ELSEVIER

Image and Vision Computing 19 (2001) 857–866

image AND vision COMPUTING

www.elsevier.com/locate/imavis

Detection ellipses by finding lines of symmetry in the images via an hough transform applied to straight lines

Adel A. Sewisy[a,*], Franz Leberl[b]

[a]*Department of Mathematics, Faculty of Science, University of Assiut, Assiut 71516, Egypt*
[b]*Institute of Computer Graphics, Technical University Graz, A-8010 Graz, Austria*

Received 22 September 1999; revised 19 January 2001; accepted 21 January 2001

Abstract

Through the use of a global geometric symmetry, detection ellipses are proposed in this paper. Based on the geometric symmetry, the proposed method first locates candidates of ellipses centers. In the meantime, according to these candidate centers, all feature points in an input image are grouped into several subimages. Then, for each subimage, by using geometric properties again, all ellipses are extracted. The method significantly reduces the time required to evaluate all possible parameters without using edge direction information. Experimental results are given to show the correctness and effectiveness of the proposed method. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Hough transform; Lines detection; Ellipses detection; Accumulator array

1. Introduction

One of the basic tasks in computer vision is the detection of straight lines, circles, ellipses, etc. from an image. The hough transform (HT) and its variants [2,4,6,7,10] are methods commonly used for line detection, but applying it to detect ellipses requires a five-dimensional array for the accumulator. Moreover, a great deal of computing time is needed to transform a feature point in an input image to many points in the parameter space. Several modified versions of HT have been proposed. Tsuji and Matsumoto [11] and Sewisy [9] first introduced the decomposition concept by the use of parallel tangents. Ballard [1] presented a method which simplifies the parameterization by deriving the ellipse center from the coordinates and tangent of an edge pixel. Illingworth et al. [12] and Illingworth and Kittler [3] proposed a better center finding process and used adaptive HT to estimate three other parameters. Tam Peter et al. [5] presented an approach which utilizes parallel edge points to deduce parameters. The methods mentioned above utilize gradient information to reduce the dimension of the parameter space, and the problem is broken down into multiple stages. But, estimates of the parameters of ellipses based on local geometric properties often suffer from poor consistency and locating accuracy because of noise and quantization error. To avoid these disadvantages, in this paper, we will present a new method that uses the global geometric symmetry of ellipses to reduce the dimension of the parameter space. In the proposed method, first a global geometric symmetry is used to locate all possible symmetric centers of ellipses in an image, and then all feature points are classified into several subimages according to these center points. Ellipses with different symmetric centers will lie in different subimages. Then the geometric properties are applied again in each subimage to find all possible sets of three parameters (the length of the major axis, the length of the minor axis and orientation) for ellipses. Finally, the accumulative concept of the HT is used to extract all ellipses in the input image. The remainder of this paper is organized as follows: Section 2 describes symmetric edge points about the straight line. Section 3 describes the proposed technique. Section 4 gives computer experiments. Section 5 presents a comparison of methods and Section 6 summary and conclusions.

2. Symmetric edge points about the straight line

Suppose that the points $P(x, y)$, $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ lie on an ellipse, we locate the point (x_m, y_m) appearing on straight line l which passes through the two points $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ with these equations

$$x_m = \tfrac{1}{2}(x_1 + x_2), \tag{1}$$

* Corresponding author.
E-mail addresses: sewisy@acc.aun.eun.eg (A.A. Sewisy).

0262-8856/01/$ - see front matter
PII: S0262-8856(01)00049-X

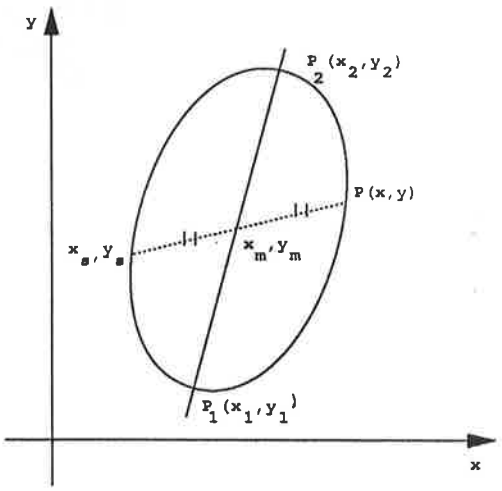

Fig. 1. Symmetric edge points about the straight line l.

$$y_m = \tfrac{1}{2}(y_1 + y_2). \quad (2)$$

The symmetric points (x_s, y_s) of $P(x, y)$ about this line l can be found using Eqs. (1) and (2), see Fig. 1. We have

$$x_s = x - 2x_m, \quad (3)$$

$$y_s = y - 2y_m. \quad (4)$$

3. Proposed technique

The proposed technique consists of two phases: detecting center with symmetric points and extracting ellipses with parameter's estimate.

3.1. Phase 1: detecting center with symmetric points

Let I be an input image with size $(n \times n)$ pixel and O be the binary 'object' result of the edge — extracted image form I, with 0 representing a black point and 1 a white point. Then, phase 1 is carried out.

First: O is scanned (rightward) from left to right and from top to bottom by the following procedure, the scanning rightward procedure:

* for each edge points (i, j) in O *\
For $i = 1$ To n * from left to right *\
For $j = 1$ To n * from top to bottom *\
$A(i,j) = 0$; * to clear image A *\
For $i = 1$ To n * from left to right *\
For $j = 1$ To n * from top to bottom *\
if $O(i,j) = 1$ then * the scan line meets an edge point (i, j) *\
For $k = i + 1$ To n * from left to right *\
if $O(k,j) = 1$ then * the scan line meets another edge point (k, j) *\
$A((i + k)/2, j) = 1$; * set the midpoints of (i, j) and (k, j) in the image A *\

Second: The vertical straight lines appearing in A are extracted using HT. According to these lines, we consider each extracted line to be a symmetric vertical axis l_v of an ellipse.

3.1.1. Symmetric edge points by scanning rightward procedure

For each vertical straight line l_{vi}, $vi = 0, \ldots, k$, where k is the number of vertical straight lines in A, we locate all pairs of edge points symmetric about l_{vi} and put them in f_v. The following algorithm is proposed:

Input: An image O * edge point *\
Output: Several subimages of O, for example, o_i, $i = 0, \ldots, k$.
Step 1: Transform O to A by the scanning rightward procedure * O is scanned rightward *\
Step 2: Extract all vertical straight lines in A by the HT. If there exist such vertical straight lines, for example, $l_{v1}, l_{v2}, \ldots, l_{vk}$, then
continue;
else
stop.
Step 3: For each vertical straight line l_{vi}, $vi = 0, \ldots, k$, where k is the number of vertical straight lines in A, we locate all pairs of edge points symmetric about l_{vi} and put them in f_v. The details are in Steps 3.1–3.3.
Step 3.1: Locate the points appearing on l_{vi}, for example, $(x_{m1}, y_{m1}), (x_{m2}, y_{m2}), \ldots, (x_{mp}, y_{mp})$.
Step 3.2: Scan O and identify the pair of edge points symmetric about l_{vi}, at x_{mi}, y_{mi}, $vi = 0, \ldots, p$.
Step 3.3: Put in all these symmetric pairs in f_v.

