2012 IEEE International Conference on Systems, Man, and Cybernetics
October 14-17, 2012, COEX, Seoul, Korea

# Fish-Eye Distortion Correction Based on Midpoint Circle Algorithm

Yongle Li, Maojun Zhang, Yu Liu, Zhihui Xiong

College of Information System and Management
National University of Defense Technology
Changsha, China
liyongle1984@gmail.com

***Abstract*—This paper presents a novel embedded real-time fish-eye image distortion correction algorithm with application in IP network camera. A fast and simple distortion correction method is introduced based on Midpoint Circle Algorithm (MCA) which aims to determine the pixel positions along a circle circumference based on incremental calculation of decision parameters. Although only the vertical is rectilinearised, experimental results show that our correction method based on MCA is efficient and effective. In particular, our method can be applied without considering planar checkerboard, iterative fitting of model parameters, complex computation, or traditional lookup tables. Therefore, our algorithm is suitable for embedded camera platform without any extra hardware resources.**

***Keywords-fish-eye image calibration; midpoint circle algorithm; distortion correction***

## I. INTRODUCTION

The wide angle cameras containing fish-eye lens are widely used in many applications, including robot navigation, 3D reconstruction, image-based rendering, and single view metrology, because they can provide a large field of view (FOV) with more visual information from a single image. Therefore, fish-eye distortion correction is a foundational key work along with the uses of fish-eye lens. Some generic calibration models are published in [1, 2, 13]. Wonpil Yu [3] proposed a distortion correction method using second-order radial distortion model for mobile computing applications. Seok-Han Lee et al. [4] discussed a framework to determine distortion vectors and distortion function directly using a planar checkerboard. Furthermore, Ying et al. [10] described a method of calibrating cameras with radial distortion using spherical perspective projection principles. However, since they used orthographic spherical perspective projection, their algorithms can only be applied to orthographic fish-eye cameras other than equidistant fish-eye cameras [12, 19]. There are some other state-of-the-art techniques of camera distortion calibration and correction. Wang et al. [7] described an approach according to which lens distortion is governed by the coefficients of radial distortion and transform from ideal image plane to real sensor array plane. Tardif et al. [8, 11] presented plane-based self-calibration of cameras with radial symmetric distortion given a set of sparse features matched in at least two views. Orekhov et al. [9] put forward an optimized full scale automatic camera calibration approach. Gennery et al. [13] described an approach for accurately calibrating cameras including radial lens distortion by using known points such as those measured from a calibration fixture. Moumen T. El-Melegy et al. [16] presented an automatic approach based on the robust least-median-of-squares estimator. Devernay et al. [17] assumed the presence of straight lines in the scene. Distortion parameters are sought which lead to lines being imaged as straight in the corrected image.

This research was partially supported by National Natural Science Foundation (NSFC) of China under project No.61175006, No.61175015, No. 60803101 and No.60872150; iCORE (Alberta Innovates) Alberta, and NSERC Canada.

Nevertheless, few literatures show the correction methods used in embedded platforms which need simple computation for real-time applications. This paper draws inspiration from Midpoint Circle Algorithm [5] in computer graphics and applies this algorithm to fish-eye distortion correction.

The remainder of this paper is structured in the following way: In Section Ⅱ, we describe the fish-eye image calibration method and estimate the distortion center using circle fitting. Next in Section Ⅲ, a new fish-eye distortion correction approach based on MCA is presented in detail. The experimental results are given in Section Ⅳ, and we draw the conclusion of the proposed work in Section Ⅴ.

## II. FISH-EYE IMAGE CALIBRATION

The purpose of fish-eye image calibration is to estimate the center $(x_c, y_c)$ and radius $r$ of fish-eye image circle. Moreover, the center and radius of fish-eye image are the main parameters in the proposed distortion correction algorithm which is described afterwards. The shape of fish-eye image is almost a circle circumference as described in Strand et al. [14]. The calibration method described in this paper utilizes the circle fitting from [6].

