



Sub-pixel edge detection based on an improved moment

Feipeng Da^{*}, Hu Zhang

School of Automation, Southeast University, Nanjing 210096, China

ARTICLE INFO

Article history:

Received 7 January 2009

Received in revised form 19 March 2010

Accepted 16 May 2010

Keywords:

Edge detection

Sub-pixel edge location

Vision measuring

ABSTRACT

A novel moment-based method for sub-pixel edge location is proposed. Based on coarse edge-location by SOBEL operator, the geometric information of the target is used to reduce the number of moment-template to only one, which can largely save the time. Experimental results demonstrate that the proposed method is effective with more robustness, higher precision and speed.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

Edge is the basic character of image, and locating the edge fast and accurately takes important part in image processing and vision measuring. There are three methods in sub-pixel edge location: fitting-based method, interpolating-based method and moment-based method. Fitting-based method attempts to obtain sub-pixel edge location by fitting gray-scales of the image according to the given edge model [1–3]. It is high-accurate but time-consuming. Interpolating-based method performs the sub-pixel edge location by interpolating the image data in terms of gray-distribution in actual images [4–6]. It is time-saving but sensitive to noises. Since the moment is the integral operator which is non-sensitive to noises, the moment-based method became a widely used method.

Tabatabai and Mitchell firstly put the moment-based method into the sub-pixel edge location [7]. They obtained the final exact edges by computing four parameters of the edge model. After that, different methods based on Spatial-Gray Moment (SGM) were proposed [8–10]. Additionally, Cheng and Wu put the moment based method in color images [11]. As these methods based on SGM needed six moment-templates for convolution, efficiencies of the above moment-based methods were not so satisfactory. Lee et al. introduced an improved method with three moment-templates based on SGM. The grayscale of target and that of background were computed by bimodal property of histogram [12]. Once the two of four parameters in edge model were computed, only three moment-templates were needed. Ghosal proposed a method based on Zernike Orthogonal Moment (ZOM) which has orthogonality and rotation invariance for sub-pixel edge location [13]. The efficiency of ZOM method is higher than SGM

method since it only needs three moment-templates for sub-pixel edge location. Recently, many authors have improved the moment-based methods of sub-pixel edge location. Bin et al. proposed a sub-pixel edge location method based on Orthogonal Fourier–Mellin Moment (OFMM) which has a better depicted ability for small shaped objects [14]. It pointed out that precision of OFMM method is higher than ZOM method. Li et al. took the amplification of moment-templates into account and selected more proper thresholds [15]. Qu et al. and Hu et al. proposed to coarsely locate edge points by SOBEL operator before using moment-based method in order to reduce computations [16,17].

In all the above moment-based methods, the number of moment-templates used in convolutions is not less than three, and therefore they may be computationally expensive. To improve the drawbacks of moment-based methods, a novel moment method for sub-pixel edge location is proposed in this paper. This method includes edge-location by using ROI and SOBEL, and most important it reduces the number of moment-template to be only one, which can largely save the computational time.

The paper will be discussed as following. A new method is introduced in Section 2. Experimental results and discussion are presented in Section 3 and conclusion is given in Section 4.

2. A novel moment-based method for sub-pixel edge location

In this section, a novel moment-based method for sub-pixel edge location is introduced. This method can be applied to locate the sub-pixel edge of objects which can be described by geometric languages. In the paper, the example of the object is the circular target which is one of representative geometric objects. For other geometric objects, the method can be slightly adjusted to its geometric parameters.

^{*} Corresponding author. Tel.: +86 25 83794974; Fax: +86 25 83793000.

E-mail address: dafp@seu.edu.cn (F. Da).

2.1. Preliminary steps

2.1.1. Initial solution for geometric parameters

For the original image, some basic image-processing steps, which include image filtering, image binary, boundary extraction and impurities removing, should be done. In these steps, the utilized filtering method in the paper is adaptive smoothing filtering, which could eliminate noises as well as maintain the original features of edges [18]. In addition, morphology method for the boundary extraction is taken [19]. After getting the boundary, the Least-Square method is used to fit these boundary points and the initial geometric parameters of objects can be obtained [20].

2.1.2. ROI (region of interest) extraction and edge detection

Usually, the geometric objects to be exacted are only one small part of the whole image. If we exact a ROI which is an area containing the geometric objects, the number of pixels participated in computations is largely reduced. This could make the algorithm much more effective and time-saving. In this paper, the geometric parameters give the solution to ROI-extraction. According to the geometric parameters of the object, one small area containing this object could be selected. This area is the ROI. The whole process for ROI-extraction is completely without any manual operations. Take circular targets as an example. The selected ROI is a square area whose center is the circular target's center, and the length of its side is the sum of circular target's diameter and a margin. The margin is necessary to prevent the influence of transition area, which is about 3–5 pixels.

Usually, the boundary points which are obtained by the morphology method are not quite accurate, and therefore SOBEL operator is utilized for edge detection in this paper as it has high speed and ability to smooth noises. Results of ROI extracting and edge detection are shown in Fig. 1.

The geometric parameters of objects are obtained by fitting the edge points detected by SOBEL operator using the Least-Square method.

2.2. Solution for parameters of the edge model and sub-pixel edge location

The edge of a continuous ideal image can be taken as a step-model, which is shown in Fig. 2, where h_1 is the background grayscale, h_2 is the object grayscale, l is the normalized distance between actual edge and the center, and θ is the angle between the edge and horizontal, which is satisfied as $\theta \in (-\frac{\pi}{2}, \frac{\pi}{2})$.

Furthermore, S_1 and S_2 respectively represent the area of target and background in the edge model. From the analyses of three moment-based methods in Section 1, it can be concluded that three moments of zero order are the same to each other. Here, we choose M_{00} as the mark for the moment of zero order.

$$M_{00} = \iint f(x, y) dx dy = S_1 h_1 + S_2 h_2 \quad (1)$$

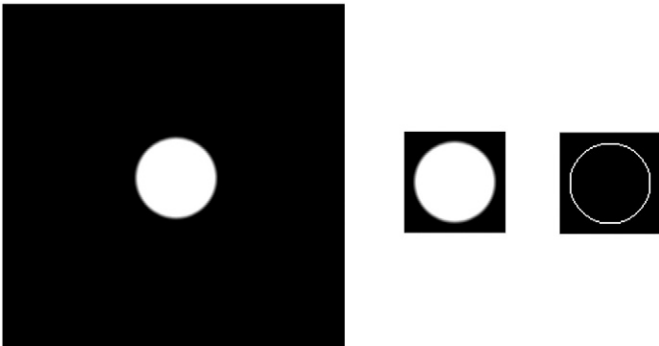


Fig. 1. ROI extraction and edge detection.

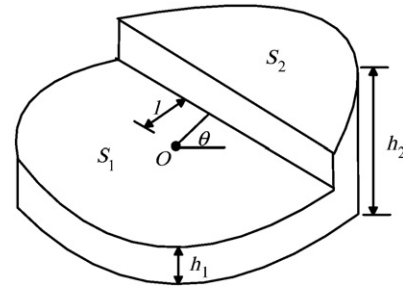


Fig. 2. 2D edge model.

Given the assumption that the edge model should be a unit circle, we have

$$M_{00} = (\pi - S_2)h_1 + S_2h_2 \quad (2)$$

2.2.1. Solution for θ

After rotating the edge model, the edge is vertical (along axis Y), so we have

$$\iint f(x, y) y dx dy = 0 \quad (3)$$

The left part of above equation could be converted to the expression which contains the moments, their combinations and θ . As a result of using the digitized template of moment, the accuracy of the results will be affected by some factors, which may include digitization, gray-scale sampling, the transition region of the edge and the number of pixels which participate in calculation of convolutions.

