Contributed paper

OPTO-ELECTRONICS REVIEW **13**(1), 61–67

A randomized algorithm for detecting multiple ellipses based on least square approach

L. LI*, Z. FENG, and K. HE

Institute of System Engineering, Xi'an Jiaotong University
Xi'an, ShanXi, 710049, P.R. China

In this paper, a randomized method for detecting multiple ellipses based on the least square approach is presented. The main concept used is that we first randomly select three edge pixels in the image, which are the centre of three windows with the same size. In order to determine a possible ellipse, we use the least square method to fit all the edge points in these three window, and to solve the ellipse parameters through Lagrange multiplier method. Then we randomly select the fourth edge pixel in the image and define a distance criterion to determine whether there is a possible ellipse in the image. After finding a possible ellipse, we apply a further verification process to determine whether the possible ellipse is a true ellipse or not. Some artificial images with different levels of noises and some natural grey images containing circular objects with some occluded ellipses and missing edges have been taken to test the performance. Experimental results demonstrate that the proposed algorithm is faster and more accurate than other methods.

Keywords: circular feature, ellipse, detection, randomized, least square.

1. Introduction

Successful detection of multiple intersecting or occluded geometric shapes such as ellipses in a digital image is an important task in pattern recognition and computer vision. Ellipse features are not only the basic elements in nature but also very common shapes in many man-made objects, which have been commonly used in robot vision fields. Circular feature is a particular case of conic feature, because its perspective projection in any arbitrary orientation is always an exact ellipse. Efficient recognition of ellipses from digital images is important for locating objects in many vision-based fields. For example, circular features have been widely used in robot vision for accurate self-locating with circular landmarks and football tracking in robot soccer competition [1,2].

A variety of approaches have been suggested for detecting the ellipse and estimating the related parameters. The Hough transform [3] is a standard method for detecting curves that may be easily parameterized, such as lines, circles and ellipses. It consists of the following steps. Firstly, a pixel in the image is mapped to a curve in some parameterized space. Secondly, the parameters of valid curve are binned into an accumulator where the number of curves in a bin equals its score, and at last, a curve with a maximum score is selected from the accumulator to represent a curve in the image. Since defining an ellipse requires five parameters, the Hough transform needs 5D accumulator array over parameter space. So, this method tends to occupy a large amount of memory, has low speed, as well as it is difficult and ambiguous to find multiple local maxima of the corresponding 5D histogram, which leads to low accuracy and even incorrect solutions.

An efficient randomized algorithm is presented for detecting circles [4]. The main concept used is that it first randomly selects four edge pixels in the image and defines a distance criterion to determine whether there is a possible circle in the image; after finding a possible circle, apply an evidence-collecting process to further determine whether the possible circle is a true circle or not. This algorithm cannot detect ellipses although it can detect multiple circles efficiently.

An improved ellipse detection method is using randomized Hough transform (RHT) [5–7], whose basic principle is a stochastic process. The algorithm includes stochastically taking any three edge pixels out of the image that lie on an edge curve, defining a small neighbourhood around the pixels, finding the line of best fit to those pixels within the neighbourhood with least squares method. The line through the midpoint of two stochastic points and the intersection of their tangents can be obtained, so the intersection of two such lines is the ellipse centre. This algorithm has serious disadvantages because the tangents at any point are sensitively changing for its neighbour pixels selected by the window, leading to the uncertainty of parameters and low accuracy.

Another improved approach is the consistent symmetric axis method (CSA) [8], which utilizes the information in-

*e-mail: longford@xjtu.edu.cn

herent in the symmetric axes throughout the entire process to compute all of the parameters. Since determination of the symmetric axes involves a set of points on the ellipse, this method fails to detect broken ellipses, and the accuracy is low.

A direct least square method [9,10] is an efficient parameters fitting of ellipse to scattered data. This approach is based on a least squares minimization and guarantees an ellipse-specific solution with no computational ambiguity. However, this algorithm is sensitive to noise and fails in detecting multiple ellipses in an image.

