standard deviation is added to distorted points. We test the performance with noise standard deviation σ from 0.1 to 1.5 pixels. Define *ERROR* as root of mean square error in Eq. (13), which is to describe the correction accuracy. For each noise level, we used 100 images to evaluate the method's performance. Brown's method does not correct prism distortion. It indicates that the method without considering the entire distortion model could not make a good image correction.

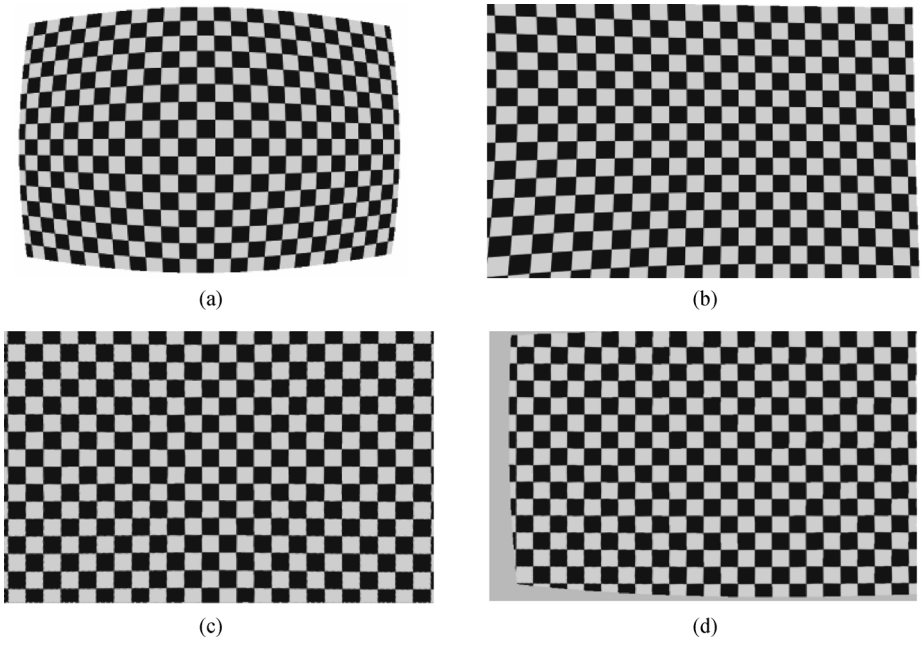


Fig. 8 (a) and (b) are images with different distortion parameters; (c) and (d) are corrected image by proposed method, which parameters are shown in Table 1

Table 1 Compares the distortion between man-made parameters and the results of proposed method

origin distortion image	Figure 8(a)	Figure 8 (b)
distortion model	$k_1 = 3.0\text{e}{-6},\ k_2 = -5.8\text{e}{-13}$ $p_1 = -2.4\text{e}{-5},\ p_2 = 2.86\text{e}{-5}$ $s_1 = 6.5\text{e}{-6},\ s_2 = -7.2\text{e}{-6}$	$k_1 = -2.3\text{e}{-6},\ k_2 = 4.3\text{e}{-13}$ $p_1 = 5.0\text{e}{-5},\ p_2 = -3.4\text{e}{-5}$ $s_1 = 7.8\text{e}{-6},\ s_2 = 5.6\text{e}{-6}$
correct image by proposed method	Figure 8(c)	Figure 8(d)
correct model by proposed method	$k_1 = 3.67\text{e}{-6},\ k_2 = -3.47\text{e}{-12}$ $p_1 = -1.99\text{e}{-5},\ p_2 = 2.36\text{e}{-5}$ $s_1 = -1.26\text{e}{-6},\ s_2 = 2.87\text{e}{-7}$	$k_1 = -2.67\text{e}{-6},\ k_2 = 1.37\text{e}{-13}$ $p_1 = 4.90\text{e}{-5},\ p_2 = -2.79\text{e}{-5}$ $s_1 = 1.48\text{e}{-5},\ s_2 = -6.87\text{e}{-6}$