Third: O is scanned (downward) from top to bottom and from left to right by the following procedure, the scanning downward procedure:

* for each edge points (i, j) in O *\
For $i = 1$ To n * from top to bottom *\
For $j = 1$ To n * from left to right *\
$B(i,j) = 0$; * to clear image B *\
For $i = 1$ To n * from top to bottom *\
For $j = 1$ To n * from left to right *\
if $O(i,j) = 1$ then * the scan line meets an edge point (i, j) *\
$B(i, (j + k)/2) = 1$; * set the midpoints of (i, j) and (i, k) in the image B *\

Fourth: The horizontal straight lines appearing in B, are extracted using HT. According to these lines, we consider each extracted line to be a symmetric horizontal axis l_h of an ellipse.

3.1.2. Symmetric edge points by scanning downward procedure

For each horizontal straight line l_{hi}, $hi = 0, \ldots, k$, where k

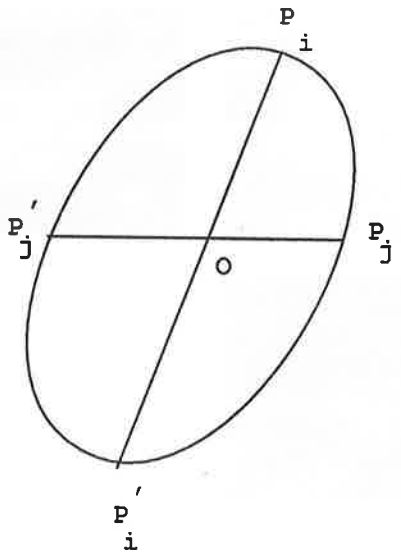

Fig. 2. Illustration of the symmetry property of an ellipse $|\overline{OP_i}| = |\overline{OP_i'}|$ for all i and $|\overline{OP_j}| = |\overline{OP_j'}|$ for all j.

is the number of straight lines in B, we locate all pairs of edge points symmetric about l_{hi} and put them in f_h. The following algorithm is proposed:

Input: An image O * edge point *\
Output: Several subimages of O, for example, o_i, $i = 0, \ldots, k$.
Step 1: Transform O to B by the scanning downward procedure * O is scanned downward *\
Step 2: Extract all horizontal straight lines in B by the HT. If there exist such horizontal straight lines, for example, $l_{h1}, l_{h2}, \ldots, l_{hk}$, then
continue;
else
stop.
Step 3: For each horizontal straight line l_{hi}, $hi = 0, \ldots, k$, where k is the number of horizontal straight lines in B, we locate all pairs of edge points symmetric about l_{hi} and put them in f_h. The details are in Steps 3.1–3.3.
Step 3.1: Locate the points appearing on l_{hi}, for example, $(x_{m1}, y_{m1}), (x_{m2}, y_{m2}), \ldots, (x_{mp}, y_{mp})$.
Step 3.2: Scan O and identify the pair of edge points symmetric about l_{hi} at x_{mi}, y_{mi}, $hi = 0, \ldots, p$.
Step 3.3: Put in all these symmetric pairs in f_h.

Fifth: We consider the cross point of line l_h and l_v to be a center of an ellipse in image (x_0, y_0).

3.2. Phase 2: parameters estimation

3.2.1. Finding ellipse boundary points

After the center has been detected, phase 2 is to locate all the points, which are symmetric to the center of an ellipse and they lie on the ellipse. In order to find the ellipse boundary points by using the detected center, we present some properties for ellipse, which are described as basic geometric properties of an ellipse.

3.2.2. Property 1

Suppose that point $O(x_0, y_0)$ is the center of an ellipse, and points P_i and P_i' are on the ellipse, see Fig. 2. If segment P_iP_i' passes through the center $O(x_0, y_0)$, then

$$|\overline{OP_i}| = |\overline{OP_i'}|. \tag{5}$$

3.2.3. Property 2

Suppose that a and b are the lengths of the semi-axis of an ellipse, $O(x_0, y_0)$ the center, and, P_1 and P_2 are two points on the ellipse, see Fig. 3. If $<P_1OP_2 = \pi/2$, then

$$\frac{1}{|\overline{OP_1}|^2} + \frac{1}{|\overline{OP_2}|^2} = \frac{1}{a^2} + \frac{1}{b^2}. \tag{6}$$

For proof, see Ref. [8]. The above properties of an ellipse are used to identify the ellipse points on the boundary of the ellipse. If the points P_1, P_2 and P_3 are considered to lie on an ellipse. Then P_1, P_2 and P_3 are shifted $O(-x_0, -y_0)$ to move the ellipse center to $O(0, 0)$.

$$x_1'^2 + Bx_1'y_1' + Cy_1'^2 + F' = 0. \tag{7}$$

Since an ellipse with center $O(0, 0)$ can be expressed by Eq. (7), by substituting the coordinates of the shifted P_1, P_2 and P_3 into Eq. (7), we can obtain three equations with three unknown parameters (B, C, F') to be found from

$$x_1'^2 + Bx_1'y_1' + Cy_1'^2 + F' = 0, \tag{8}$$

$$x_2'^2 + Bx_2'y_2' + Cy_2'^2 + F' = 0, \tag{9}$$

$$x_3'^2 + Bx_3'y_3' + Cy_3'^2 + F' = 0. \tag{10}$$

The three parameters (a, b, θ) for the ellipse can be obtained by the following formulas, see Ref. [8].

$$a = \left[\frac{-2F'}{\left((1 + C) + \sqrt{(1 - C)^2 + B^2}\right)}\right]^{1/2}, \tag{11}$$