In this paper, we present a new ellipse detection algorithm (RED), which combines the advantages of the least square method with the randomized algorithm. Since the pixels in three windows, centred at three stochastic pixels, are enough to determine an ellipse with the least square method, suppose that many sets of three chosen edge pixels all come from the same ellipse, and then it seems very probable that the ellipse is real. Our proposed algorithm first randomly selects three edge pixels that are the centre of three windows with the same size in the image, and fit all these edge points in these three windows to an supposed ellipse by the least square method; then randomly selects the fourth edge pixel in the image, and defines a distance criterion to determine whether this pixel also comes from the supposed ellipse in the image. After finding a possible ellipse, then with high probability the ellipse seems to be real, so we apply an evidence-collecting process to further determine whether the possible ellipse is the desired ellipse. As shown in Fig. 1, three different elliptical edge arcs in their windows centred at the stochastic pixels can generally determine an ellipse. Since the proposed algorithm is not based on the technique of voting in the parameter space, it does not need extra accumulator storage. In fact, the memory requirements needed in the proposed algorithm are small. The proposed algorithm has some other advantages such as high speed and being robust to noise. Some synthetic images with different levels of noise and

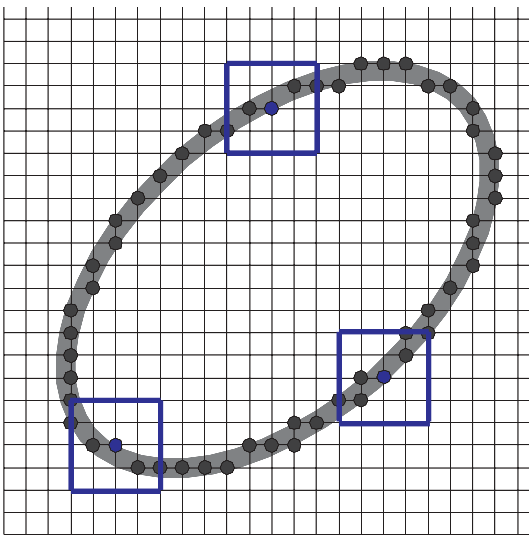


Fig. 1. A digital ellipse.

some realistic images that contain circular objects with some occluded ellipses and missing edges have been taken to verify the memory-saving and computational advantages of the proposed algorithm when compared to previous methods.

2. Determination of possible ellipses by least squares fitting

This section consists of three subsections which describe how to determine a possible ellipse according to the edge points in three windows centred at the stochastic pixels and the fourth randomly selected edge pixel. The first subsection describes how to obtain the coefficients of the ellipse equation by scattered data. The second subsection presents how to obtain the five parameters of the ellipse through six coefficients. The last subsection provides the distance criterion used to determine whether the fourth edge pixel selected lies on a possible ellipse or not.

2.1. Determination of ellipse coefficients

It is well known that accurate estimation of basic parameters of an elliptical shape is important for the accuracy of the 3D model of circular features. The general form of a common quadratic curve can be expressed as the form

$$F(u,v) = US = au^2 + buv + cv^2 + du + ev + f = 0, \quad (1)$$

where $U = [u^2\ uv\ v^2\ u\ v\ 1]$ and $S = [a\ b\ c\ d\ e\ f]^T$.

We can decompose the coefficients of the ellipse into

$$S = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix}, \quad (2)$$

where $S_1 = [a\ b\ c]^T$, $S_2 = [d\ e\ f]^T$.

Looking for accurate estimation of elliptical parameters, the least squares method is centred on finding the set of parameters that minimize the squares sum of an error of fit between the data points and the ellipse

$$e = \sum_{i=1}^{n} (au_i^2 + bu_iv_i + cv_i^2 + du_i + ev_i + f)^2 = \|SW\|^2, \quad (3)$$

where $W = [U_1\, U_2 \ldots U_n]^T$ is called the design matrix which can be described in detail as the size of $n \times 6$ matrix

$$W = \begin{bmatrix} u_1^2 & u_1v_1 & v_1^2 & u_1 & v_1 & 1 \\ u_2^2 & u_2v_2 & v_2^2 & u_2 & v_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_n^2 & u_nv_n & v_n^2 & u_n & v_n & 1 \end{bmatrix}. \quad (4)$$

We can decompose the design matrix W into its quadratic and linear parts

$$W = [W_1\, W_2], \quad (5)$$

where

$$W_1 = \begin{bmatrix} u_1^2 & u_1v_1 & v_1^2 \\ u_2^2 & u_2v_2 & v_2^2 \\ \vdots & \vdots & \vdots \\ u_n^2 & u_nv_n & v_n^2 \end{bmatrix}$$

and

$$W_2 = \begin{bmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ \vdots & \vdots & \vdots \\ u_n & v_n & 1 \end{bmatrix}.$$