As shown in Fig. 1, Fig. 1(a) is an ideal fish-eye image, of which the center is the center of CCD or CMOS sensors and the radius is the half of image width or height. Fig. 1(b) shows a fish-eye image which is captured by our embedded IP cameras without distortion correction. Here, the fish-eye image edge can be expressed as:

$$(x-x_c)^2+(y-y_c)^2=r^2 \tag{1}$$

In (1), $(x_c, y_c)$ is the center of image and $r$ is the radius of the image circle. In this work, we estimate the center and radius by LMS (least mean square error) [18] using circle fitting as described in Fig. 2, where, the twelve red points are given by manual operation for circle fitting, as shown in Fig. 3.

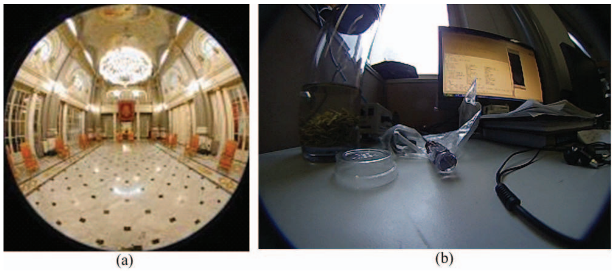


Figure 1. Fish-eye images: (a) ideal fish-eye image; (b) real fish-eye image.

Figure 2. Fish-eye image circle fitting: red points are given by manual operation for circle fitting, and the blue circle is the fitting result.

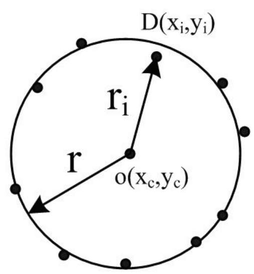


Figure 3. Circle fitting.

And the aim of circle fitting is the error as

$$error = \min \sum_{i=1}^{12} (r_i - r)^2 \tag{2}$$

In (2), $r_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}$ , $(x_i, y_i)$ is the point we choose. Then we can estimate the three parameters $(x_c, y_c, r)$. Fig. 4 shows the calibration result which obtains the center and radius of fish-eye image.

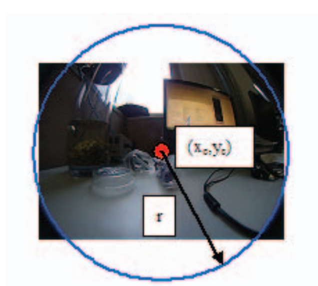


Figure 4. Calibration result of a fish-eye image: the red point $(x_c, y_c)$ is the center and $r$ is the radius of the blue circle which is the fitted image edge.

It should be pointed out that the fish-eye image calibration is implemented by using the video snapshot image captured by embedded IP camera. Then the estimated values of image center and radius will be applied in embedded platforms for distortion correction in real-time.

## III. Distortion Correction

### A. Midpoint Circle Algorithm

Midpoint Circle Algorithm (MCA) is to determine the pixel positions along a circle circumference based on incremental calculation of decision parameters. Compared with the Cartesian equation involving square root calculations and the polar coordinates parametric equations containing trigonometric calculations, MCA involves only simple integer operations [5].

As in the raster line algorithm, we sample at unit intervals and determine the closest pixel position to the specified circle path at each step. For a given radius $r$ and screen center position $(x_c, y_c)$, we can firstly set up our algorithm to calculate pixel positions around a circle path centered at the coordinate origin $(0, 0)$. Then each calculated position $(x, y)$ is moved to its proper screen position by adding $x_c$ to $x$ and $y_c$ to $y$. Along the circle section from $x=0$ to $x=y$ in the first quadrant, the slope of the curve varies from 0 to -1. Therefore, we can take unit steps in the positive $x$ direction over this octant and use a decision parameter to determine which of the two possible pixel positions in any column is vertically closer to the circle path. Positions in other seven octants are then obtained using symmetry.