In this paper, the parameter θ is simply and easily calculated with the principle that gradient-orientation of the edge point is the orientation of normal line which just passes through it. Since the geometric parameters of the object are already calculated, the slope of the normal line could be obtained. Now θ is the inverse tangent of the slope. For a circular target, the orientation of the gradient can be solved in light of position of the center.

$$\theta = \tan^{-1} \frac{y_p - y_c}{x_p - x_c} \quad (4)$$

where (x_p, y_p) is the coordinate of the edge point, (x_c, y_c) is the coordinate of the center which can be obtained by geometric parameter of the circular target.

More widely, for quadratic curve $G(x, y)$,

$$G(x, y) = x^2 + Axy + By^2 + Cx + Dy + E = 0$$

the slope of the gradient is

$$k = \frac{\frac{\partial G(x, y)}{\partial y}}{\frac{\partial G(x, y)}{\partial x}} = \frac{Ax_p + 2By_p + D}{2x_p + Ay_p + C} \quad (5)$$

Apparently, the final precision of θ is influenced by initial geometric parameters. This will be discussed in Section 4.

2.2.2. Solution for h_1 and h_2

A method for separately solution to the gray-scale of target and background is presented in [12]. It utilizes the histogram of the whole image to choose the values of bimodal as h_1 and h_2 . There are some drawbacks. First, it is time-consuming since it needs to add up the histogram and compare several times. Secondly, it is restricted to the cases for which the histogram of the image follows a bimodal

distribution. Third, the result is not accurate if there are several targets in the image in the interruption of uneven illuminations.

In this paper, according to the gray-scale distribution of the geometric target, a line or connecting line is selected in ROI, and the average of gray-scale of pixels on the chosen line or connecting line is the gray-scale of background or target. Obviously, the chosen line or connecting line is crucial. Taking the circular target for example, we choose the line or connecting line as follows.

In Fig. 3, the whole square area is the extracted ROI, and the area in the circle denotes circular target, r is the radius of the circle, the two bold square lines are the chosen lines whose centers are the initial center of the circular target. Moreover, as there is a transitional area in a practical image when choosing the lines, it is necessary to take into account some allowances, which are usually taken as 3 pixels, shown as c_1, c_2 in Fig. 3. From the geometric knowledge, half side-lengths of bold squares are shown below.

$$\begin{cases} l_1 = \frac{\sqrt{2}}{2}r - c_1 \\ l_2 = r + c_2 \end{cases} \quad (6)$$

Avoiding from noises in practical, pixels in the selected line or connecting line should be sorted first. After eliminating one part of front and final pixels, the average of remaining pixels is taken as the gray-scale of target h_1 or the gray-scale of background h_2 .

In addition, when the edge model is moving on edge, for the circular target, the gray-scale of target and background in the left part of circular target is opposite to that in the right part, the final results of h_1 and h_2 need to be exchanged in left part or right part.

For ellipse-shaped target, solution for h_1 and h_2 is similar. For the line-shaped-edge target, the gray-scale exchanging issue does not hold.

2.2.3. Solution for l and sub-pixel location of edge

The ideal edge model should be digitized to $N_c \times N_c$ image. N_c is the dimension of the template, and it is usually chosen as 3, 5, or 7. The computation of moment in the digitized image is that of convolution between the template and gray-scale of pixels. Since only M_{00} is needed to be solved in our new method, only one template is needed.

Usually, the edge model is normalized as a unit-circle. Then, M_{00} changes as

$$M_{00} = \iint_{x^2 + y^2 \leq 1} f_m(x, y) dx dy \quad (7)$$

where $f_m(x, y)$ is the gray-value of the pixel whose coordinate is (x, y) .

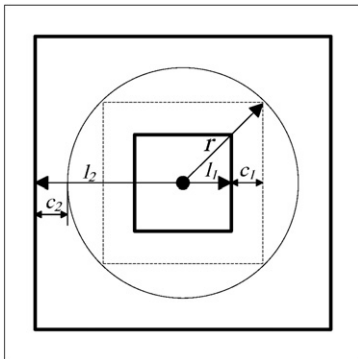


Fig. 3. Schematic diagram for grayscale of target and background.

After discretization of $N_c \times N_c$,

$$M_{00} = \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \iint_{\Omega_{ij}} h_m(x, y) f_m(x, y) dx dy \quad (8)$$

where Ω_{ij} is the pixel at row j and col i , $h_m(x, y) = \begin{cases} 1, x^2 + y^2 \leq 1 \\ 0, x^2 + y^2 > 1 \end{cases}$. Since $f_m(x, y)$ is constant, M_{00} changes as

$$M_{00} = \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} (f_m(i, j)) \cdot \iint_{\Omega_{ij}} h_m(x, y) dx dy. \quad (9)$$

Let $a_{ij} = \iint_{\Omega_{ij}} h_m(x, y) dx dy$, and the integral template of M_{00} is $\{a_{ij}|i, j = 1 \dots N_c\}$.

The $3 \times 3, 5 \times 5, 7 \times 7$ templates of M_{00} are as follows.

The value of M_{00} is the convolution of gray-scale and template. According to Eq. (2), S_2 can be solved.

Rotate the edge model clockwise by θ . The first moment after rotation is

$$M'_{00} = M_{00} \quad (10)$$

The vertical view of the edge model after rotating is described in Fig. 4. l is the distance between the center and the true edge. The shaded part denotes the area of background in the edge model. From the knowledge in geometry, we have

$$S_2 = \beta - \frac{1}{2} \sin(2\beta) \quad (11)$$

$$l = \cos \beta \quad (12)$$

From Eqs. (11) and (12), l can be easily obtained. The final coordinate of sub-pixel edge is shown as

$$\begin{cases} x_{\text{subpixel}} = x_{\text{pixel}} + l \cos(\theta) \\ y_{\text{subpixel}} = y_{\text{pixel}} + l \sin(\theta) \end{cases} \quad (13)$$

where $(x_{\text{subpixel}}, y_{\text{subpixel}})$ is the true sub-pixel coordinate of edge, $(x_{\text{pixel}}, y_{\text{pixel}})$ is the coordinate of the center in unit circle, which is the coordinate of edge point detected by SOBEL operator in Section 2.1.2. Taking the amplification of template into account [15], Eq. (13) is changed as

$$\begin{cases} x_{\text{subpixel}} = x_{\text{pixel}} + \frac{N_c}{2} l \cos(\theta) \\ y_{\text{subpixel}} = y_{\text{pixel}} + \frac{N_c}{2} l \sin(\theta) \end{cases} \quad (14)$$

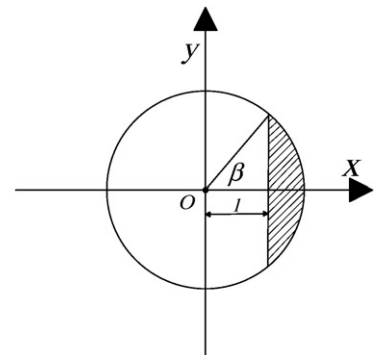


Fig. 4. Vertical view of rotated edge model.

2.3. Algorithm outline

1. Solve the geometric parameters as shown in [Section 2.1.1](#).
2. Exact ROI and detect edge using Sobel operator as shown in [Section 2.1.2](#).
3. Generate the moment template of zero order as shown in [Tables 1.1–1.3](#).
4. Compute the angle of the corresponding edge point as described in [Section 2.2.1](#).
5. Compute the gray-scale of target and background in the image as described in [Section 2.2.2](#).
6. For each edge point detected by Sobel operator:
 - 6.1. Compute the value of the moment of zero order by convoluting the template and grayscale of edge point.
 - 6.2. According to Eqs. (2), (11), and (12), compute the edge parameter l .
 - 6.3. According to Eq. (14), compute the position of final sub-pixel edge.