In order to fit the ellipse with the data points, the constraint for the conic is well known that the discriminant $b^2 - 4ac$ is negative. Since we have the freedom to arbitrarily scale the parameters of the conic, we can impose the equality constraint $4ac - b^2 = 1$, which can be expressed in the matrix form of $S^T GS = 1$, this is

$$S^T \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} S = 1. \quad (6)$$

In the same way, the constraint matrix G can be expressed as

$$G = \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix}, \quad (7)$$

where

$$G_1 = \begin{bmatrix} 0 & 0 & 2 \\ 0 & -1 & 0 \\ 2 & 0 & 0 \end{bmatrix}.$$

Applying the decomposition principle, the constraint equation can be reformulated to

$$S_1^T G_1 S_1 = 1. \quad (8)$$

By introducing the Lagrange multiplier λ, we can get the simultaneous equations

$$\begin{cases} HS = \lambda GS \\ S^T GS = 1 \end{cases}, \quad (9)$$

where $H = W^T W$ is the scatter matrix of the size 6×6, which can be described in detail as

$$H = \begin{bmatrix} H_{u^4} & H_{u^3v} & H_{u^2v^2} & H_{u^3} & H_{u^2v} & H_{u^2} \\ H_{u^3v} & H_{u^2v^2} & H_{uv^3} & H_{u^2v} & H_{uv^2} & H_{uv} \\ H_{u^2v^2} & H_{uv^3} & H_{v^4} & H_{uv^2} & H_{v^3} & H_{v^2} \\ H_{u^3} & H_{u^2v} & H_{uv^2} & H_{u^2} & H_{uv} & H_u \\ H_{u^2v} & H_{uv^2} & H_{u^3} & H_{uv} & H_{v^2} & H_v \\ H_{u^2} & H_{uv} & H_{v^2} & H_u & H_v & H_1 \end{bmatrix}, \quad (10)$$

in which the element of H denotes the sum $H_{u^p v^q} = \sum_{i=1}^{n} u_i^p v_i^q$.

Similarly, the scatter matrix H can be split as the following blocked matrices

$$H = \begin{bmatrix} H_1 & H_2 \\ H_2^T & H_3 \end{bmatrix}, \quad (11)$$

where $H_1 = W_1^T W_1, H_2 = W_1^T W_2, H_3 = W_2^T W_2$.

Combining all the decompositions, we can get the following equation

$$\begin{bmatrix} H_1 & H_2 \\ H_2^T & H_3 \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = \lambda \begin{bmatrix} G_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix}, \quad (12)$$

which is equivalent to the following two equations

$$\begin{cases} H_1 S_1 + H_2 S_2 = \lambda G_1 S_1 \\ H_2^T S_1 + H_3 S_2 = 0 \end{cases}. \quad (13)$$

In the above equation, S_2 can be expressed as

$$S_2 = -H_3^{-1} H_2^T S_1. \quad (14)$$

So, we can yield

$$(H_1 - H_2 H_3^{-1} H_2^T) S_1 = \lambda G_1 S_1, \quad (15)$$

which can be rewritten as

$$G_1^{-1}(H_1 - H_2 H_3^{-1} H_2^T) S_1 = \lambda S_1. \quad (16)$$

Considering all the decomposition processes, we can obtain the following set of equations

$$\begin{cases} RS_1 = \lambda S_1 \\ S_2 = -H_3^{-1} H_2^T S_1, \\ S = [S_1 S_2]^T \end{cases} \quad (17)$$

where $R = G_1^{-1}(H_1 - H_2 H_3^{-1} H_2^T)$.

Since the fitting of a general conic to a set of points may be approached by minimizing the sum of squared algebraic distances of the points to the conic which is expressed by the coefficients S

$$\begin{aligned} &\min_S \sum_{i=1}^{n} F(u_i v_i)^2 = \min_S \|WS\|^2 = \min_S S^T W^T WS \\ &= \min_S \lambda S^T GS = \min_S \lambda \end{aligned}. \quad (18)$$

To solve these equations, we first need to get all possible solutions of the generalized eigenvectors then select those that corresponding to the minimal positive eigenvalue λ.