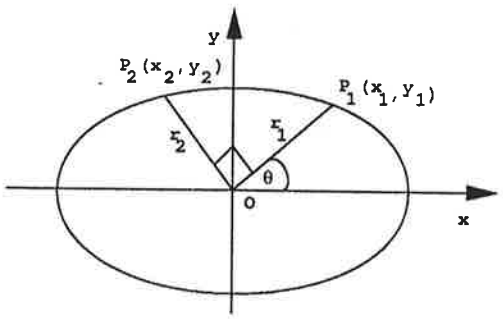

Fig. 3. The geometric property of an ellipse $(x^2/a^2) + (y^2/b^2) = 1$: $(1/|\overline{OP_1}|^2) + (1/|\overline{OP_2}|^2) = (1/a^2) + (1/b^2)$.

$$b = \left[\frac{-2F'}{\left((1+C) - \sqrt{(1-C)^2 + B^2}\right)}\right]^{1/2}, \tag{12}$$

$$\theta = \frac{1}{2}\tan^{-1}\left(\frac{B}{1-C}\right). \tag{13}$$

Then the accumulator $acc(a, b, \theta)$ is incremented by unity. After processing all points in f_h or f_v in this way, we obtain a local peak of the array $acc(a, b, \theta)$, which gives us valuable information of an ellipse. The details of phase 2 are described as follows:

Initialize an array $acc(a, b, \theta)$
Input (x_0, y_0) * center for each subimage f_h or f_v *\
For $i = 1$ To n * image point i in f_h or f_v *\
For $j = i + 1$ To n * image point j in f_h or f_v *\
If (points i and j satisfy Property 2) then
For $k = 1$ To n * excluding $k = i$ and $k = j$ *\
If (points k and i satisfy Property 1) then
finding P_1, P_2 and P_3
P_1, P_2 and P_3 are shifted to (x_0, y_0)
If ($P_1 \neq P_3$) then
obtain coefficient of an ellipse (B, C, F') by solving the systems (8), (9) and (13) using Gauss elimination,
then
locate a, b and θ then * parameters of an ellipse with Eqs. (11)–(13) *\ $acc(a, b, \theta) = acc(a, b, \theta) + 1$,
end * for $P_1 \neq P_3$ *\
else
using P_1 and P_2 * $P_1 \neq P_2$ *\
locate C and F' * with Eq. (14) *\
estimate a and b then * with Eqs. (15) and (16) *\
$acc1(a, b,) = acc1(a, b) + 1$,
end * for else *\
end * for points k and i satisfy Property 1 *\
end * for image point k *\
end * for points i and j satisfy Property 2 *\
end * for j *\
end * for i *\

with the peaks found in accumulator array $acc(a, b, \theta)$, they are considered to be parameters of ellipses. Note that if $B = 0$, i.e. ellipse without any orientation ($\theta = 0$), we use two points to find (C, F') from this equation

$$x'^2 + Cy'^2 + F' = 0, \tag{14}$$

then we get a and b from Eqs. (11) and (12) after we let $B = 0$

$$a = \sqrt{F'}, \tag{15}$$

$$b = \sqrt{\frac{F'}{C}}, \tag{16}$$

and the array $acc(a, b, \theta)$ is replaced by $acc1(a, b)$.

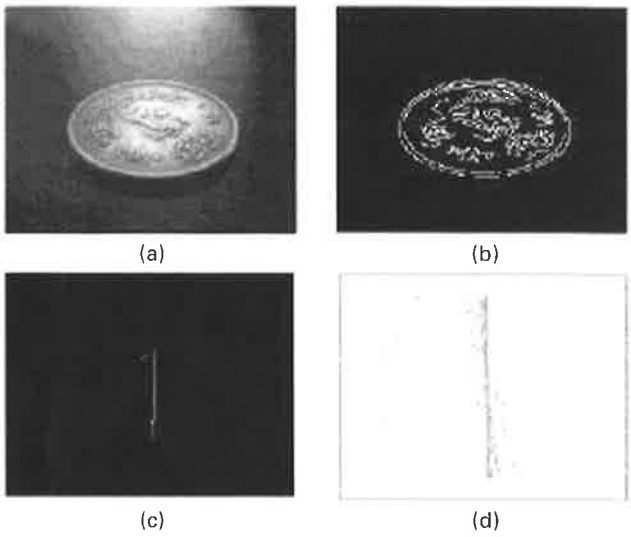

Fig. 4. The detection of ellipses in image 1: (a) image 1; (b) the extracted edge points of (a); (c) and (d) the results of applying the proposed method.

4. Computer experiments

Image 1, see Fig. 4(a) (320 × 240), shows a real image of Egyptian coin. The extracted edge points of image 1 are shown in Fig. 4(b). After the scanning rightward is applied, Fig. 4(b) is transformed into another image 'A' shown in

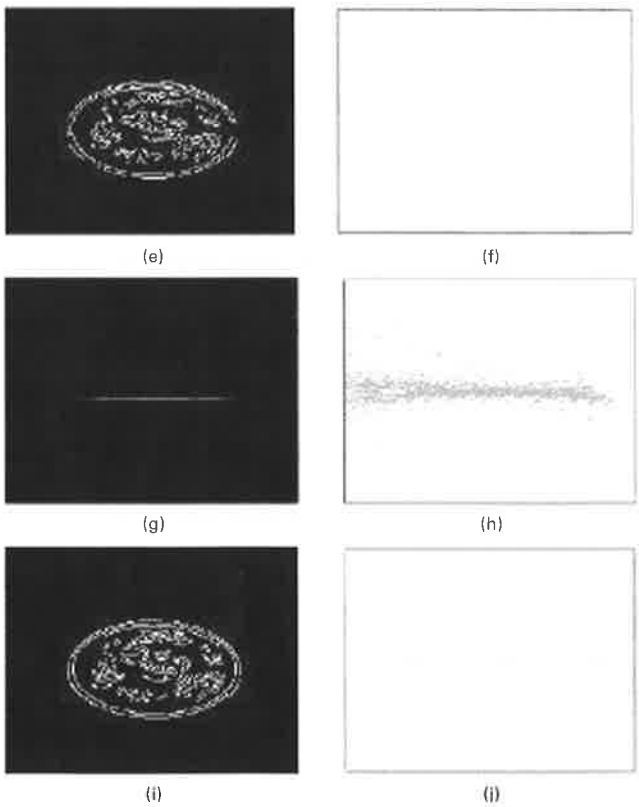

Fig. 5. Continued Fig. 4. The detection of ellipses in image 1: (e)–(j) the results of applying the proposed method.