Also as in Bresenham's line algorithm [5], the midpoint circle method calculates pixel positions along the circumference of a circle using integer additions and subtractions, which assumes that the circle parameters are specified in integer display coordinates. The steps of midpoint circle algorithm are summarized as follows.

*1)* Input radius $r$ and circle center $(x_c, y_c)$, then set the coordinates for the first point on the circumference of a circle centered on the origin as:

$$(x_0, y_0) = (0, r) \tag{3}$$

*2)* Calculate the initial value of the decision parameter as:

$$p_0 = 1 - r \tag{4}$$

*3)* At each $x_k$ position, starting at $k=0$, perform the following test. If $p_k<0$, the next point along the circle centered on $(0, 0)$ is $(x_{k+1}, y_k)$ and

$$p_{k+1} = p_k + 2x_{k+1} + 1 \tag{5}$$

Otherwise, the next point along the circle is $(x_{k+1}, y_{k-1})$ and

$$p_{k+1} = p_k + 2x_{k+1} + 1 - 2y_{k+1} \tag{6}$$

where, $2x_{k+1} = 2x_k + 2$ and $2y_{k+1} = 2y_k - 2$.

*4)* Determine symmetry points in the other seven octants.

*5)* Move each calculated pixel position $(x, y)$ onto the circular path centered at $(x_c, y_c)$ and plot the coordinates values:

$$x = x + x_c,\ \ y = y + y_c \tag{7}$$

*6)* Repeat steps 3 through 5 until $x>=y$.

Fig. 5 shows the sketch map of this algorithm.

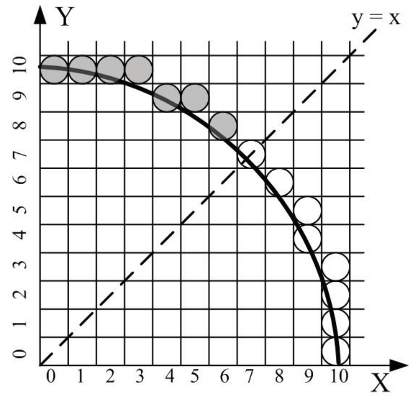


Figure 5. Pixel positions (solid circles) along a circle path centered on the origin and with radius $r=10$, as calculated by the midpoint circle algorithm. Open ("hollow") circles show the symmetry positions in the first quadrant.

## B. Distortion Correction

The proposed fish-eye distortion correction algorithm based on MCA is a real-time approximate correction method, which directly calculates the coordinate mapping between corrected image and fish-eye image. It is also pointed out that this approach is based on and utilizes the fact that a straight line in world space projects to a circle on the fish-eye image plane, as in Strand [14]. If a straight line maps to an elliptical segment in the image plane, e.g. orthographic model, we also can use the fast correction algorithm based on the fast elliptical algorithm as in [5].

Fig. 6 describes the distortion correction sketch map and the image coordinate system is *XOY*.

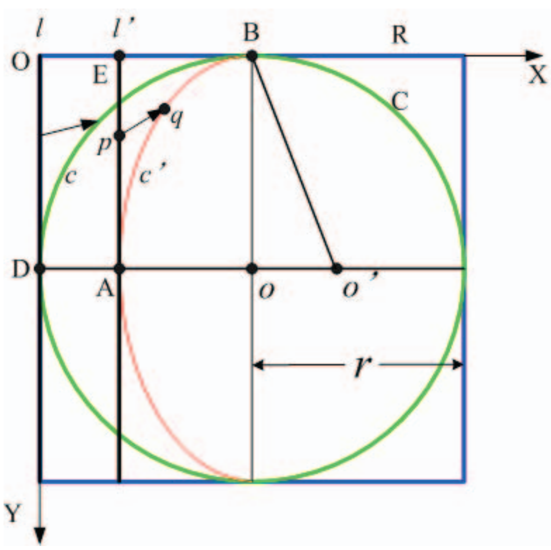


Figure 6. Distortion correction sketch map.