2.2. Determination of ellipse parameters

The given set of edge points $(u_i, v_i), i = 1, \ldots, n$ in the image coordinate system (u, v)

$$au^2 + buv + cv^2 + du + ev + f = 0. \quad (19)$$

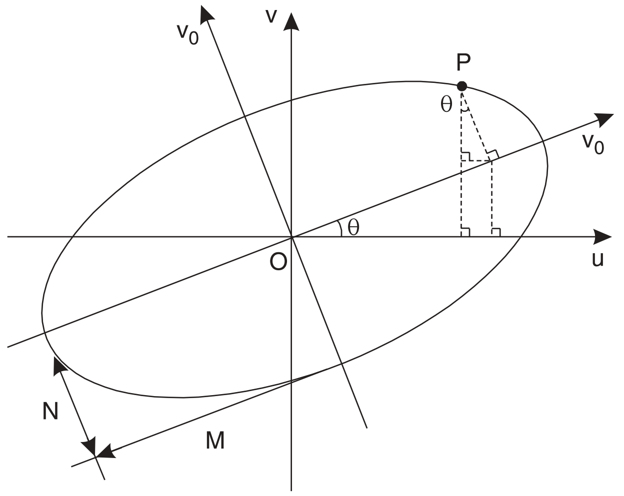


Fig. 2. Transformation for image coordinate.

If there is a new coordinate system (u_0, v_0) whose axes are parallel to major axis and minor axis of the ellipse, as shown in Fig. 2, then we have the following equation

$$\begin{cases} u = u_0 \cos\theta - v_0 \sin\theta \\ v = u_0 \sin\theta - v_0 \cos\theta' \end{cases} \quad (20)$$

which is equivalent to the following transformation matrix

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}. \quad (21)$$

The coefficients of $u_0 v_0$ is zero when the rotational angle of the ellipse is zero, so we have

$$-2a \sin\theta \cos\theta + 2c \sin\theta \cos\theta + b(\cos^2\theta - \sin^2\theta) = 0, \quad (22)$$

which can be solved as

$$\tan(2\theta) = \frac{b}{a-c}. \quad (23)$$

Since in the coordinate system (u, v), the curve should be satisfied with

$$au^2 + buv + cv^2 + du + ev + f = 0. \quad (24)$$

In the coordinate system (u_0, v_0), the curve should be

$$\begin{aligned} & a(u_0 \cos\theta - v_0 \sin\theta)^2 + b(u_0 \cos\theta - v_0 \sin\theta) \\ & \times(u_0 \sin\theta + v_0 \cos\theta) + c(u_0 \sin\theta + v_0 \cos\theta)^2 \\ & + d(u_0 \cos\theta - v_0 \cos\theta) + e(u_0 \sin\theta + v_0 \cos\theta) + f = 0 \end{aligned} \quad (25)$$

Therefore, the five parameters of an ellipse such as the centre point coordinates (u_c, v_c), the major axis length M, the minor axis length N, the angle or orientation of the ellipse θ, which is illustrated in Fig. 3 can be calculated using the following formulas:

– the rotational angle of the ellipse

$$\theta = \frac{1}{2}\arctan\frac{b}{a-c}, \quad (26)$$

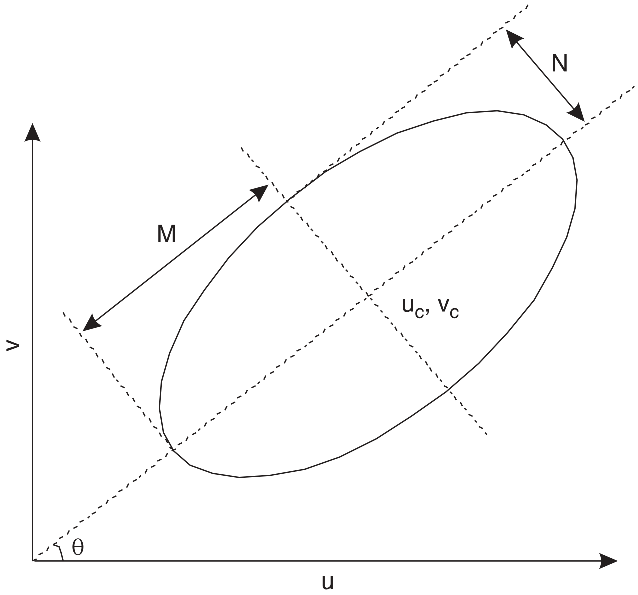


Fig. 3. Illustration of ellipse parameter.