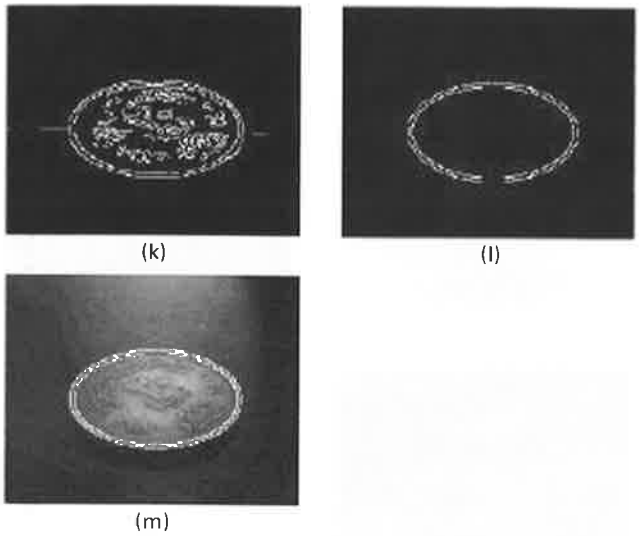

Fig. 6. Continued Fig. 4. The detection of ellipses in image 1: (k) and (l) the results of applying the proposed method; (m) the detected ellipses superimposed on Fig. 4(a).

Fig. 4(c). Then, the HT is applied to detect vertical straight lines l_{vi}. There are 25 vertical straight lines, see Fig. 4(d). The symmetric edge points about the detected vertical straight lines in Fig. 4(d) are shown in Fig. 5(e). The vertical straight line l_v with the maximum peak by using a window 9×9 and threshold equal to 10 is shown Fig. 5(f). After the scanning downward is applied, Fig. 4(b) is mapped into another image 'B' shown in Fig. 5(g). There are 82 horizontal straight lines l_{hi} detected by using HT see Fig. 5(h). Fig. 5(i) shows symmetric edge points about the detected lines l_{hi} in Fig. 5(h). The horizontal straight line l_h with the maximum peak is shown in Fig. 5(j). The detected center of ellipse with the intersection of the vertical straight line l_v and the horizontal straight line l_h is shown in Fig. 6(k). When we apply the proposed technique to all of the 25 subimages or 82 subimages, the ellipses are extracted and superimposed on the original image, see Fig. 6(l) and (m). The ellipses cannot be extracted from 25 or 82 subimages, because (1) there are two peaks detected and (2) there are many of straight lines that are not the real symmetric axis of the ellipses, see Fig. 7. Consequently, only two ellipses are extracted from the 82 subimages. Image 2, see Fig. 8(a) (320×240), shows a real image that includes partially occluded ellipses. Fig. 8(b) shows the extracted edge points of image 2. Applying the proposed method to Fig. 8(b) yielded two subimages corresponding to the two centers,

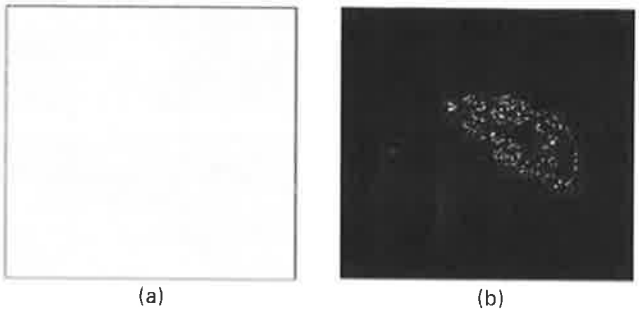

Fig. 7. (a) One of the other straight lines detected in Fig. 5(h). (b) The edge symmetric points about the straight line in (a).

as shown in Figs. 8(c), (d) and 9(e)–(i). The final result is shown in Fig. 9(j). Image 3, see Fig. 10(a) (300×300), shows a real image of lens cap. Fig. 10(b) shows the extracted edge points of image 3. Applying the proposed method to Fig. 10(b) yielded one subimage corresponding to one center, as shown in Figs. 10(c), (d), 11(e)–(h) and 12(i), image 'A' as shown in Figs. 10(c), (d), 11(e)–(h) and 12(i), (j). The result of applying the proposed method to Fig. 10(b), and (Fig. 12(k)) the detected ellipse superimposed on Fig. 10(a). Image 4, see Fig. 13(a) (256×256), shows a synthetic image. It includes two overlapping ellipses and two concentric ellipses. Image 'A' is shown in Fig. 13(b), with the procedure rightward is applied of, Fig. 13(a). There are three symmetric vertical axis l_{vi} to be extracted by using a window 15×15 and threshold equal to 8, see Fig. 13(c). Accordingly to extract l_{vi} lines, we can separate the original (Fig. 13(a)) into three subimages by finding the symmetric edge points about the detected of these l_{vi} (Figs. 13 (d) and 14(e), (f)). Note that the ellipses with different symmetrical vertical axis, we can separate, i.e. we can put every ellipse into one subimage, see Figs. 13(d) and 14(e), but the ellipses with the same symmetrical vertical axis, we cannot separate, see Fig. 14(f). Applying the scanning downward procedure to Fig. 13(a) yielded the centers of ellipses as shown in Figs. 14(g)–(j) and 15(k). The final result is shown in Fig. 15(l). Image 5, see Fig. 16(a), shows another synthetic image including three ellipses: intermittent, 25% and 50% defect. The detecting results are shown in Fig. 17(g). From this figure, we can see that ellipse with 50% defect is not detected. This is due to that many points on the boundary of the ellipse cannot be used from parallelograms of the type employed in phase 2. In general, if the defectiveness of an ellipse exceeds 50%, the proposed method may not work well.

Table 1
The final results for author's methods and the proposed method

Method	Tsuji and Matsumoto	Yuen et al.'s	Yip et al.'s	Proposed method
The final result of image 1	Successful	Successful	Failure	Successful
The final result of image 3	Failure	Failure	Failure	Successful

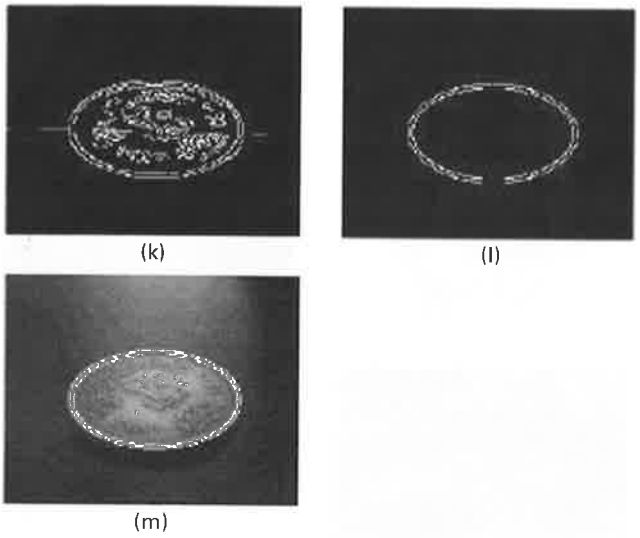

Fig. 6. Continued Fig. 4. The detection of ellipses in image 1: (k) and (l) the results of applying the proposed method; (m) the detected ellipses superimposed on Fig. 4(a).