Suppose that $p$ is a random pixel in vertical line $l'$ of the corrected image $R$, and $q$ is a random pixel in circle section $c'$ of fish-eye image $C$. $c'$ is the tangent circle section of $l'$ and $o'$ is the center of $c'$. The main idea of our correction algorithm is identifying the mapping relationship between $p$ and $q$ based on MCA. The steps of distortion correction algorithm are summarized as follows.

*1)* Calculating the corresponding center $o'$ and radius of circle section $AB$. According to (8):

$$|Ao'| = |Bo'| \tag{8}$$

Where $o(x_c, y_c)$, $A(a, y_c)$, $B(x_c, 0)$, $o'(x_{o'}, y_c)$, then

$$x_{o'} = \frac{x_c^2 + y_c^2 - a^2}{2(x_c - a)}\ ,\ |Ao'| = \frac{x_c^2 + y_c^2 - a^2}{2(x_c - a)} - a \tag{9}$$

*2)* Calculating each pixel position in circle section $AB$.

Based on MCA, we can calculate each pixel position $q(x_q, y_q)$ in circle section AB immediately.

*3)* Identifying the mapping relationship between $p$ and $q$. According to (10):

$$\frac{|Ep|}{|AE|} = \frac{|\overline{Bq}|}{|AB|} \tag{10}$$

We can identify each pixel position $p(x_p, y_p)$ in vertical line $AE$ corresponded to $q$ in circle section $AB$.

Lastly, we can calculate each pixel mapping relationship between corrected image region $OBoD$ and fish-eye image region $DBo$, when the vertical line goes from $OD$ to $Bo$ at unit pixel step. Meanwhile, the pixel mapping in other three quadrants can be easily obtained using symmetry.

# IV. Experimental Results

## A. Hardware platform

With the application in video surveillance, we choose the TI TMS320DM368 (http://www.ti.com/product/tms320dm368). This processor has 432-MHz Clock Rate and two video image co-processors (HDVICP, MJCP) engines which can support a range of encode, decode, and video quality operations. It is capable of 1080p and 30fps H.264 video processing, etc.

The image sensor is Aptina Micron MT9P031 which has a 1/2.5-Inch 5Mp Digital Sensor. It has column and row binning modes to improve image quality when resizing and to reduce image size without reducing field-of-view (FOV). It is suitable for the application with High resolution network cameras and wide FOV cameras.

The lens is DAIWON DW1634D [20], which horizontal angle of view is 172.2° and vertical angle of view is 126.2°.

For the application in video surveillance, the distortion correction algorithm based on MCA is embedded into the IP camera which we design by the hardware above.

## B. Correction Results

The experimental results applied the proposed method of ideal fish-eye image is presented in Fig. 7. From the corrected results of Fig. 8 and Fig. 9, the proposed correction algorithm is effective to handle with the indoor and outdoor situation both.

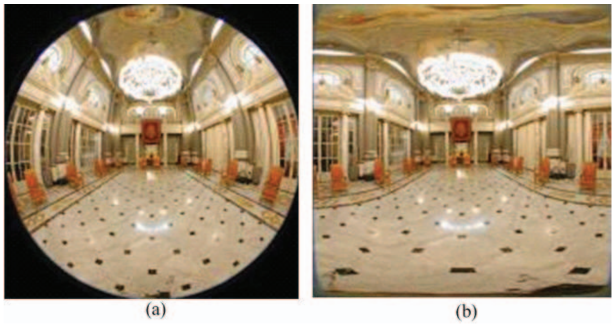

Figure 7. The experimental results of ideal fish-eye image: (a) fish-eye image; (b) corrected result using our proposed method.