– the center point coordinates

$$\begin{cases} u_c = -\dfrac{n_1}{2m_1}, \\ v_c = -\dfrac{n_2}{2m_2} \end{cases} \quad (27)$$

– the major axis length

$$M = \sqrt{\frac{m_2 n_1^2 + m_1 n_2^2 - 4m_1 m_2 f}{4m_1^2 m_2}}, \quad (28)$$

– the minor axis length

$$N = \sqrt{\frac{m_2 n_1^2 + m_1 n_2^2 - 4m_1 m_2 f}{4m_1 m_2^2}}, \quad (29)$$

where

$$m_1 = a\cos^2\theta + b\sin\theta\cos\theta + c\sin^2\theta,$$

$$n_1 = d\cos\theta + e\sin\theta,$$

$$m_2 = a\sin^2\theta - b\sin\theta\cos\theta + c\cos^2\theta,$$

$$n_2 = -d\sin\theta + e\cos\theta.$$

2.3. Determination of possible ellipse

Let V denotes the set of all edge pixels in the image, and $p_4 = (u_4, v_4)$ be the fourth selected edge pixel, then the distance between p_4 and the boundary of the ellipse, denoted by dist, can be calculated by

$$dist = \left| au_4^2 + bu_4 v_4 + cv_4^2 + du_4 + ev_4 + f \right|. \quad (30)$$

If p_4 lies on the ellipse, the ideal value of *dist* is zero. Since the image is digital, it rarely happens that the edge pixel lies exactly on an ellipse. Therefore, the goal of ellipse detection is to detect a set of edge pixels that lie not exactly but roughly on a digital ellipse. For convenience, the set of edge pixels that form a digital ellipse is also called an ellipse and these edge pixels are called co-elliptical. As shown in Fig. 1, if p_4 lies on the boundary of the ellipse, then the value of *dist* in the above equation is very small, which can be used to determine whether p_4 lies on the ellipse or not. Once we find the distance is smaller than the given threshold T_d, we claim that the fourth edge pixel is co-elliptical with the possible ellipse.

However, let us consider an undesirable case. When two of the three agent pixels of the possible ellipse are too close, the possible ellipse may not be the true ellipse. For example, p_1, p_2, and p_3 lie on a true ellipse, but the ellipse determined by p_1, p_2, and p_3 differs from the true ellipse. The undesirable case occurs when p_2, and p_3 are too close. To avoid this case, the distance between any two selected pixels must be greater than the given threshold T_a. If so, it means that the three agent pixels have strong evidence to be the representatives of the possible ellipse.

2.4. Determination of true ellipse

After detecting a possible ellipse with five parameters, whether the possible ellipse is a true ellipse can be checked by the following evidence-collecting process. Initially, we set a counter $num = 0$ for this possible ellipse in order to count how many edge pixels lie on the possible ellipse. For each edge pixel p_e in V, the distance *dist* can be obtained. If *dist* is not larger than the given distance threshold T_d, we increment the counter *num* by one and take p_e out of V, otherwise we proceed to the next edge pixel. We continue the above process until all the edge pixels in V have been examined. In the evidence-collecting process, let n_e denotes the number of edge pixels on the possible ellipse. In fact, the final value of *num* is equal to n_e. If n_e is larger than the given global threshold T_r, we claim that the possible ellipse is a true ellipse. Otherwise, the possible ellipse is a false ellipse and we return those n_e edge pixels into the set V. Ellipses with different axes have different circumferences. Therefore, employing some large global threshold T_r is unfair to those ellipses with small axes. To overcome the normalized problem, we apply a ratio to this threshold. Note that when a true ellipse is detected, then the edge pixels lying on the ellipse are taken out of the set of current edge pixels. This leads to speeding up the detection of the next ellipse.

3. Illustration of the proposed algorithm

From the basic principle described above, the proposed algorithm consists of the following steps.

Step 1. Store all edge pixels $p_i = (u_i, v_i)$ to the set V and initialize the failure counter f to be zero. Let T_f, T_{em}, T_a, T_d, T_r be the given thresholds. Here, n_p denote the number of edge pixels retained in V, and T_f denotes the number of failures that we can tolerate. If there are less than T_{em} pixels in V, we stop the task of ellipse detection. The distance between any two selected pixels of the possible ellipse should be larger than the distance threshold T_a. T_d is the given threshold for the distance between the fourth selected pixel p_4 and the boundary of the ellipse. T_r is the ratio threshold.