3. Experimental results and discussion

3.1. Analyses of three other moment-based methods for sub-pixel edge location

The common moment-based methods for sub-pixel edge location are Gray-Moment (GM) method, SGM method, ZOM method and OFMM method. Since the SGM method has already proved to be better than GM method [9], we only discuss the latter three methods.

For two-dimension continuous function $f_m(x, y)$, the SGM of $p + q$ order is defined as [9]

$$M_{pq} = \iint x^p y^q f_m(x, y) dx dy \quad (15)$$

For digitized image, it is defined as

$$M_{pq} = \sum \sum x^p y^q f_m(x, y) \quad (16)$$

To simplify the analysis and the calculation, the edge model is rotated by θ . The SGM after rotation is

$$M'_{pq} = \sum_{r=0}^p \sum_{s=0}^q \binom{p}{r} \binom{q}{s} (-1)^{q-s} (\cos\theta)^{p-r+s} (\sin\theta)^{q+r-s} M_{p+q-r-s, r+s} \quad (17)$$

The computations for parameters of edge model are shown as

$$\begin{cases} \theta = \tan^{-1}(M_{01}/M_{10}) \\ l = (4M'_{20} - M'_{00}) / (3M'_{10}) \\ k = (3M'_{10}) / (2\sqrt{(1-l^2)^3}) \\ h = [2M'_{00} - k(\pi - 2\sin^{-1}l - 2l(\sqrt{1-l^2}))] / (2\pi) \end{cases} \quad (18)$$

where k is the edge contrast which equals to the difference between h_1 and h_2 . The values of SGM can be calculated by convolution between the SGM-template and grayscale of the image pixel.

For two-dimension continuous function $f_m(x, y)$, the ZOM of $p + q$ order is defined as [12]

$$Z_{pq} = \frac{p+1}{\pi} \iint f(x, y) V_{pq}(\rho, \theta) dx dy \quad (19)$$

After rotating angle θ , ZOM is

$$Z'_{pq} = Z_{pq} \cdot \exp(j\theta) \quad (20)$$

Computations for parameters of edge model are shown as follows.

$$\begin{cases} \theta = \tan^{-1}(\text{Im}(Z_{11}) / \text{Re}(Z_{11})) \\ l = \frac{Z_{20}}{Z'_{11}} \\ k = (3Z'_{11}) / (2\sqrt{(1-l^2)^3}) \\ h = [Z'_{00} - k(\pi / 2 - \sin^{-1}l - l(\sqrt{1-l^2}))] / \pi \end{cases} \quad (21)$$

Table 1.1
3 × 3 M_{00} template.

0.2424	0.4319	0.2424
0.4319	0.4444	0.4319
0.2424	0.4319	0.2424

Table 1.2
5 × 5 M_{00} template.

0.0219	0.1231	0.1573	0.1231	0.0219
0.1231	0.1600	0.1600	0.1600	0.1231
0.1573	0.1600	0.1600	0.1600	0.1573
0.1231	0.1600	0.1600	0.1600	0.1231
0.0219	0.1231	0.1573	0.1231	0.0219

For two-dimension continuous function $f_m(x, y)$, the OFMM of $p + q$ order is defined as [13]

$$\phi_{pq} = \frac{p+1}{\pi} \sum_{s=0}^p \alpha_{ps} \int_{-\infty}^{\infty} f(x, y) (x + jy)^{\frac{s-p}{2}} (x - jy)^{\frac{s+p}{2}} dx dy \quad (22)$$

After rotating angle θ , OFMM is

$$\phi'_{00} = \phi_{00}, \phi'_{10} = \phi_{10}, \phi'_{20} = \phi_{20}, \phi'_{01} = \phi_{01} e^{j\theta}, \phi'_{11} = \phi_{11} e^{j\theta} \quad (23)$$

Computations for parameters of edge model are shown as follows.

$$\begin{cases} \theta = \tan^{-1} \left(-\frac{\text{Im}(2(\phi_{01} + \phi_{11}))}{\text{Re}(2(\phi_{01} + \phi_{11}))} \right) \\ h = [\phi'_{00} - k(\sin^{-1}(\sqrt{1-l^2}) - l(\sqrt{1-l^2}))] / \pi \\ k = (2\phi'_{01} + \phi'_{11}) / (2\sqrt{(1-l^2)^3}) \\ l = \frac{3}{5} \left(\frac{4\phi_{10} + \phi_{20}}{2\phi'_{01} + \phi'_{11}} \right) \end{cases} \quad (24)$$

According to Eqs. (15), (19) and (22) and the definitions of SGM, ZOM and OFMM, we have

$$\begin{cases} \theta_M = \tan^{-1}(M_{01} / M_{10}) \\ \theta_Z = \tan^{-1}(\text{Im}(Z_{11}) / \text{Re}(Z_{11})) = \tan^{-1}(M_{01} / M_{10}) \\ \theta_\phi = -\tan^{-1} \left(\frac{\text{Im}(2(\phi_{01} + \phi_{11}))}{\text{Re}(2(\phi_{01} + \phi_{11}))} \right) = -\tan^{-1}(-M_{01} / M_{10}) \\ \quad = \tan^{-1}(M_{01} / M_{10}) \end{cases} \quad (25)$$

From Eq. (25), it can be easily found that the parameter θ calculated by SGM, ZOM and OFMM is the same.

As mentioned above, parameter θ of three methods is the same, so θ_M , θ_Z and θ_ϕ in Eq. (25) are all labeled as θ . Computations of k and h in the three methods are as follows.

$$\begin{cases} k_M = \frac{3M'_{10}}{2\sqrt{(1-l_M^2)^3}} = \frac{3(\cos\theta M_{10} + \sin\theta M_{01})}{2\sqrt{(1-l_M^2)^3}} \\ k_Z = \frac{3Z'_{11}}{2\sqrt{(1-l_Z^2)^3}} = \frac{3(\cos\theta \text{Re}(Z_{11}) + \sin\theta \text{Im}(Z_{11}))}{2\sqrt{(1-l_Z^2)^3}} = \frac{3(\cos\theta M_{10} + \sin\theta M_{01})}{2\sqrt{(1-l_M^2)^3}} \\ k_\phi = \frac{2\phi'_{01} + \phi'_{11}}{2\sqrt{(1-l^2)^3}} = \frac{(2\phi_{01} + \phi_{11})(\cos\theta + \sin\theta j)}{2\sqrt{(1-l^2)^3}} = \frac{3(M_{10} - M_{01}j)(\cos\theta + \sin\theta j)}{2\sqrt{(1-l^2)^3}} \\ \quad = \frac{3[M_{10} \cos\theta + M_{01} \sin\theta + (M_{10} \sin\theta - M_{01} \cos\theta)j]}{2\sqrt{(1-l^2)^3}} = \frac{3[M_{10} \cos\theta + M_{01} \sin\theta]}{2\sqrt{(1-l^2)^3}} \end{cases} \quad (26)$$

Table 1.3
7 × 7 M_{00} template.