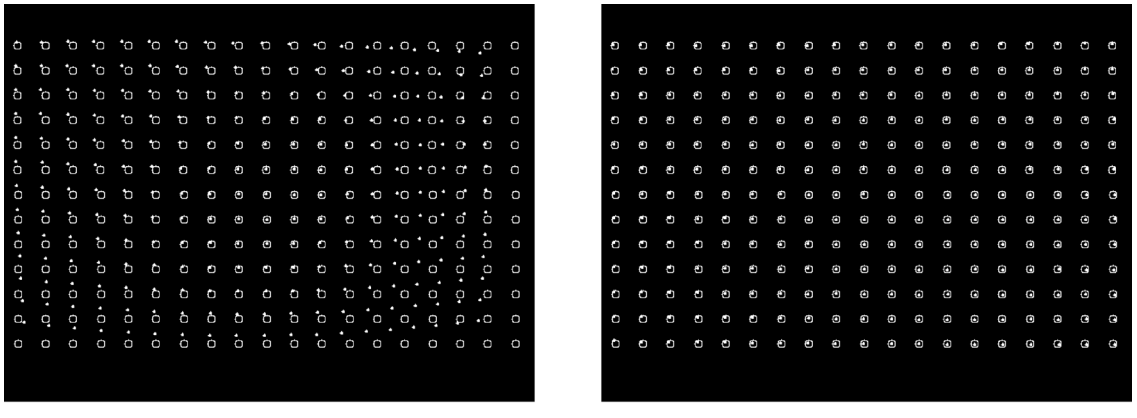


Fig. 9 Performance of the proposed method with the synthetic data, dots are distorted points and circles are non-distorted points. (a) The distorted image; (b) the corrected image by the proposed method

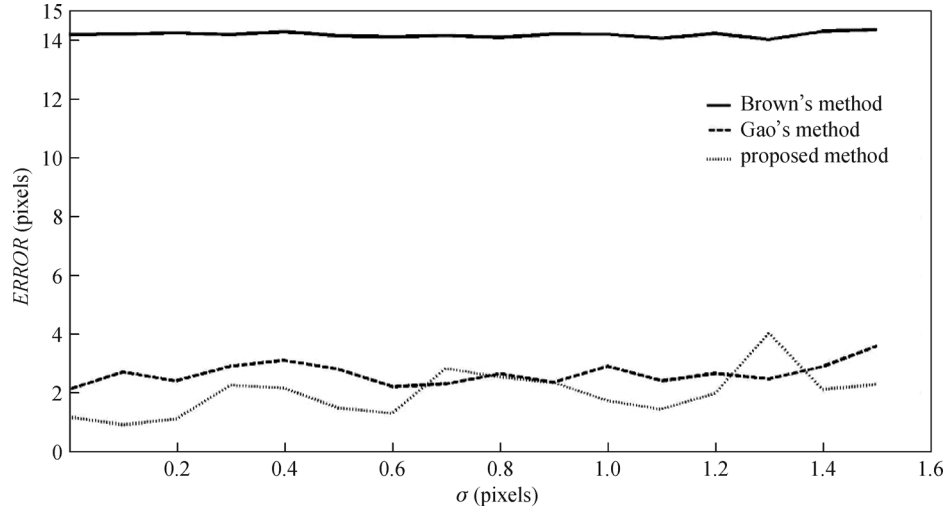


Fig. 10 Compare *ERROR* of the proposed method with Gao's and Brown's method versus different noises levels

$$ERROR = \frac{\sum_i \sqrt{(x_d - x_r)^2 + (y_d - y_r)^2}}{n}, \qquad (13)$$

where n is the number of samples, (x_d,y_d) is test data and (x_r,y_r) is the real data.

To test the robust of the proposed method, the image data are rotated around z-axis or x-axis. The results are shown in Figs. 11 and 12. Figure 11 shows the *ERROR* results when the chessboard is rotated around z-axis. It shows that the correction accuracy is similar between different rotation angles. In Fig. 12, though the rotation around x-axis is even up to 7.5°, the *ERROR* is similar with Gao's method.