Step 2. If $f = T_f$ or $n_p < T_{em}$, then stop; otherwise, we randomly pick four pixels p_i, $i = 1,2,3,4$ out of V. When p_i has been chosen, take out of the set of current edge pixels $V = V - \{p_i\}$.

Step 3. From the four edge pixels, find out the possible ellipse such that the distance between any two of the three selected pixels is larger than T_a and the distance between the fourth pixel and the boundary of the possible ellipse is larger than T_d; then go to step 4. Otherwise, put p_i, $i = 1,2,3,4$ back to V; perform $f = f + 1$; then go to step 2.

Step 4. Assume E_{ijk} is the possible ellipse. Set the counter *num* to be 0. For each p_m in V, we check whether *dist* is not larger than the given distance threshold T_d. If yes, $num = num + 1$ and take p_m out of V. After examining all the edge pixels in V, assume $num = n_e$, i.e., there are n_e edge pixels satisfying $dist < T_d$.

Step 5. If $n_e \geq T_r$, go to step 6. Otherwise, regard the possible ellipse as a false ellipse, return these n_e edge pixels into V, perform $f = f + 1$; and go to step 2.

Step 6. The possible ellipse E_{ijk} has been detected as a true ellipse. Set f to be zero and go to step 2.

4. Experimental results

We performed all the experiments on a Pentium IV 2.4 GHz computer using Matlab language. The first experiment is tested on the 580 × 480 synthetic images that are created by adding spiced salt noise at various levels to three original images. The original synthetic image with 1164 edge pixels and 6984 noise points is shown in Fig. 4(a), which consists of three separate ellipses with different directions. The original synthetic image with 959 edge pixels and 5754 noise points is shown in Fig. 4(c), which consists of three overlapped ellipses with different directions. The original synthetic image with 712 edge pixels and 4272 noise points is shown in Fig. 4(e), which consists of three overlapped elliptical arcs with different directions. The detected ellipses proposed by our algorithm are shown in Fig. 4(b), Fig. 4(d), and Fig. 4(f), respectively, where the broad lines in the image display the detected major axis and minor axis of the ellipses, from which we can see their intersections is the center of the ellipses, and the slope of the major axis is the tangent of the rotational angle of the ellipse. The estimated five parameters of the ellipses detected in images in shown in Table 1.

For the purpose of comparison, we apply the RHT, the CSA and our proposed RED to a typical image as shown in Fig. 4(a). We perform detection of an algorithm by adding

Table 1. The estimated five parameters of the ellipses detected in Fig. 4.

Parameters	u_c	v_c	M	N	$\theta(°)$
Fig. 4(a)	371.4247 371.0429 227.2782	271.9050 131.8847 208.8658	79.8157 81.2140 81.1467	54.0932 53.9996 53.8514	89.9467 50.0629 0.2008
Fig. 4(c)	366.2288 204.8841 294.5926	203.4263 227.0736 263.0277	80.4720 80.8179 80.7442	53.9710 53.8436 54.0069	49.8350 89.8228 0.0321
Fig. 4(e)	261.3917 367.9825 366.1103	181.7210 254.7325 140.1047	80.2635 85.3853 66.3604	53.9371 52.9958 42.5389	–70.5887 9.8093 –41.1210