0	0.0287	0.0686	0.0807	0.0686	0.0287	0
0.0287	0.0815	0.0816	0.0816	0.0816	0.0815	0.0287
0.0686	0.0816	0.0816	0.0816	0.0816	0.0816	0.0686
0.0807	0.0816	0.0816	0.0816	0.0816	0.0816	0.0807
0.0686	0.0816	0.0816	0.0816	0.0816	0.0816	0.0686
0.0287	0.0815	0.0816	0.0816	0.0816	0.0815	0.0287
0	0.0287	0.0686	0.0807	0.0686	0.0287	0

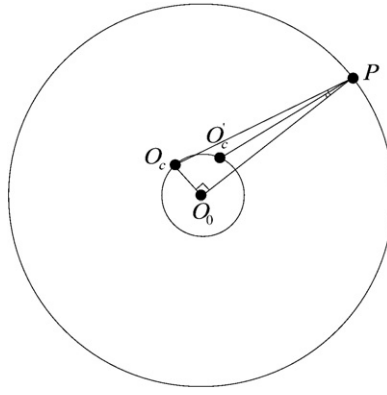


Fig. 5. Schematic diagram of the angle deviation.

$$\begin{cases} h_M = \left[2M'_{00} - k(\pi - 2 \sin^{-1} l - 2l(\sqrt{1-l^2})) \right] / (2\pi) = \left[M_{00} - k(\pi/2 - \sin^{-1} l - l(\sqrt{1-l^2})) \right] / \pi \\ h_Z = \left[Z'_{00} - k(\pi/2 - \sin^{-1} l - l(\sqrt{1-l^2})) \right] / \pi = \left[M_{00} - k(\pi/2 - \sin^{-1} l - l(\sqrt{1-l^2})) \right] / \pi \\ h_\phi = \left[\phi'_{00} - k(\sin^{-1}(\sqrt{1-l^2}) - l(\sqrt{1-l^2})) \right] / \pi = \left[M_{00} - k(\pi/2 - \sin^{-1} l - l(\sqrt{1-l^2})) \right] / \pi \end{cases} \quad (27)$$

It can be seen that differences of k and h calculated by SGM, ZOM and OFMM are differences of the parameter l . Computations for l in the three methods are as follows.

$$\begin{cases} l_M = \frac{4M'_{20} - M'_{00}}{3M'_{10}} = \frac{4(\cos^2 \theta M_{20} + 2 \cos \theta \sin \theta M_{11} + \sin^2 \theta M_{02}) - M_{00}}{3(\cos \theta M_{10} + \sin \theta M_{01})} \\ l_Z = \frac{Z'_{20}}{Z'_{11}} = \frac{2M_{20} + 2M_{02} - M_{00}}{\cos \theta M_{10} + \sin \theta M_{01}} \\ l_\phi = \frac{3}{5} \left(\frac{4\phi'_{10} + \phi'_{20}}{2\phi'_{01} + \phi'_{11}} \right) = \frac{3}{5} \cdot \frac{5}{3} \cdot \left(\frac{-M_{00} + 2M_{02} + 2M_{20}}{(M_{10} - M_{01}j)(\cos \theta + \sin \theta)} \right) \\ = \frac{-M_{00} + 2M_{02} + 2M_{20}}{(M_{10} \cos \theta + M_{01} \sin \theta + (M_{10} \sin \theta - M_{01} \cos \theta)j)} \\ = \frac{2M_{20} + 2M_{02} - M_{00}}{\cos \theta M_{10} + \sin \theta M_{01}} \end{cases} \quad (28)$$

From Eq. (28), the parameter l calculated by ZOM and OFMM is the same, while the difference of the parameter l calculated by SGM and ZOM is

$$\Delta l = l_M - l_Z = \frac{4(\cos^2 \theta M_{20} + 2 \cos \theta \sin \theta M_{11} + \sin^2 \theta M_{02}) - M_{00}}{3(\cos \theta M_{10} + \sin \theta M_{01})} - \frac{2M_{20} + 2M_{02} - M_{00}}{\cos \theta M_{10} + \sin \theta M_{01}} \quad (29)$$

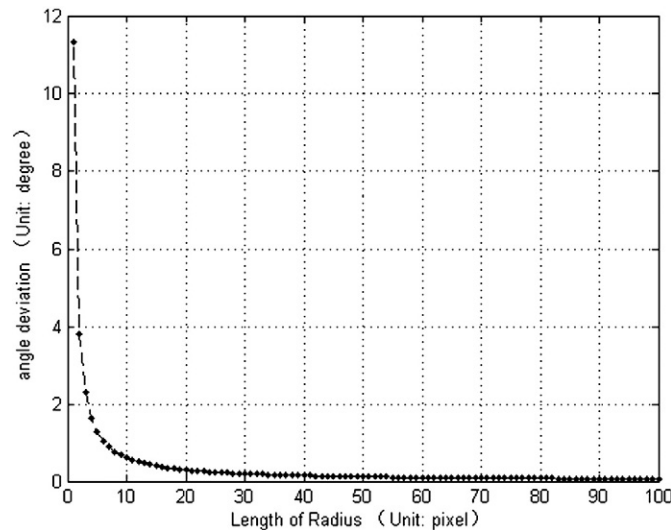


Fig. 6. Angle deviation VS length of radius.