The proposed method is very fast to converge. In these experiments, the iteration times are nearly 2 to 4, and the total time consuming to solve distortion model is almost 10 ms (image size is 752 × 480, using PC with CPU of i3 2.5 GHz).

4.3 Real data experiment

Real distortion images are taken by the Pentax CCTV lens (F1.2) with the focal lengths of 4 and 8 mm, which are captured by smart camera based on TMS320DM6437 DSP. Figure 7 is the object for taking photos. Figure 13 shows the pictures with different lens.

Use the proposed method to correct the distortion images in Fig. 13, and results are shown in Fig. 14. It shows that proposed method makes the collinear point in straight pattern in image. Table 2 presents the solutions in detail. We evaluate distortion residual of the distance between point and line. Since the points on each columns and rows are actually collinear, the distortion residual is the

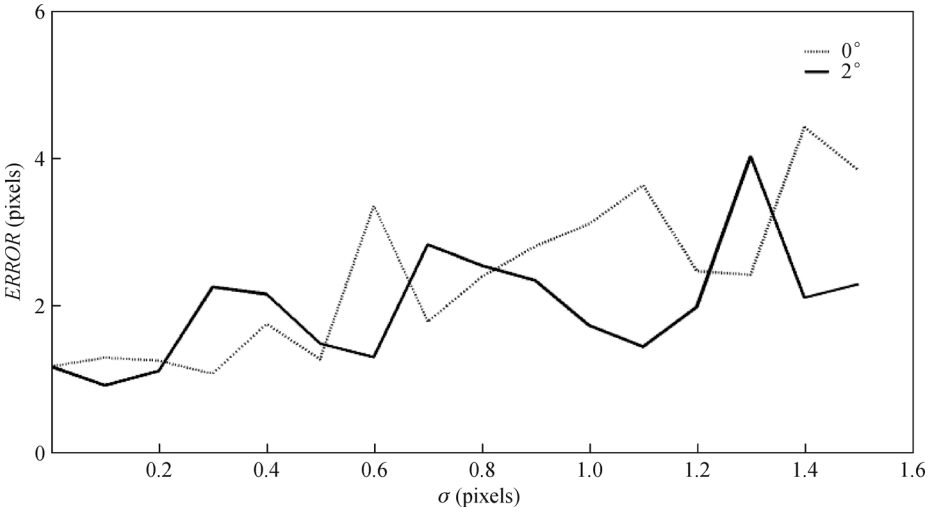


Fig. 11 *ERROR* of different rotations around the z-axis of image versus noises levels