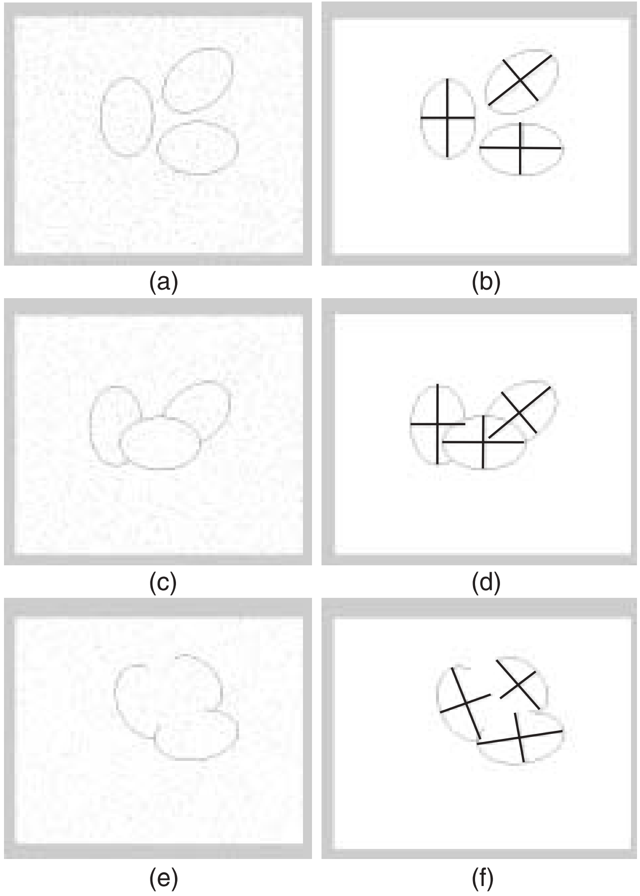


Fig. 4. The experiment on the synthetic images with noises. (a) The image with three separate ellipses. (b) The detected ellipses of the proposed algorithm. (c) The image with three occluded ellipses. (d) The detected ellipses of the proposed algorithm. (e) The image with three intersected elliptical arcs. (f) The detected elliptical arcs of the proposed algorithm.

increasing noise to the original image and the number of the noise is one multiple to six multiple of the edge pixels. Every image with different multiple of noise is performed 200 times to test the tolerance of the algorithms to uniform. The accuracy reduces when the noise increases from low to high level. The result of the experiment is shown in Fig. 5(a), from which we can conclude that the accuracy of the RED is much higher than the RHT and the CSA.

The execution time required in each method is measured in terms of seconds and it is obtained from the average of 200 simulations. Figure 5(b) illustrates the execution time required in the related three methods against the number of multiple of noise to the edge pixels. It is observed that the execution time required in the proposed RED is smaller than the time in the RHT and the CSA.

The second experiment is carried out on real images captured by our vision system in the lab, which consisted of the VC-C4 camera with the resolution of 320 × 240, and the image board BT848. We use an image with several ellipse features as shown in Fig. 6(a) to measure the validity of our proposed algorithm. The detected ellipses and elliptical arcs are traced out with curves, while the detected major axis and minor axis are expressed with lines, whose intersection is the centre of ellipse, as shown in Fig. 6(b). The estimated five ellipse parameters are shown in Table 2.

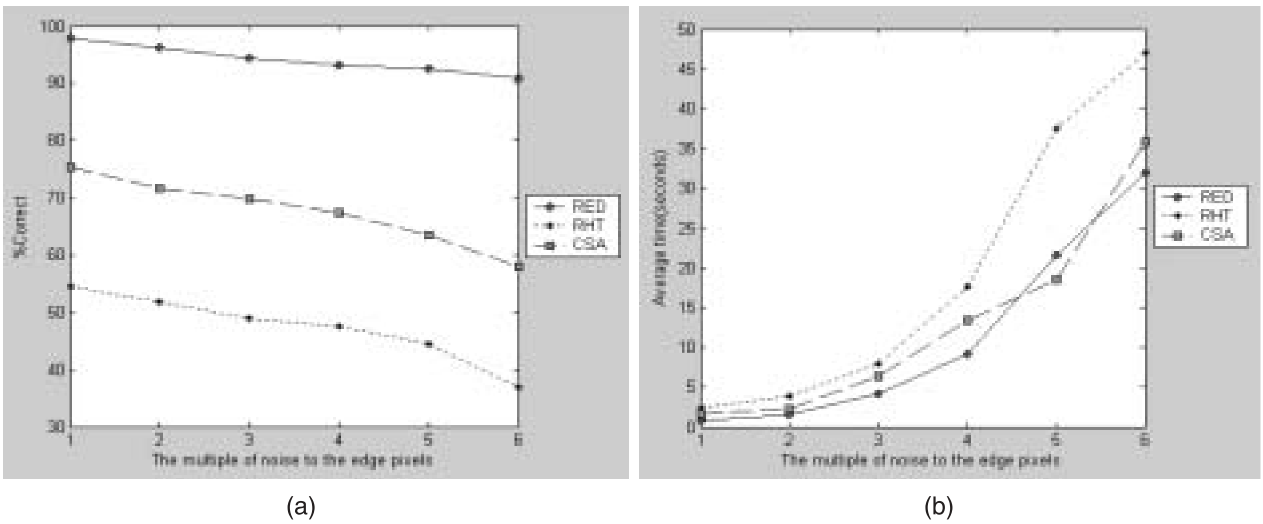
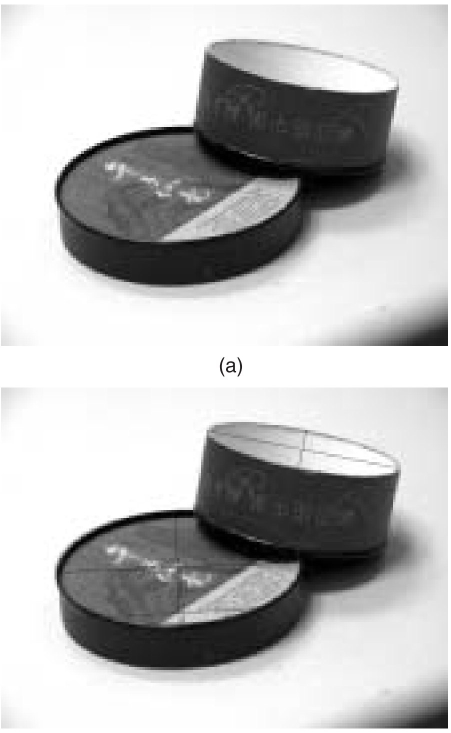