Fig. 4(c). Then, the HT is applied to detect vertical straight lines l_{vi}. There are 25 vertical straight lines, see Fig. 4(d). The symmetric edge points about the detected vertical straight lines in Fig. 4(d) are shown in Fig. 5(e). The vertical straight line l_v with the maximum peak by using a window 9×9 and threshold equal to 10 is shown Fig. 5(f). After the scanning downward is applied, Fig. 4(b) is mapped into another image 'B' shown in Fig. 5(g). There are 82 horizontal straight lines l_{hi} detected by using HT see Fig. 5(h). Fig. 5(i) shows symmetric edge points about the detected lines l_{hi} in Fig. 5(h). The horizontal straight line l_h with the maximum peak is shown in Fig. 5(j). The detected center of ellipse with the intersection of the vertical straight line l_v and the horizontal straight line l_h is shown in Fig. 6(k). When we apply the proposed technique to all of the 25 subimages or 82 subimages, the ellipses are extracted and superimposed on the original image, see Fig. 6(l) and (m). The ellipses cannot be extracted from 25 or 82 subimages, because (1) there are two peaks detected and (2) there are many of straight lines that are not the real symmetric axis of the ellipses, see Fig. 7. Consequently, only two ellipses are extracted from the 82 subimages. Image 2, see Fig. 8(a) (320×240), shows a real image that includes partially occluded ellipses. Fig. 8(b) shows the extracted edge points of image 2. Applying the proposed method to Fig. 8(b) yielded two subimages corresponding to the two centers,

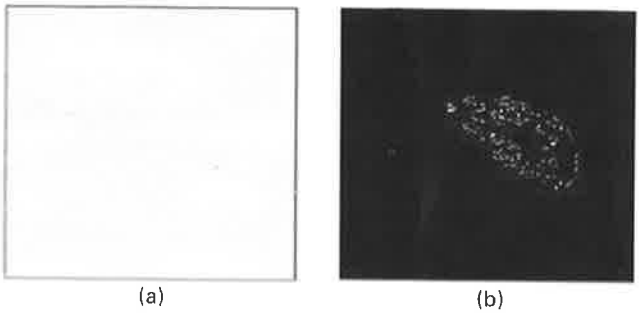

Fig. 7. (a) One of the other straight lines detected in Fig. 5(h). (b) The edge symmetric points about the straight line in (a).

as shown in Figs. 8(c), (d) and 9(e)–(i). The final result is shown in Fig. 9(j). Image 3, see Fig. 10(a) (300×300), shows a real image of lens cap. Fig. 10(b) shows the extracted edge points of image 3. Applying the proposed method to Fig. 10(b) yielded one subimage corresponding to one center, as shown in Figs. 10(c), (d), 11(e)–(h) and 12(i), image 'A' as shown in Figs. 10(c), (d), 11(e)–(h) and 12(i), (j). The result of applying the proposed method to Fig. 10(b), and (Fig. 12(k)) the detected ellipse superimposed on Fig. 10(a). Image 4, see Fig. 13(a) (256×256), shows a synthetic image. It includes two overlapping ellipses and two concentric ellipses. Image 'A' is shown in Fig. 13(b), with the procedure rightward is applied of, Fig. 13(a). There are three symmetric vertical axis l_{vi} to be extracted by using a window 15×15 and threshold equal to 8, see Fig. 13(c). Accordingly to extract l_{vi} lines, we can separate the original (Fig. 13(a)) into three subimages by finding the symmetric edge points about the detected of these l_{vi} (Figs. 13 (d) and 14(e), (f)). Note that the ellipses with different symmetrical vertical axis, we can separate, i.e. we can put every ellipse into one subimage, see Figs. 13(d) and 14(e), but the ellipses with the same symmetrical vertical axis, we cannot separate, see Fig. 14(f). Applying the scanning downward procedure to Fig. 13(a) yielded the centers of ellipses as shown in Figs. 14(g)–(j) and 15(k). The final result is shown in Fig. 15(l). Image 5, see Fig. 16(a), shows another synthetic image including three ellipses: intermittent, 25% and 50% defect. The detecting results are shown in Fig. 17(g). From this figure, we can see that ellipse with 50% defect is not detected. This is due to that many points on the boundary of the ellipse cannot be used from parallelograms of the type employed in phase 2. In general, if the defectiveness of an ellipse exceeds 50%, the proposed method may not work well.

Table 1
The final results for author's methods and the proposed method

Method	Tsuji and Matsumoto	Yuen et al.'s	Yip et al.'s	Proposed method
The final result of image 1	Successful	Successful	Failure	Successful
The final result of image 3	Failure	Failure	Failure	Successful

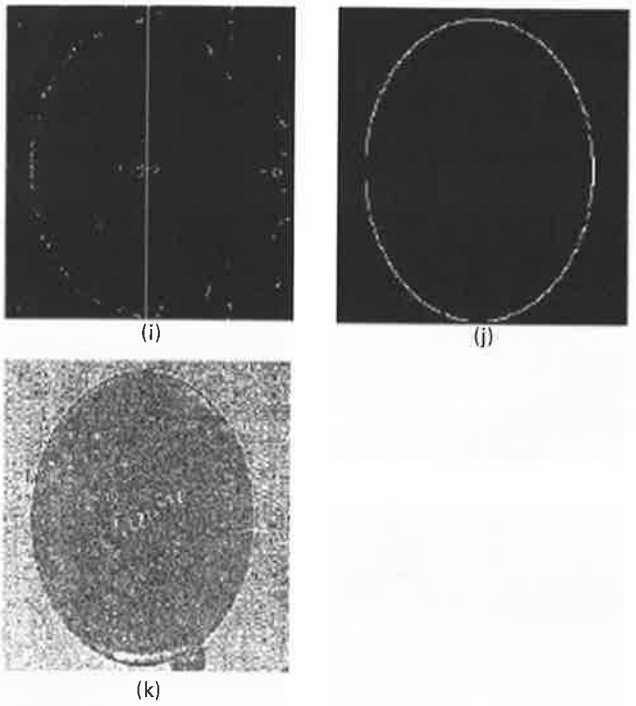

Fig. 12. Continued Fig. 10. The detection of ellipses in image 3: (i) and (j) the results of applying the proposed method; (k) the detected ellipse superimposed on Fig. 10(a).

