

Non-analytic object recognition using the Hough transform with the matching technique

P.-K. Ser, BEng
W.-C. Siu, CEng, FIEE

Indexing terms: Detection, Transformers

Abstract: By using the combination of the Hough and the contour sequence matching techniques, we suggest a new algorithm of scale and rotation invariant for pattern recognition. In our proposed algorithm, the conventional four-dimensional Hough space is replaced by a two-dimensional one. Because of the significant reduction of memory requirement, we can provide a much faster and more efficient generalised Hough algorithm. In order to enhance the performance of the Hough transform, we also suggest a new peak searching technique on the Hough space to achieve an accurate location of the detected objects.

1 Introduction

In image recognition analysis, researchers are often interested in recognising objects from images. The robust Hough transform is one of the most useful algorithms for the detection of analytic objects. Research workers [1, 2] have provided successful applications of the Hough techniques on the recognition of lines and curves in pictures. A modification of the Hough approaches [3, 4] has been exploited for the detection of simple geometrical objects such as circles and ellipses. However, it is not easy to directly apply Hough techniques in detecting non-analytic objects. Merlin and Farber [5] and Sklansky [6] made some preliminary studies on the utilisation of the Hough techniques to detect arbitrary objects. Some matching techniques for curve detection are suggested in References 5 and 6.

Ballard [7] suggested a generalised Hough algorithm for pattern recognition. This generalised Hough transform (GHT) uses the edge operator such as Sobel to determine the gradient direction of each boundary pixel. According to the gradient direction of the pixel, the relative vector between the reference position of the object to the corresponding pixel is stored in the R-table. The cost for the detection of different variations of the object is to increase the number of dimensions of the corresponding Hough parameter space. Each unknown parameter in the image space is represented by a special dimension in the Hough space. Nevertheless, the critical dilemma of the application of the Hough method is the requirement of

enormous dimensions in the Hough space for the detection of different characteristics of an object.

Many algorithms have been suggested to improve the capabilities of the generalised Hough transform. Davies [8–10] modified the GHT algorithm to detect curved objects. For the analysis of low eccentricity ellipses in Reference 8, a point spread function (PSF) is accumulated for each edge pixel with all possible orientations in a single Hough parameter plane. The scheme is able to reduce the Hough domain to be an $N_x * N_y$ two-dimensional plane but it might cause spurious peaks in the single plane. A similar situation described in Reference 10, where the approach for polygon detection is to move each edge pixel with a constant distance along the gradient direction. The variation of the orientation of a polygon does not affect the corresponding peak in the Hough space. For the case of regular polygons, a two-dimensional space is capable of detecting the positions of the polygons. Kang *et al.* [11] proposed the GHT method for the extraction of straight line segments. The advantage of the suggested rotation Hough transform is that it is suitable for the determination of the lengths of straight lines in an image.

A modification of the construction of PSF is suggested by Thomas [12]. According to the proposed compressing technique, the displacement vector forming the PSF is relative to the gradient direction at that pixel. However, the inherent problem of the extra process on the accumulation of a series of votes in the parameter space has not been resolved.

Different techniques for the detection of invariance of objects are suggested in References 13 and 14. Ma and Chen [13] suggested the use of local properties which include slopes and curvatures to determine the scale and the orientation of an object. Jeng and Tsai [14] proposed an alternative approach to tackle this problem. The strategy of accumulation for their orientation invariant algorithm is to use a circle instead of a position vector. This circular increment approach requires an extra computation time on the accumulation stage.

The problem of peak searching of the Hough space present another difficulty to achieving a satisfactory result. In fact, it is an important process in verifying the performance of an algorithm. Van and Groen [15] provided an analysis of the voting scheme of line detection from the Hough space. The converging squares algorithm [16] defined the peak as the point of maximum value within the region of maximum density. However, the situation becomes complex as there are multiplexes in the Hough space for the detection of several objects in an image.

In this paper, we suggest a technique to tackle the problems of heavy computation and large memory

© IEE, 1994

Paper 9876E (E4, E5), first received 27th October 1992 and in revised form 17th May 1993

The authors are with the Department of Electronic Engineering, Hong Kong Polytechnic, Hung Hom, Kowloon, Hong Kong

IEE Proc.-Comput. Digit. Tech., Vol. 141, No. 1, January 1994

11

requirement of the generalised Hough transform (GHT). The reduction of two dimensions in the Hough space simplifies the complexity of the detection significantly. Furthermore, the properties of the new technique are scale and orientation invariant for the recognition of close object. We can consider our recognition algorithm to merge the advantages of the Hough techniques and that of contour sequence matching approach. An accurate process of peak searching in the Hough space is also suggested on the final stage of the Hough algorithm.