Fig. 5. Percentage accuracy with multiple noises (a) and average computation time with multiple noises (b).



(a)

(b)

Fig. 6. Real image (a) and ellipses detected by the proposed algorithm (b).

Table 2. The estimated five parameters of the ellipses detected in Fig. 7.

Parameters	u_c	v_c	M	N	$\theta(°)$
Ellipse 1	130.6774	125.6068	81.1252	38.4464	3.2145
Ellipse 2	133.5000	157.5221	78.9923	36.8724	1.0104
Ellipse 3	215.6895	45.8776	64.6687	13.3672	–11.1879

5. Conclusions

In this paper, a randomized algorithm based on the least square approach has been presented for the efficient detection of ellipses. The proposed algorithm is based on randomly picking three edge pixels that are the centre of three windows with the same size in the image, fitting ellipses to data points in these three windows by minimizing the algebraic distance, then randomly selects the fourth edge pixel and defines a distance criterion to determine whether there is a possible ellipse in the image. After we find a possible ellipse, we use an evidence-collecting process to check whether the possible ellipse is a true ellipse or not. The proposed algorithm does not need to vote in the parameter space, so it indeed does not need any extra storage for representing the accumulator which is needed in the previous Hough transform based methods. Some synthetic images with different levels of noise and some realistic images that contain circular objects with some occluded ellipses and missing edges have been taken to justify the memory-saving and computational advantages of the proposed algorithm. Experimental results demonstrate that the proposed algorithm is more accurate and faster than other methods in the literature.

References

1. N. Yamaguchi and H. Mizoguchi, "Robot vision to recognize both face and object for human-robot ball playing", *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, Vol. 2, 999–1004 (2003).
2. G. Adorni, S. Cagnoni, and M. Mordonini, "Landmark-based robot self-localization: a case study for the RoboCup goal-keeper", *Int. Conf. on Information Intelligence and Systems*, 164–171 (1999).
3. P.V.C. Hough, "Method and means for recognizing complex patterns", U.S. Patent 3069654, 1962.
4. Ch. The-Chuan and Ch. Kuo-Liang, "An efficient randomized algorithm for detecting circles", *Computer Vision and Image Understanding* **83**, 172–191 (2001).
5. Ch. Cheng and Y. Liu, "Efficient technique for ellipse detection using restricted randomized Hough transform", *Int. Con. on Information Technology: Coding and Computing*, Vol. 2, 714–718 (2004).
6. R.A. McLaughlin, "Randomized Hough transform: better ellipse detection", *IEEE Proc. on TENCON, Digital Signal Processing Applications*, Vol. 1, 409–414 (1996).
7. R.A. McLaughlin, "Randomized Hough transform: improved ellipse detection with comparison" **19**, 299–305 (1998).
8. H.T. Sheu, H.Y. Chen, and W.C. Hu, "Consistent symmetric axis method for robust detection of ellipses", *IEE Proceedings on Vision, Image and Signal Processing* **144**, 332–338 (1997).
9. R. Halir and J. Flusser, "Numerically stable direct least squares fitting of ellipses", *6th Int. Conf. in Central Europe on Computer Graphics and Visualization*, (1998).
10. A. Fitzgibbon, M. Pilu, and R.B. Fisher, "Direct least square fitting of ellipses", *IEEE Transactions on Pattern Analysis and Machine Intelligence* **21**, 476–480 (1999).