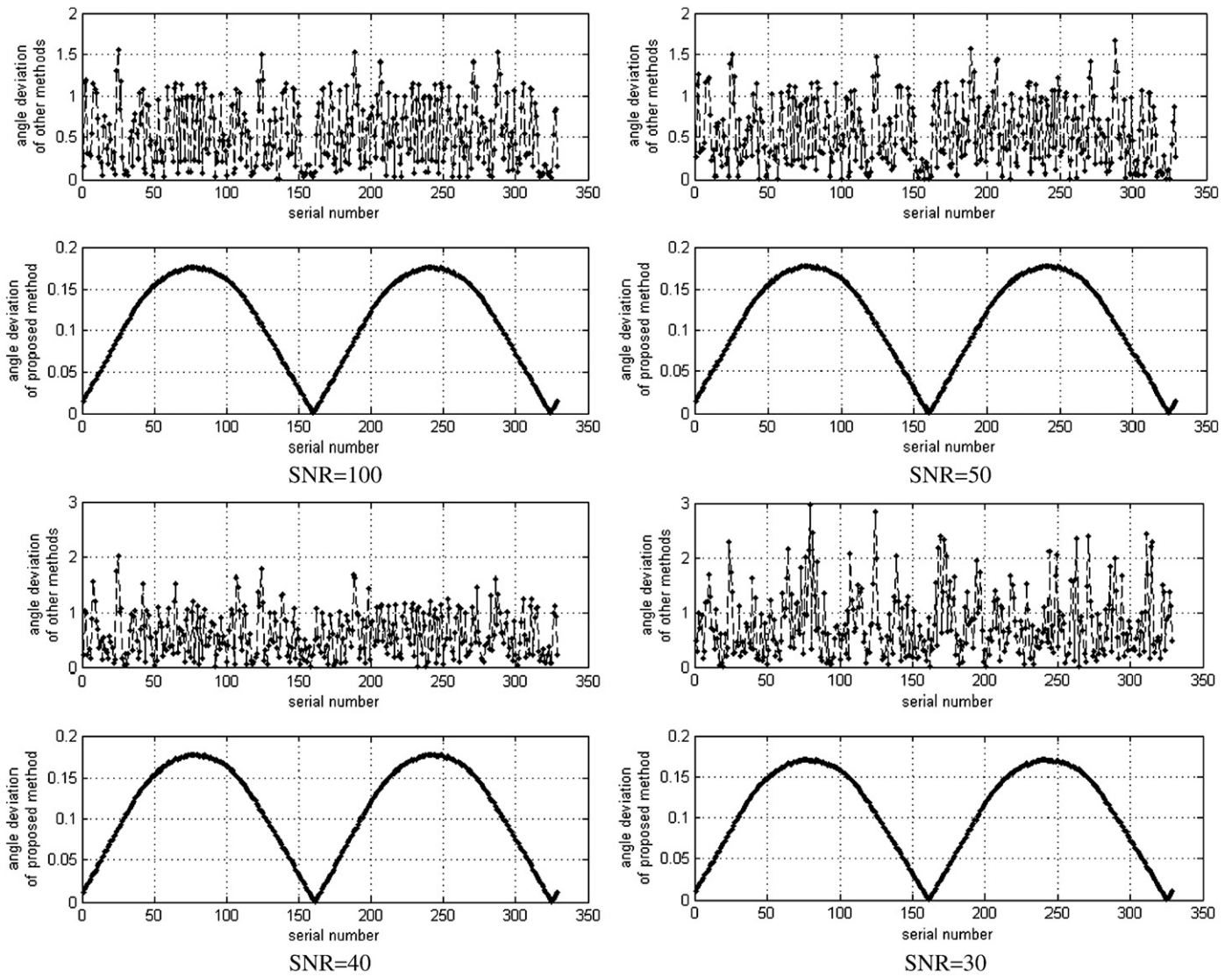


Fig. 7. Angle deviations in proposed method and other methods with different SNR.

From the edge model, the final edge position is depended on parameter θ and parameter l . And we have already concluded that these three methods get the same parameter θ , the accuracy of parameter l accordingly depends entirely on the final accuracy of the edge position which will be discussed later.

Table 2.1

Effect of white Gaussian noise of circular target (unit: 0.01 pixel, SNR: signal to noise ratio).

Algorithm	SNR = 100 dB	SNR = 50 dB		SNR = 40 dB		SNR = 30 dB	
	RMS and MAX	RMS	MAX	RMS	MAX	RMS	MAX
SGM	8.78	8.79	9.22	8.92	9.68	9.36	10.63
ZOM	8.77	8.78	9.14	8.92	9.58	10.12	11.85
OFMM	8.77	8.78	9.14	8.92	9.58	10.12	11.85
Proposed method	2.59	2.60	2.71	2.67	2.94	3.33	4.25

Table 2.2

Effect of white salt and pepper noise of circular target (unit: 0.01 pixel, ND: noise density).

Algorithm	ND = 0	ND = 0.002	ND = 0.004	ND = 0.006	ND = 0.008
SGM	8.78	8.90	9.32	10.07	12.48
ZOM/OFMM	8.77	8.82	9.26	10.34	12.74
Proposed method	2.59	2.64	2.97	3.59	4.86

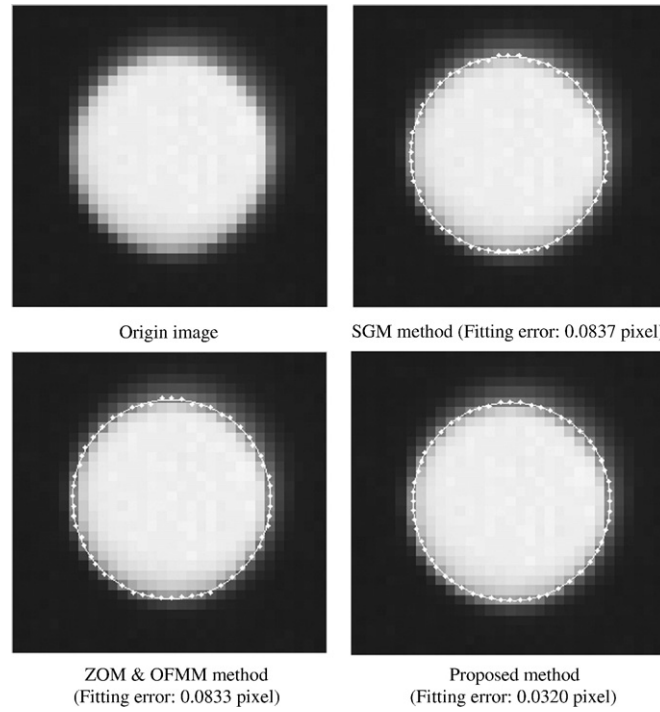


Fig. 8. Origin image and results of edge location.

3.2. Analysis of accuracy of the angle in the edge model

3.2.1. Analysis of the angle in the proposed method

Due to space limitations, circular target is taken as the representative geometric target. In the proposed method, the initial center determines the angle in the edge model as

$$\theta_c = \tan^{-1} \frac{y_c - y}{x_c - x} \quad (30)$$

Let the ideal coordinate of center be (x_0, y_0) , then the ideal coordinate of the angle in the edge model is

$$\theta_0 = \tan^{-1} \frac{y_0 - y}{x_0 - x} \quad (31)$$

The deviation between the ideal angle and the computed angle is

$$\Delta\theta = |\theta_c - \theta_0| \quad (32)$$

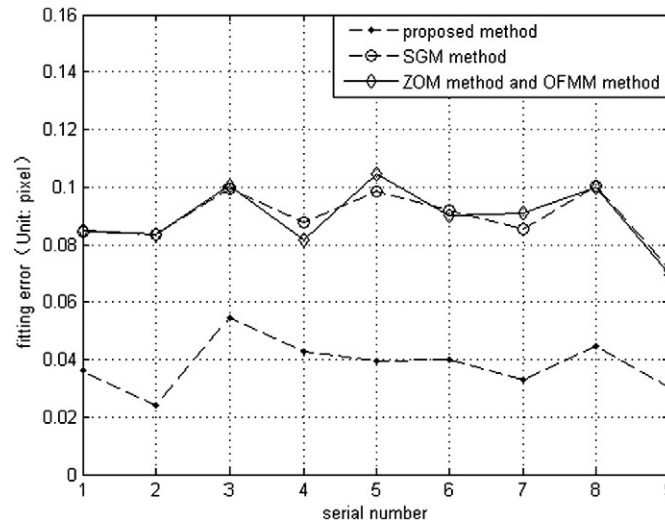


Fig. 9. Fitting error of edge location for circular target.

Table 3.1

Effect of white Gaussian noise of ellipse-shaped target (unit: 0.01 pixel, SNR: signal to noise ratio).

Algorithm	100 dB	50 dB		40 dB		30 dB	
	RMS and MAX	RMS	MAX	RMS	MAX	RMS	MAX
SGM	8.64	8.69	8.79	8.72	8.88	8.91	9.35
ZOM	8.90	8.93	9.04	8.97	9.26	9.81	10.60
OFMM	8.90	8.93	9.04	8.97	9.26	9.81	10.60
Proposed method	3.05	3.08	3.15	3.10	3.27	4.19	4.78

Table 3.2

Effect of white salt and pepper noise of ellipse-shaped target (unit: 0.01 pixel, SNR: signal to noise ratio).