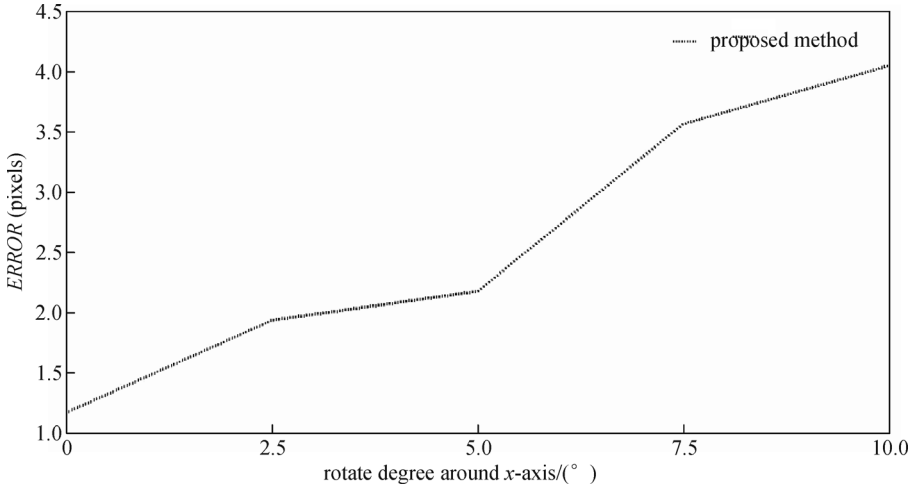


Fig. 12 *ERROR* versus different rotation around the *x*-axis of image

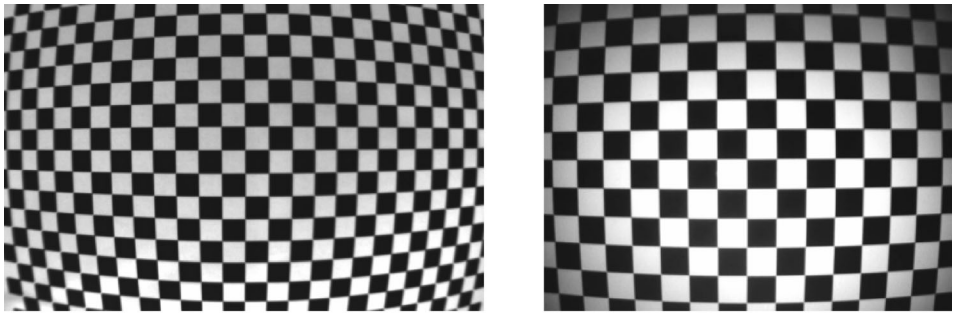


Fig. 13 (a) Pictures taken with 4 mm lens, image size is 752×480; (b) pictures is taken with 8 mm lens, image size is 640×480

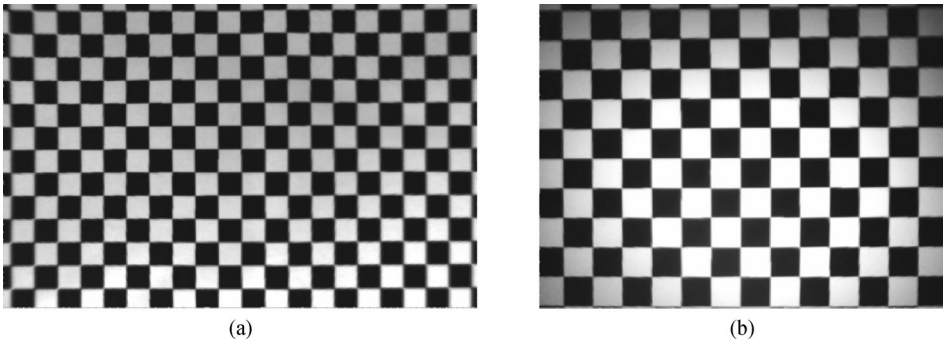


Fig. 14 (a) Corrected image taken with 4 mm lens; (b) corrected image taken with 8 mm lens

average distance between points and line. First, least square method is used to estimate the line function. Second, the absolute distance from points to the line is computed. The average absolute distance of all the point indicates the error of distortion, named as root mean square error (*RMSE*). Maximum distortion rate is calculated by the maximum distance from point to line dividing the image diagonal length.

Table 2 indicates that proposed method is effective, which makes the image with little distortion and even the maximum distortion rather small.

5 Conclusions

This paper proposed a new method for lens distortion

Table 2 Experiment results with proposed method

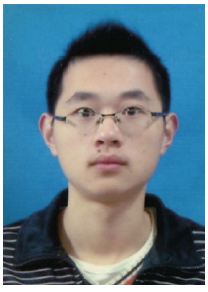
	len 1 focal $f = 4$ mm	len 2 focal $f = 8$ mm
correct model	$k_1 = 8.86\text{e}{-7}$, $k_2 = 3.97\text{e}{-12}$ $p_1 = 1.23\text{e}{-5}$, $p_2 = 5.64\text{e}{-5}$ $s_1 = 1.43\text{e}{-5}$, $s_2 = -2.23\text{e}{-5}$	$k_1 = 7.72\text{e}{-7}$, $k_2 = 1.98\text{e}{-14}$ $p_1 = 1.05\text{e}{-8}$, $p_2 = 2.89\text{e}{-8}$ $s_1 = 1.37\text{e}{-8}$, $s_2 = 2.15\text{e}{-8}$
RMSE	0.34 pixel	0.25 pixel
maximum distortion rate	0.13%	0.10%

correction. Only one chessboard pattern image was used to rectify the three distortion models, which makes the implement convenient. To minimize the influence by the tiny distortion of the center four points, iterative procedure was introduced to optimize the distortion model and reevaluate the non-distorted points. Both synthetic data and real data have been adopted to test by the proposed method. Synthetic data experiment showed that the proposed method had good performance versus noises, and it was more accurate for lens distortion correction by considering the whole distortion models. Besides, chessboard rotation experiment was conducted to show the robustness of this method. The result of real data experiment also indicated that the proposed method was effective.