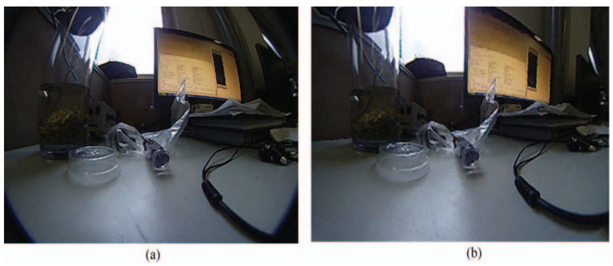

Figure 8. The experimental results of our captured image: (a) indoor fish-eye image; (b) corrected result using our proposed method.

Figure 9. The experimental results of our captured image: (a) outdoor fish-eye image; (b) corrected result using our proposed method.

For the comparison with the previous approach, we illustrate some comparative results in Fig. 10 and Fig. 11. Obviously, the corrected results using our method has smaller distortion in the margin of corrected image compared to the results described in Lee et al. [4]. More importantly, the corrected images using the proposed algorithm can preserve more original image information as full as possible, particularly in horizontal field of view. As shown in Fig. 10 and Fig. 11, the objects marked in the original fish-eye images using red circles exist in our corrected images, but disappear in the corrected results using Lee's [4] approach. This is very useful with application in video surveillance.

Figure 10. Comparative experimental results #1: (a) fish-eye image; (b) corrected result described in Lee [4]; (c) our method.

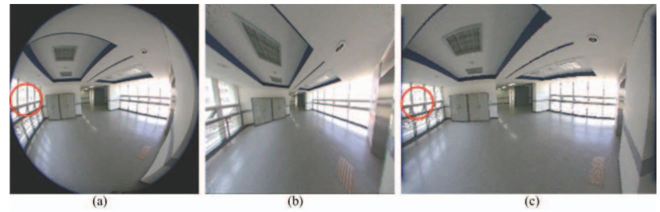

Figure 11. Comparative experimental results #2: (a) fish-eye image; (b) correction result described in Lee [4]; (c) our method.

### C. Computation Complexity

Table I shows that the proposed algorithm which does not need trigonometric and much multiplication calculation has much less computation than equidistant model methods, While other methods are relatively complex to analyze. Table II shows the time consuming and video frame rate in our embedded IP camera. In particular, only the ARM resource in TMS320DM368 can be used and programmed, in which the DSP can not be programmed.

The experimental results show that the proposed correction method can obtain more fine visual quality with less computation in comparison with other methods [4, 15]. It also verifies that the proposed method is effective to handle with embedded IP camera fish-eye distortion problem in real-time.

TABLE I. COMPUTATION COMPLEXITY (TIMES)

| Operators | Equidistant Model | Our Method |
|---|---|---|
| multiplication | 9 | 4 |
| trigonometric | 9 | 0 |
| addition | 12 | 10 |

TABLE II. PERFORMANCE USED OUR METHOD IN EMBEDDED PLATFORM

| Output Resolution | Time Consuming (ms) | Average Frame Rate (fps) |
|---|---|---|
| D1 (640x480) | 15 | 20.2 |
| 720p (1280x720) | 40 | 8.6 |
| 1080p (1920x1080) | 70 | 3.4 |

## V. CONCLUSION

We discussed a novel algorithm to correct fish-eye distortion. Firstly, we calculated the center and radius of the fish-eye image using circle fitting. Then we implemented distortion correction based on MCA. Experimental results show that the proposed algorithm is efficient and effective. More importantly, the corrected images using our proposed algorithm can preserve more original image information compared with other methods. The method can be applied in embedded platform, such as video surveillance in real-time without complex computation, lookup tables or any extra hardware resources. As a future work, we will conduct further study to find simple correction methods in regard to non-symmetric images and to get more comfortable corrected results visually. We believe that the proposed scheme can be employed as a useful approach in the embedded camera-lens related works. We will conduct further study to employ the proposed method for some other real-time imaging systems.