Algorithm	ND = 0	ND = 0.002	ND = 0.004	ND = 0.006	ND = 0.008
SGM	8.64	8.78	8.96	9.57	11.21
ZOM/OFMM	8.90	9.12	9.87	10.73	12.38
Proposed method	3.05	3.18	3.49	3.85	5.01

Fig. 5 is the schematic diagram of the angle deviation. O_0 is the ideal center, O_c is the initial computed center, P is the edge point. Let the maxima distance from O_c to O_0 be ρ . With the geometric knowledge, when the maxima angle deviation occurs, O_c moves around the circle whose center is O_0 and whose radius is ρ . At this time, $\overline{O_c O_0}$ is perpendicular to $\overline{O_0 P}$. According to the diagram, we have

$$\tan(\Delta\theta_{\max}) = \frac{\rho}{O_0 P} \quad (33)$$

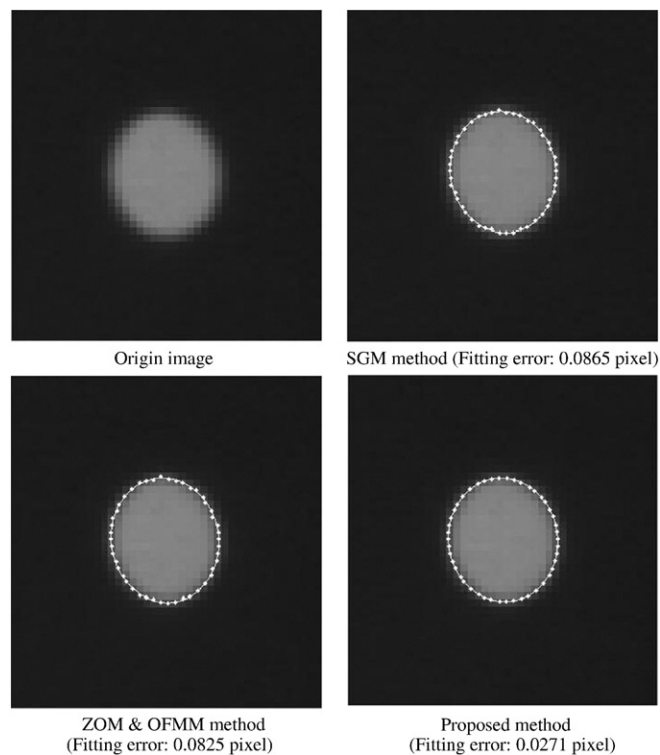
Considering the sampling influence of the grayscale of pixel, we have

$$O_0 P = (R \pm 0.5)\text{pixel} \quad (34)$$

According to Eqs. (33) and (34),

$$\Delta\theta_{\max} = \tan^{-1}\left(\frac{\rho}{R-0.5}\right), \quad (35)$$

where R denotes the length of radius of the circular target.

**Fig. 10.** Origin image and results of edge location.

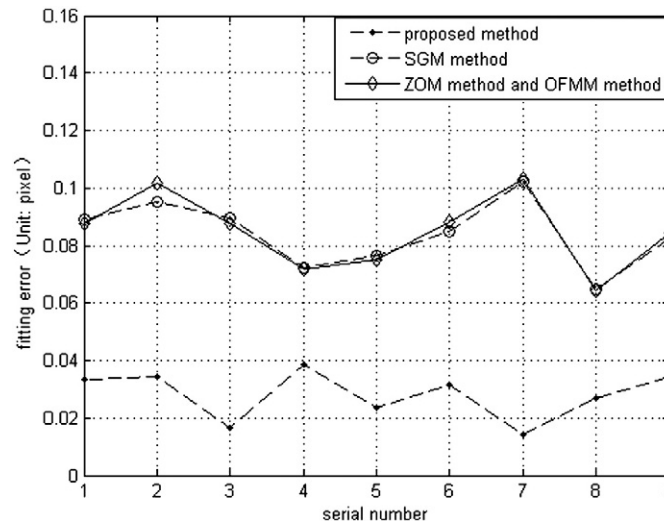


Fig. 11. Fitting error of edge location for circular target.

Give the assumption that SNR is equal to 20, the accuracy of the center, which is calculated by Least-Square fitting for edges observed by SOBEL operator, is more than 0.1 pixel [20]. This gives

$$\rho \leq 0.1 \tag{36}$$

In the worst case, we take $\rho = 0.1$. Then the Eq. (35) is changed as:

$$\Delta\theta_{\max} = \tan^{-1} \left(\frac{0.1}{R-0.5} \right) \tag{37}$$

From the Eq. (37), $\Delta\theta_{\max}$ is closely related with radius R . Fig. 6 is the relation of R and $\Delta\theta_{\max}$.

Form Fig. 6, it can be concluded that $\Delta\theta_{\max}$ decreases with the increase of radius R . When R is larger than 10 pixel, $\Delta\theta_{\max}$ is less than 0.6031° . Moreover, the radius of circular target is usually larger than 10 pixel in practical. The above result is solved under several worst cases, and the probability to meet the cases is quite low. Therefore, $\Delta\theta$ would be smaller and accepted.

3.2.2. Comparisons of angle between the proposed method and the three other methods

As presented above, angles calculated by the three moment-based methods are all the same. Simulated images with noises are processed by the proposed method and other moment-based methods. The true value of angle in the simulated image is known. Results of angle deviation are shown in Fig. 7. SNR is defined as

$$SNR = 20 \log_{10} \frac{k}{\sigma} \text{ dB} \tag{38}$$

where k is the contrast of the image, σ is the standard deviation of the added random noise. From the Fig. 7, deviations of angle computed in new method are much smaller than that computed in other methods. It can be concluded that angles computed in new method are better.

Table 4.1
Effect of white Gaussian noise of line-shaped-edge target (unit: 0.01 pixel, SNR: signal to noise ratio).

Algorithm	100 dB	50 dB		40 dB		30 dB	
	RMS and MAX	RMS	MAX	RMS	MAX	RMS	MAX
SGM	8.40	8.44	8.51	8.38	8.53	8.98	9.54
ZOM	8.46	8.49	8.56	8.53	8.62	10.33	11.33
OFMM	8.46	8.49	8.56	8.53	8.62	10.33	11.33
Proposed method	4.16	4.17	4.24	4.14	4.29	4.28	4.74

Table 4.2
Effect of white salt and pepper noise of line-shaped-edge target (unit: 0.01 pixel, SNR: signal to noise ratio).

Algorithm	ND = 0	ND = 0.002	ND = 0.004	ND = 0.006	ND = 0.008
SGM	8.40	8.52	8.89	9.79	10.95
ZOM/OFMM	8.46	8.59	8.73	10.02	11.79
Proposed method	4.16	4.21	4.49	4.89	6.01

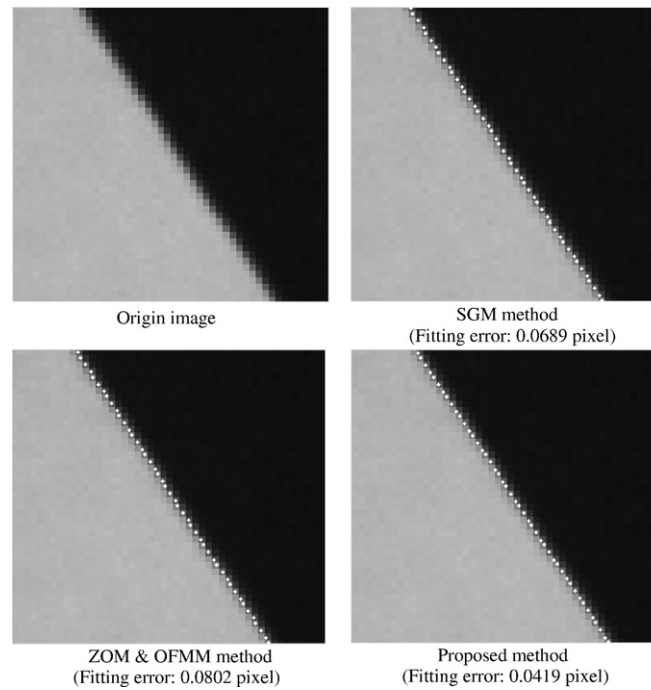


Fig. 12. Origin image and results of edge location.