2 The process of accumulation in the Hough space

For our suggested algorithm, the procedure of recognition can be divided into four sections. Each of the sections are described in this paper. They are

- (i) construction of the R-table
- (ii) setting the array of the scaling parameter
- (iii) accumulation of the Hough space
- (iv) contour matching and peak detection in the Hough space.

2.1 Construction of the R-table

The construction of the standard R-table depends on the gradient directions of the corresponding edge pixels. Each edge pixel in an image retrieves the position vector from the table according to its gradient direction. For a rotated object, the gradient of its boundary has to set an offset in order to match the same group of position vectors from the R-table.

Eqn. 1 shows the relationship between the prototype and the corresponding scaled and rotated objects in Fig. 1. To determine the corresponding reference positions of the objects, the position vector of each edge pixel has to be changed according to the invariance of the detected objects. In Fig. 1, we will classify each object with a class number i . The class number is assigned for the identification of the scaling parameter of the corresponding enclosed contour.

$$\begin{bmatrix} x \\ y \end{bmatrix} = k * \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (1)$$

where k is the corresponding scaling parameter of an object to that of the prototype, θ is the relative orientation of an object with that of the prototype and (u_0, v_0) is the relative translation vector of an object and prototype.

From the construction of the modified R-table, the vector P_1 to A is stored as an index according to the difference of gradient directions $(\theta_1 - \theta_2)$ between the pair of edge pixels P_1 and P_2 . The relative gradient direction of edge pair P_1 and P_3 gives another index to store the same position vector in the R-table. The pixel length between P_1 and P_2 is equal to that of P_1 and P_3 . In order to simplify the explanation of the proposed technique, we concentrate our analysis on the edge pair (P_1, P_2) in the following studies. In fact the whole process of recognition is applicable to both pairs of edge pixels. In the first part of the R-table, it consists of the position vector, R , and the sum of gradient angles for each entry. The sum of gradient angles $(\theta_1 + \theta_2)$ stored in the R-table is for the evaluation of the relative angle of orientation of the object to be detected in the image compared to the prototype shape (see eqn. 2 below). As shown in the following equation, S_i stores the position vector and the sum of gradient angles of a pair of edge pixels using the relative gradient angle, i , as the index. As two adjacent pixels with an appropriated distance from each of

the edge pixel in the contour are considered, the total number of edge pairs is equal to $2 * \text{number of edge pixels}$.

$$\begin{array}{ll} \text{Different of gradient} & \text{Position vector } R \text{ and the} \\ \text{angle} & \text{sum of gradient angle} \\ (\theta_1 - \theta_2) & (\theta_1 + \theta_2) \\ i & S_i = \{(R, \theta_1 + \theta_2) | \theta_1 - \theta_2 = i\} \end{array}$$

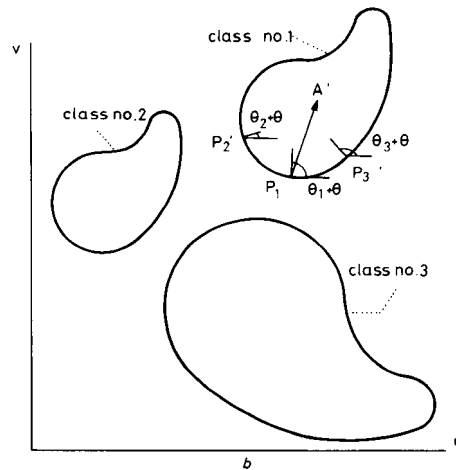
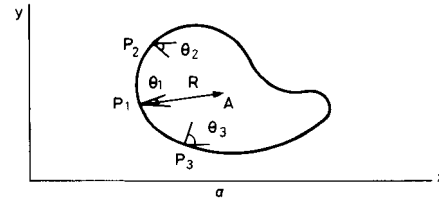


Fig. 1
a Original object for the construction of R-table
b Objects with different orientations and scaling factors

In our proposed construction of the R-table, additional data are included so as to perform the detection of objects with different variants in the image. The R-table consists of three main parts. They are

- (i) the corresponding set of S_i for each edge pixel to the reference point A
- (ii) the length of the perimeter of the prototype, l_{pro}
- (iii) the contour sequence shown in Fig. 2a of the prototype.