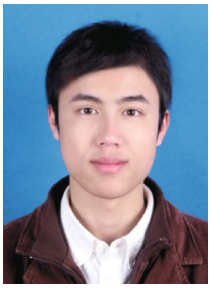
References

1. Abellard A, Bouchouicha M, Ben Khelifa M. A genetic algorithm application to stereo calibration. In: Proceedings of 2005 IEEE International Symposium on Computational Intelligence in Robotics and Automation. Espoo, Finland. 2005, 285–290
2. Brown D C. Close-range camera calibration. Photogrammetric Engineering, 1971, 37(8): 855–866
3. Weng J, Cohen P, Herniou M. Camera calibration with distortion models and accuracy evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(10): 965–980
4. Zhang Z. On the epipolar geometry between two images with lens distortion. In: Proceedings of the 13th International Conference on Pattern Recognition (ICPR), Vienna. 1996, 407–411
5. Du X, Li H, Zhu Y. Camera lens radial distortion correction using two-view projective invariants. Optics Letters, 2011, 36(24): 4734–4736
6. Goljan M, Fridrich J. Estimation of lens distortion correction from single images. In: Proceedings of SPIE 9028, Media Watermarking, Security, and Forensics. 2014, 90280N
7. Prescott B, McLean G. Line-based correction of radial lens distortion. Graphical Models and Image Processing, 1997, 59(1): 39–47
8. Gao D, Yin F. Computing a complete camera lens distortion model by planar homography. Optics & Laser Technology, 2013, 49: 95–107
9. Hartley R, Kang S B. Parameter-free radial distortion correction with center of distortion estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(8): 1309–1321
10. Park J, Byun S C, Lee B U. Lens distortion correction using ideal image coordinates. IEEE Transactions on Consumer Electronics, 2009, 55(3): 987–991
11. Manolis L. Levenberg-Marquardt nonlinear least squares algorithms in C/C ++

Yubin Wu is an associate professor in the School of Optical and Electronic Information, Huazhong University of Science and Technology. He received the M.E. degree in optical engineering from Institute of Optics and Electronics of the Chinese Academy of Sciencesin 1987; the B.E. degree in optical instruments from Huazhong University of Science and Technologyin 1984. His research interests include optoelectronic sensing and signal processing, machine vision, and the development of high-tech products.



Shixiong Jiang is a Ph.D. candidate in the School of Optical and Electronic Information, Huazhong University of Science and Technology. He received the B.E. degree in the School of Optical and Electronic Information, Huazhong University of Science and Technology in 2011. His research interests include machine vision, 3D reconstruction and pattern recognition.



Zhenkun Xu received the B.E. and M.E. degrees in the School of Optical and Electronic Information, Huazhong University of Science and Technology. His research interests include camera calibration and image processing.

Song Zhu is a Ph.D. candidate in the School of Optical and Electronic Information, Huazhong University of Science and Technology. He received the B.E. degree in the School of Optical and Electronic Information, Huazhong University of Science and Technology, in 2010. His research interests include 3D computer vision, image segment and pattern recognition.

Danhua Cao is a professor in the School of Optical and Electronic Information, Huazhong University of Science and Technology. She received the Ph.D. degree in electronic physics and devices from Huazhong University of Science and Technology in 1993; the B.E. degree in measuring and control technology and instrumentations from Huazhong University of Science and Technology in 1987. She is the permanent member of the Professional Committee of Opto-electronic Technology in the Chinese Optical Society. Her research interests include optoelectronic sensing and signal processing, machine vision algorithm and systems.