3.3. Analyses of final location of sub-pixel edge by proposed method

In order to evaluate algorithms accurately and effectively, in the following experiments, dimensions of the templates with all kinds of moments are equal to 5 and the fitting error are chosen as the evaluation criteria.

3.3.1. Circular target

3.3.1.1. Simulation experiment. An ideal image which contains a circular target is first generated. It should be smoothed by 3×3 smoothing operators considering the smoothing function of camera lens. Then, the image after adding white Gaussian noises is separately processed by SGM, ZOM, OFMM and the proposed method. Results are shown in Table 2.1. We also test the effect of white salt and pepper noise. The results are shown in Table 2.2.

From the Tables 2.1 and 2.2, we can find that, the error of proposed method is slightly lower than other methods, which illustrates that the proposed method has higher accuracy and better robustness to noise. Moreover, it can be easily found that, results of ZOM and OFMM are identity, which is the same with the conclusion in Section 3.1.

3.3.1.2. Practical Experiment. A practical image is processed by SGM, ZOM, OFMM and the proposed method. The origin image and results are shown in Fig. 8. Then, nine images are processed by four methods to compute the sub-pixel edge considering the stability of the algorithm. Results are shown in Fig. 9. The unit is pixel.

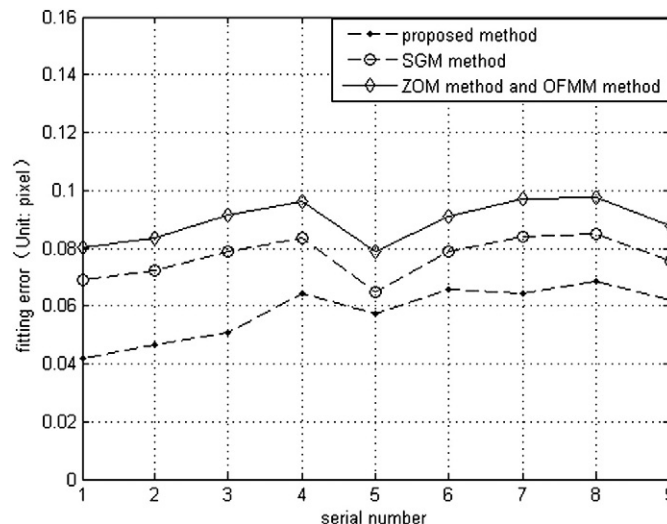


Fig. 13. Fitting error of edge location for circular target.

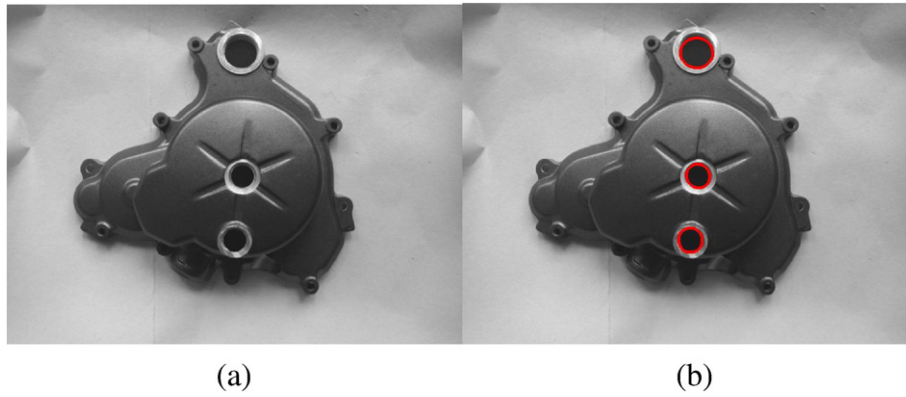


Fig. 14. (a) Testing image. (b) Result of the proposed method.

From results in Figs. 8 and 9, it can be easily found that the proposed method is superior to other three methods in precision. This means that the edge parameters computed in the proposed method is more accurate. Moreover, comparing the results in simulated experiment and practical experiment, the fitting error in practical experiment is larger than the one in simulated experiment. The reasons for this phenomenon are: Firstly, the precision of the sub-pixel edge location method based on the moment descends with the increase of the width in transitional area. Secondly, the width of transitional area in the practical image is larger than the one in the simulated image.

3.3.2. Ellipse-shaped target

3.3.2.1. Simulation experiment. Similar to the simulated experiment in Section 3.3.1.1, results by four methods are shown in Tables 3.1 and 3.2. From the tables, the precision and robustness to noise of the proposed method performs better than other three methods.

3.3.2.2. Practical experiment. A practical image shown in Fig. 10 is processed by four methods, and results are shown in Fig. 10. Then, nine practical images are located by four methods. Results are shown in Fig. 11.

From the results in above practical experiments, it can be seen that the fitting error in new method is less than that in others, which illustrates that the new method also has better precision for ellipse target.

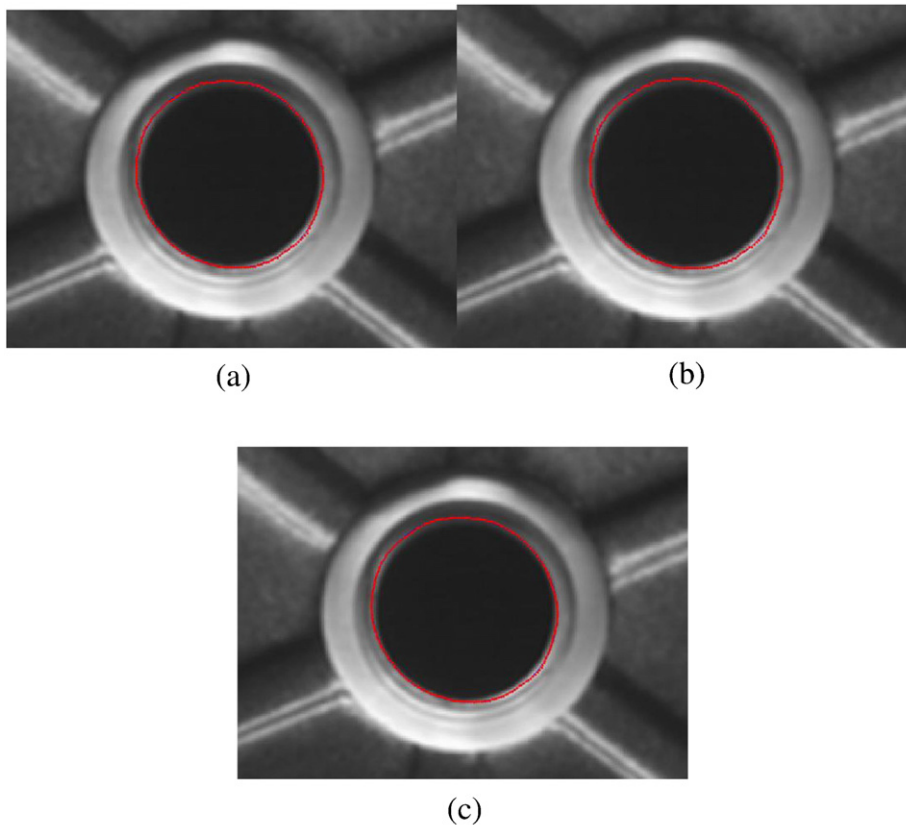


Fig. 15. (a) SGM method (fitting error: 0.2543 pixel). (b) OFMM/ZOM method (fitting error: 0.2691 pixel). (c) The proposed method (fitting error: 0.2088 pixel).

Table 5
Comparisons of run-time.