## REFERENCES


[1] Kannala, J., Brandt, S.S., A generic camera model and calibration method for conventional, wide-angle, and fish-eye lenses. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(8), 1335-1340, 2006.

[2] D.Schneider, E.Schwalbe, Validation of geometric models for fisheye lenses, ISPRS Journal of Photogrammetry and Remote Sensing 64, 259-266, 2009.

[3] Wonpil Yu, An Embedded Camera Lens Distortion Correction Method for Mobile Computing Applications, IEEE Transactions on Consumer Electronics, Vol. 49, No. 4, pp. 894-901, Nov. 2003.

[4] Seok-Han Lee, Sang-Keun Lee, Correction of Radial Distortion using a Planar Checkerboard Pattern and its Image, IEEE Transactions on Consumer Electronics, Vol. 55, No. 1, pp. 27-33, Feb. 2009.

[5] Donald Hearn M. Pauline Baker, Computer Graphics with OpenGL, Third Edition, Pearson Education, Inc. and Publishing House of Electronics Industry, pp. 103-109, 2004.

[6] Celestino A. Corral, Claude S. Lindquit, On Implementing Kasa's Circle Fitting Procedure, IEEE Transactions on Instrumentation and Measurement, Vol. 47, No. 3, pp. 789-795, 1998.

[7] Jianhua Wang, Fanhuai Shi, Jing Zhang, and Yuncai Liu, A new calibration model of camera lens distortion, Pattern Recognition, Vol. 41, pp. 607-615, 2008.

[8] Jean-Philippe Tardif, Peter Sturm, S. Roy, Plane-based self-calibration of radial distortion, Proceedings of the Eleventh IEEE International Conference on Computer Vision, 2007.

[9] Vitaliy Orekhov, Besma Abidi, Chris Broaddus, Mongi Abidi, Universal Camera Calibration with Automatic Distortion Model Selection, ICIP 2007, pp. 397-400.

[10] Xianghua Ying, Zhanyi Hu, Hongbin Zha, Fisheye Lenses Calibration Using Straight-Line Spherical Perspective Projection Constraint, ACCV 2006, pp. 61-70.

[11] Jean-Philippe Tardif, Peter Sturm, S. Roy, Self-calibration of a general radially symmetric distortion model, ECCV Vol. 4, 2006, pp. 186-199.

[12] C. Hughes, Robert McFeely, P. Denny, M. Glavin, E. Jones, Equidistant fish-eye perspective with application in distortion center estimation, Image and Vision Computing, Vol. 28, pp. 538-551, 2010.

[13] Donald B. Gennery, Generalized Camera Calibration Including Fish-Eye Lenses, International Journal of Computer Vision 68(3), pp. 239-266, 2006.

[14] Rickard Strand, Eric Hayman, Correcting radial distortion by circle fitting, Proceedings of the British Machine Vision Conference 2005.

[15] Richard I. Hartley, Sing Bing Kang, Parameter-free Radial Distortion Correction with Center of Distortion Estimation, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (8), pp. 1309-1321, 2007.

[16] Moumen T. El-Melegy, Aly A. Farag, Nonmetric Lens Distortion Calibration: Closed-form Solutions, Robust Estimation and Model Selection, Proceedings of the Ninth IEEE International Conference on Computer Vision 2003, 2-Volume Set.

[17] F. Devernay, Olivier Faugeras, Straight lines have to be straight, Machine Vision and Applications, 13(1), pp. 14-24, 2001.

[18] F. Schaffalitzky, A. Zisserman, Planar grouping for automatic detection of vanishing lines and points, Image and Vision Computing 18(9), 2000, pp. 647-658.

[19] C. Hughes, P. Denny, M. Glavin and E. Jones, " Equidistant Fish-Eye Calibration and Rectification by Vanishing Point Extraction", IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(12), pp. 2289-2296, December 2010

[20] http://www.dwopt.com/