5. A comparison of methods

For comparison purposes, we apply the Tsuji's [11] method, Yuen et al.'s [12] method and Yip et al.'s [5] method to system 4(a) and Fig. 10(a). Table 1 summarizes the final result for these methods and proposed method. The CPU times for these methods and proposed method are indicated in Table 2.

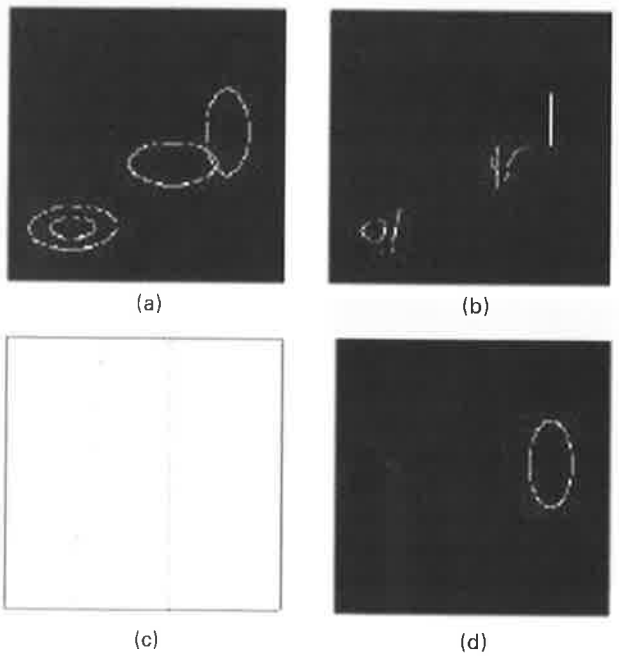

Fig. 13. The detection of ellipses in image 4: (a) image 4; (b) shows an image 'A'; (c) there are three symmetric vertical axis l_{vi} detected in (b); (d) the ellipse with the symmetric axis on line in (c).

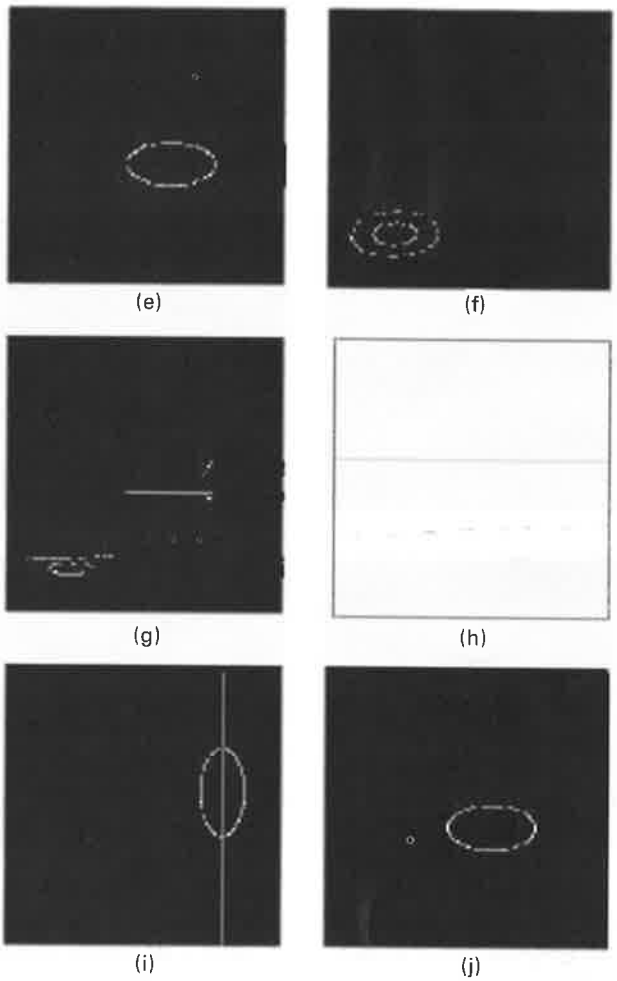

Fig. 14. Continued Fig. 13. The detection of ellipses in image 4: (e) the ellipse with the symmetric axis on line in Fig. 13(c); (f) the ellipses with the same symmetric axis on line in Fig. 13(c); (g)–(j) applying the scanning downward procedure.

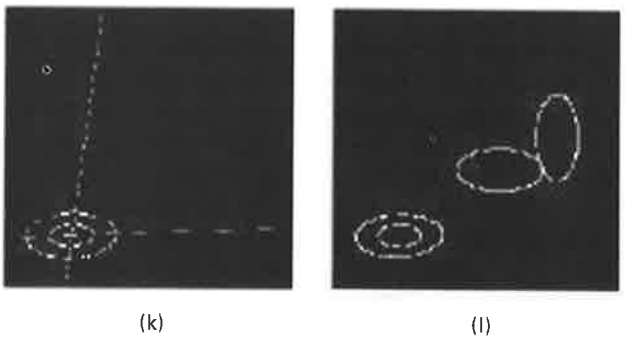

Fig. 15. Continued Fig. 13. The detection of ellipses in image 4: (k) applying the scanning downward procedure; (l) the extracted ellipses.

Table 2
The CPU times for author's methods and the proposed method

Method	Tsuji and Matsumoto	Yuen et al.'s	Yip et al.'s	Proposed method
CPU time for image 1 (s)	1312.25	3208.2	720.67	195.80
CPU time for image 3 (s)	3074.13	4785.80	3105.73	200.94

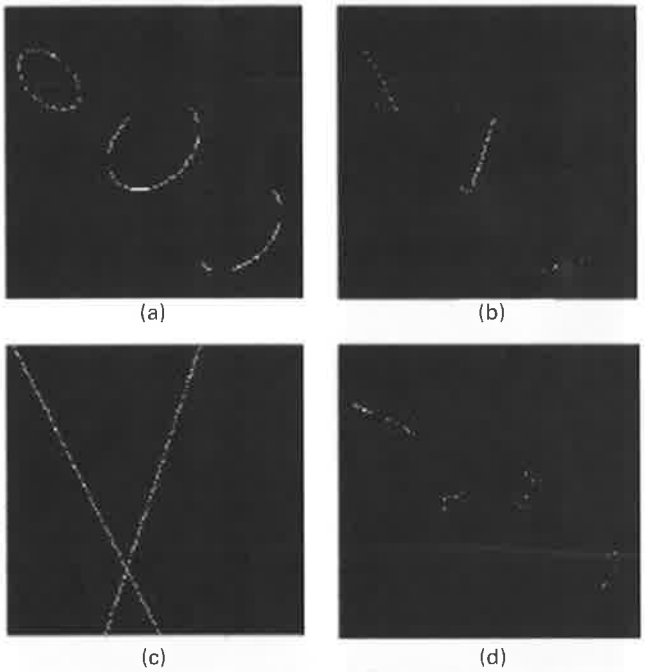

Fig. 16. The detection of ellipses in image 5: (a) three ellipses: intermittent, 25% defect and 50% defect; (b)–(d) the results of applying the proposed method.