Method	Addition operator	Multiply operator
SGM (5×5 template)	$25 \times 6 \times N_p$	$25 \times 6 \times N_p$
ZOM (5×5 template)	$25 \times 3 \times N_p$	$25 \times 3 \times N_p$
OFMM (5×5 template)	$25 \times 6 \times N_p$	$25 \times 6 \times N_p$
Proposed method (5×5 template)	$25 \times 1 \times N_p$	$25 \times 1 \times N_p$

3.3.3. Line-shaped-edge target

3.3.3.1. Simulated experiment. Similar to above simulated experiments, results of locating edges of a simulated image whose edge is line-shaped by four methods are shown in Tables 4.1 and 4.2. From the table, we can find the proposed method has better precision and robustness to noise for line-shaped-edge object.

3.3.3.2. Practical experiment. A practical image shown in Fig. 12 is processed by four methods, and results are shown in Fig. 12. Then, nine images are processed by four methods, and results are shown in Fig. 13.

From results in practical experiments, it can be concluded that the proposed method performs better for line-shaped-edge target.

3.3.4. Complex object

We test the image shown in Fig. 14(a) with the proposed method. The result is shown in Fig. 14(b).

According to the influence of the illuminations, the middle circle in the three circles is chosen as the comparable object. Since the tested targets are three standard round orifices, the fitting error is chosen as the estimated error. The location results by SGM, OFMM/ZOM and the proposed method are shown in Fig. 15(a), (b) and (c), respectively. The fitting errors of results by SGM, OFMM/ZOM and the proposed method are: 0.2543 pixel, 0.2691 pixel and 0.2088 pixel, respectively. Obviously, the proposed method has better accuracy.

3.4. Time analyses and comparisons

3.4.1. Analyses of preliminary steps

Preliminary steps are necessary and important for saving computing time. First, the initial geometric parameters could be utilized to extract special geometric objects and perform the object-recognition. Second, extracting ROI makes the following image processes only deal with pixels not in the whole image but in ROI, which largely decreases the number of pixels which participate in computing. Third, SOBEL operator is used to locate the edge coarsely, which makes that pixels taking part in moment-template convolution could be edge points observed by SOBEL operators. This may improve the efficiency.

3.4.2. Time analyses of sub-pixel-edge location step and comparisons

In the proposed method, time spent for computing could be reduced by following some guidelines. The first step is the solution for gray-scale of target and background. Since the solution is simple and fast, moreover, it processes only once for every geometric target, this step is quite time-saving. The second step is the solution for the angle of the edge model. This step includes only one equation solution. Obviously, time spent for computing in this step is negligible. The last step is the solution for nonlinear equation described by Eq. (11). A look-up table method is utilized for solving the equation, which reduces the time cost. Therefore, time spent for convolution between the moment template and gray-scale of special pixels in the image could be taken as the total running time in sub-pixel-edge location step.

Comparisons of running time in convolution for the new method and other methods (SGM, ZOM and OFMM) are shown in Table 5. N_p is the number of coarsely located edge points. From the table, it can be seen that frequencies of addition and product in proposed method are less than those of others, which illustrates that the new method performs more efficiently.

4. Conclusion

A new method for moment-based sub-pixel edge location is proposed after analyzing the three common moment-based methods. This method utilizes initial geometric parameters to solve the edge model parameters, and it needs only one moment-template to perform the edge location with high-precision. From results in experiments, the proposed method is not only more accurate, but also much more time-saving. Moreover, the new method could be easily applied in vision measuring, and it could also be important for solving the geometric parameters of objects and some situations using edge points such as calibrating, matching and so on.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grant 60775025 and the Natural Science Foundation of Jiangsu Province BK2007116.

References

- [1] V.S. Nalwa, T.O. Binford, On detecting edges, IEEE Transaction on Pattern Analysis and Machine Intelligence 8 (6) (1986) 699–714.
- [2] J. Ye, G.K. Fu, U.P. Poudel, High-accuracy edge detection with blurred edge model, Image and Vision Computing 23 (5) (2005) 453–467.
- [3] F. Bouchara, S. Ramdani, Subpixel edge refinement using deformable models, Optical Society of America 26 (4) (2009) 820–832.
- [4] J.W. Hwang, H.S. Lee, Adaptive image interpolation based on local gradient features, IEEE Signal Processing Letters 11 (3) (2004) 359–362.
- [5] L. Zhang, X.L. Wu, An edge-guided image interpolation algorithm via directional filtering and data fusion, IEEE Transactions on Image Processing 15 (8) (2006) 2226–2238.
- [6] T. Hermosilla, E. Bermejo, A. Balaguer, L.A. Ruiz, Non-linear fourth-order image interpolation for subpixel edge detection and localization, Image and Vision Computing 26 (9) (2008) 1240–1248.
- [7] A.J. Tabatabai, O.R. Mitchell, Edge location to subpixel values in digital imagery, IEEE Transactions on Pattern Analysis and Machine Intelligence 6 (2) (1984) 188–201.
- [8] E.P. Lyvers, O.R. Mitchell, Precision edge contrast and orientation estimation, IEEE Transactions on Pattern Analysis and Machine Intelligence 10 (6) (1988) 927–937.
- [9] E.P. Lyvers, O.R. Mitchell, M.L. Akey, A.P. Reeves, Subpixel measurements using a moment-based edge operator, IEEE Transactions on Pattern Analysis and Machine Intelligence 11 (12) (1989) 1293–1309.

- [10] S. Cheng, T. Wu, Subpixel edge detection of color images by principal axis analysis and moment-preserving principle, *Pattern Recognition* 38 (2005) 527–537.
- [11] Y. Shan, G.W. Boon, Sub-pixel localization of edges with non-uniform blurring: a finite closed-form approach, *Image and Vision Computing* 18 (2000) 1015–1023.
- [12] C.K. Lee, W.C. So, A fast edge detector with subpixel accuracy, *Proceedings of IEEE Record of the 1992 International Power Electronics and Motion Control Conference on Industrial Electronics Control, Instrumentation and Automation*, 1992, pp. 710–715.
- [13] S. Ghosal, R. Mehrotra, Orthogonal moment operators for subpixel edge detection, *Pattern Recognition* 26 (2) (1993) 295–306.
- [14] T.J. Bin, A. Lei, J.W. Cui, W.J. Kang, D.D. Liu, Subpixel edge location based on orthogonal Fourier–Mellin moments, *Image and Vision Computing* 26 (4) (2008) 563–569.
- [15] J.Q. Li, J.W. Wang, S.B. Chen, L. Wu, Improved algorithm of subpixel edge detection using Zernike orthogonal moments, *Optical Technique* 29 (4) (2003) 500–504.
- [16] Y.D. Qu, C.S. Cui, S.B. Chen, J.Q. Li, A fast subpixel edge detection method using Sobel–Zernike moments operator, *Image and Vision Computing* 23 (1) (2005) 11–17.
- [17] Z.F. Hu, H.S. Dang, X.R. Li, A novel fast subpixel edge location method based on Sobel–OFMM, *Proceedings of the IEEE International Conference on Automation and Logistics*, 2008, pp. 828–832.
- [18] R.C. Gonzalez, R.E. Woods, *Digital Image Processing*, Prentice-Hall, Englewood Cliffs, NJ, 2002, pp. 121–123.
- [19] L.Z. Xia, J.X. Li, *Digital Image Processing*, Southeast University Press, China, 2005, pp. 234–236.
- [20] S.J. Ahn, W. Rauh, H.J. Warnecke, Least-squares orthogonal distances fitting of circle, sphere, ellipse, hyperbola, and parabola, *Pattern Recognition* 34 (12) (2001) 2283–2303.