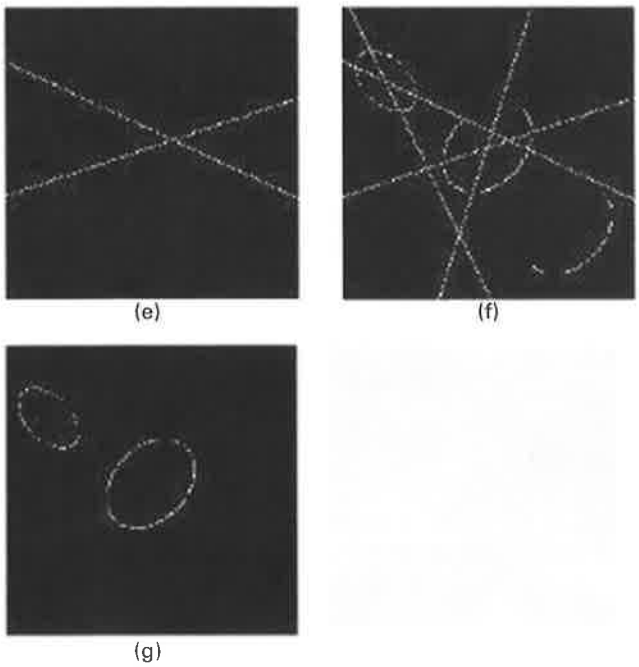

Fig. 17. Continued Fig. 16. The detection of ellipses in image 5: (e) and (f) the results of applying the proposed method; (g) two ellipses are detected, the ellipse with 50% defect is not located.

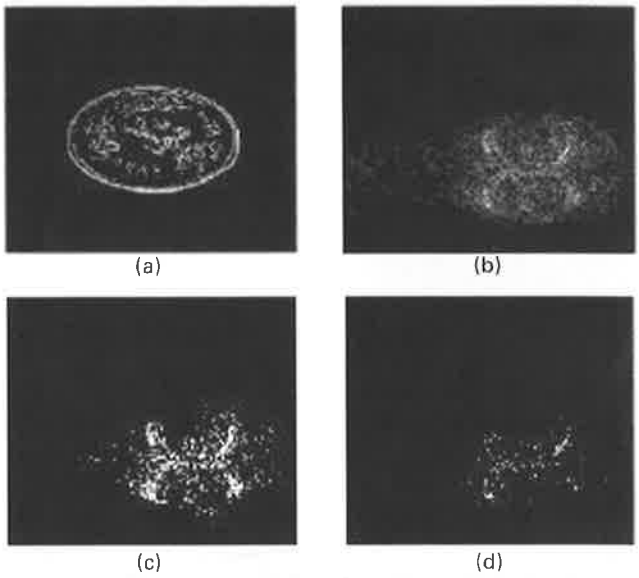

Fig. 18. Experimental results of applying Yip et al.'s method to detect the ellipses in an image 1 (Fig. 4(a)): (a) the edge-enhanced image of Fig. 4(a) using gradient edge operator with threshold at 10; (b) shows the accumulator array for detecting vertex positions; (c) and (d) show the threshold begin set to 25 and 50% of the maximum value of the accumulator array.

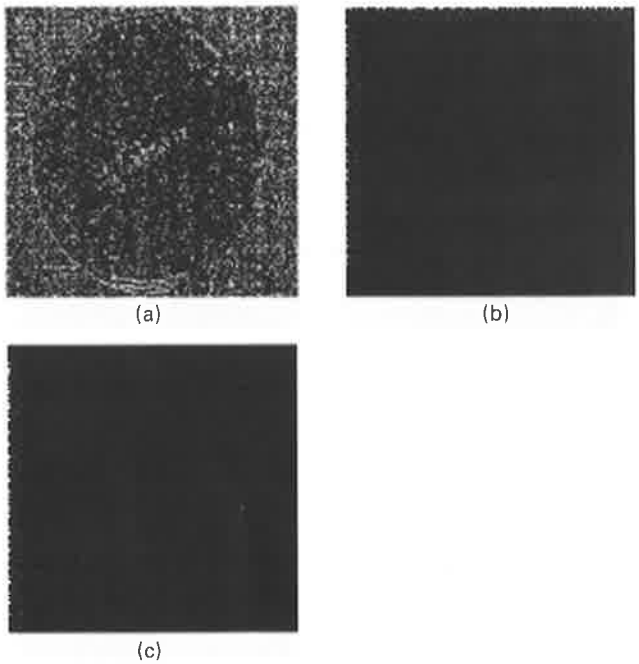

Fig. 19. Experimental results of applying the Yip et al.'s method to detect ellipses in image 3 (Fig. 10(a)): (a) the edge-enhanced image of Fig. 10(a) using gradient edge operator with threshold at 36; (b) shows the accumulator array; (c) shows the threshold set to 95% of the maximum counts of the accumulator.

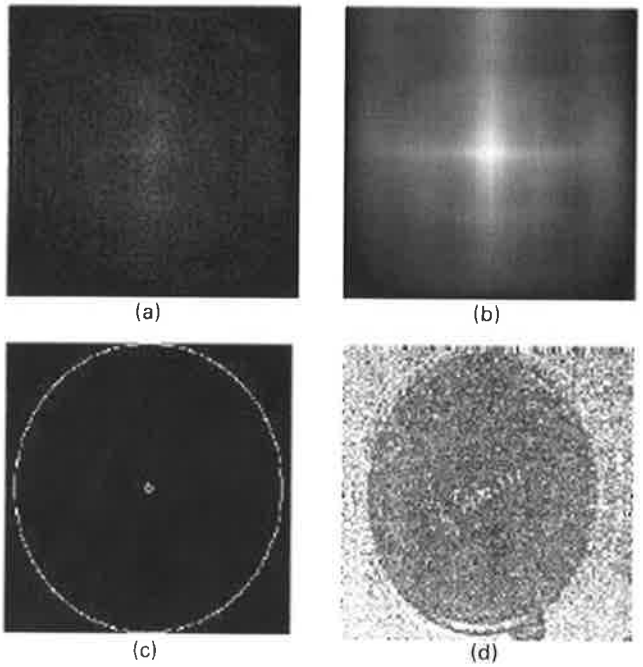

Fig. 20. Experimental results of applying the Tsuji's method and Yuen et al.'s method to detect ellipses in image 3 (Fig. 10(a)): (a) shows the accumulator array for detecting the center after applying Tsuji's method to Fig. 19(a); (b) shows the accumulator array for detecting the center using Yuen et al.'s method; (c) shows the extracted ellipses; (d) shows the detected ellipses superimposed on Fig. 10(a).

Fig. 18 shows the experimental results of Yip et al.'s [5] method for detecting ellipses in an image 1 (Fig. 4(a)). Fig. 18(a) shows the edge-enhanced image of Fig. 4(a) using gradient edge operator with threshold at 10; Fig. 18(b) shows the corresponding two-dimensional array of the accumulator. Fig. 18(c) and (d) shows the threshold begin set to 25 and 50% of the maximum counts of the accumulator array, respectively, for detecting the local peak positions. The large dots in these figures indicate the local false peaks positions. Thus, the Yip et al.'s [5] method fails for this type of images. But Table 1 shows that both Tsuji's [11] method, Yuen et al.'s [12] method as well as proposed method are effective of such images. Fig. 19 shows the experimental results of Yip et al.'s [5] method for detecting ellipses in an image 3 (Fig. 10(a)). Fig. 19(a) shows the edge-enhanced image of Fig. 10(a) using gradient edge operator with threshold at 36; Fig. 19(b) shows the corresponding two-dimensional array of the accumulator. Fig. 19(c) shows the threshold set to 95% of the maximum counts of the accumulator array to detect the local peak positions. The large dots in these figures indicate the local false peaks position which are caused by background votes. Fig. 20 shows the experimental results of Tsuji's [11] method and Yuen et al.'s [12] method to detect ellipses in image 3 (Fig. 10(a)). Fig. 20(a) shows the accumulator array for detecting the center after applying Tsuji's [11] method to Fig. 19(a). The centers of ellipses are shown to move to the end side of the input image. Therefore, Tsuji's [11] method fails for detecting the center. Fig. 20(b) shows the accumulator array for detecting the centers using Yuen et al.'s [12] method. The local peak in the accumulator array which should indicate the center is a false peak. Therefore, Yuen et al.'s [12] method is not capable to find the remaining parameters of ellipses. Fig. 20(c) shows the extracted false ellipses, extracted by Yuen et al.'s [12] method. Fig. 20(d) shows such ellipses superimposed on Fig. 10(a).

5.1. Complexity of the technique

The computational time needed to transform image O to image A is bounded by $O(n^3)$ and that needed to detect edge points is bounded by $O(E_A n)$, where E_A is the number of edge points in A. Process 3 is bounded by $O(E_{f_v} n)$, where E_{f_v} is the number of edge points in f_v. Because E_A and E_{f_v} are at most n^2 and k is less than n, the total computational cost for scanning (rightward) and from top to bottom is bounded by $O(n^3)$. In the same way, the computational time needed for scanning (downward) and from left to right is bounded by $O(n^3)$, and the computational time for estimating the remaining parameters (voting stage) for ellipse is bounded by $O(E_{f_v} n^2)$. On the other hand, the computation cost for the voting stage of the SHT is bounded by $O(E_o n^4)$, which is about two orders of magnitude greater than that the proposed technique. Ballard's [1] method reduces the cost to $O(E_o n^3)$. This is still one order of magnitude greater than that of the proposed technique.

6. Summary and conclusions

The method used the symmetric points to detect the center of an ellipse, then uses the geometrical properties of an ellipse to locate the boundary points of ellipse. This method can also be used to detect shapes, such as rectangles and circles. Furthermore, the proposed technique can be detected partially occluded ellipses. Computer experiments show that the proposed technique is effective and robust. Moreover, the gradient information is not used in the proposed technique. Finally, from Table 1, it has been shown that methods [5,11,12] will be totally ineffective in more complicated image 3. Thus, proposed method is more efficient than these methods and also it is very faster. Because detection of the center of ellipses by finding the lines which are shown proposed method can control that detecting of the center, using a symmetric edge points about l_v or l_h. This idea leads to isolate many background points, which generate a large amount of false peaks. Using edge direction leads to some techniques that are not sensitive to the presence of any part of an instance and the addition of extraneous data, since the effects of noise and extraneous results of the transformation are a problem of concern when dealing with real image data. Moreover, Yip et al.'s [5] method involves many operations of trigonometric functions such as $\cos\theta$, $\sin\theta$ and $\tan^{-1}\theta$.

References

[1] D.H. Ballard, Generalizing the hough transform to detect arbitrary shapes, PR 13 (2) (1981) 111–122.
[2] L. Xu, H. Kalviainen, P. Hirvonon, E. Oja, Probabilistic and non-probabilistic hough transforms: overview and comparisons, IVC 13 (4) (1995) 239–252.
[3] J. Illingworth, J. Kittler, The adaptive hough transform, IEEE Trans. Patt. Anal. Mach. Intell. 10 (1987) 690–698.
[4] J. Illingworth, J. Kittler, A survey of the hough transform, CVGIP 44 (1988) 87–116.
[5] K.S. Tam Peter, K.K. Yip Raymond, N.K. Leung Dennis, Modification of hough transform for circles and ellipses detection using a 2-dimensional array, PR 25 (1992) 1007–1022.
[6] E. Oja, L. Xu, P. Kultanen, A new curve detection method, PR 11 (5) (1990) 257–263.
[7] T.S.L. Lam, S.Y.K. Yuen, D.N.K. Leung, Connective hough transform, IVC 11 (5) (1993) 295–301.
[8] A.A. Sewisy, Detection of lines and curves with hough transform and other methods, Ph.D. Thesis, Department of Mathematics, Faculty of Science, Assiut, Egypt, 1996.
[9] A.A. Sewisy, Detection of circular object with a high speed, in: Proceedings of the 13th International Conference on Industrial and Engineering Application of AI and Expert System (IEA/AIE), New Orleans, USA, June 2000, pp. 522–533.
10 A.A. Sewisy, Graphical techniques for detecting line with hough transform, Int. J. Comput. Math. (2001) (in press).
[11] S. Tsuji, M. Matsumoto, Detection of ellipses by modified hough transform, IEEE Trans. Comput. 27 (1978) 777–781.
[12] J. Illingworth, K. Yuen, J. Kittler, Detecting partially ellipses using the hough transform, IVC 7 (1) (1989) 31–37.