2.2 Setting the array of scaling parameter

As shown in Fig. 1b, there are three objects with different orientations and scales. Before finding the relative rotation angle θ of each object to the prototype in Fig. 1a, the scaling factor of each enclosed contours is determined. Let the perimeters of an object with class number i and that of the prototype be l_i and l_{pro} , respectively. We can determine the scaling parameter k_i of the object i by taking the ratio of l_i and that of the prototype l_{pro} .

$$\begin{array}{ll} \text{Classification of object} & \text{Scaling factor, } k \\ i & k_i = l_i / l_{pro} \end{array}$$

For the determination of the scaling parameter of an image, we can trace the contour which passes through the object. Once the perimeter is defined, the scaling

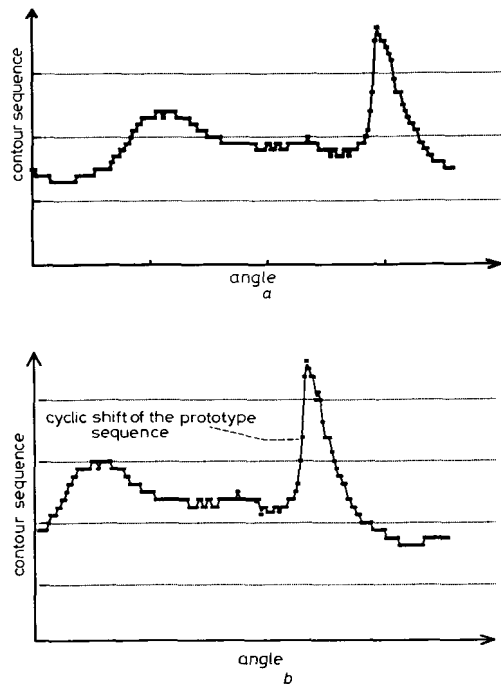


Fig. 2
 a Contour sequence of the prototype object with reference point at A
 b Contour sequence of a rotated and scaled object

factor is stored into an array with the class number of the contour as the index. Hence pixels lying on the contour are assigned with the same scaling factor, the scaling parameter k is retrieved from the array. We make use of the array storage to reduce the duplication of scaling computation of pixels lying on the same curve.

2.3 Accumulation of the Hough space

For the conventional GHT algorithm, the relative angle of rotation is obtained from the Hough space. Hence, possible reference positions of objects on Hough planes with different orientations will be incremented by the scaled position vector for each pixel. In our algorithm, the parameter of orientation θ is found by using the relative angle approach. The gradient angle θ_i of an edge pixel P_i is applied to detect the orientation of the corresponding pixel.

Considering the object with class number 1 as shown in Fig. 1(b), it is rotated with an angle θ degree with respect to the prototype as shown in Fig. 1(a). The edge pixels P'_i and A' of the object are the corresponding pixels of P_i and A in Fig. 1 and the gradient angles of P'_1 , P'_2 and P'_3 are $\theta_1 + \theta$, $\theta_2 + \theta$ and $\theta_3 + \theta$, respectively. It is seen that the relative difference in angles of the pixel pair (P_1, P_2) is the same as that of (P'_1, P'_2) . Both pairs of edge pixels (P_1, P_3) and (P'_1, P'_3) have similar conditions. Comparing Figs. 1(a) and 1(b), it is found that the relative position of $P'_1 A'$ is equal to the vector $P_1 A$ with an angle of rotation of θ degree. The problem of rotation variant will then be resolved, if the corresponding angle θ is found. In fact, the angle of rotation of the vector R is

exactly equal to the angle of rotation of the boundary with respect of the prototype object.

In order to maintain the consistency of the relative angle of each pair of edge pixels, the corresponding separation between P'_1 and P'_2 of the scaled object has to be scaled to have the same separation of P_1 and P_2 with the parameter k_i . Hence, the relative angle of P_1 and P_2 is the same as that of P'_1 and P'_2 . From the pre-defined R-table, the position vector from P_1 to the reference position A , is retrieved back according to the index of the relative angle of the corresponding edge pair. By using an edge operator, the gradient directions of P'_1 and P'_2 are found whereas the corresponding S_i for this edge pair can then be retrieved from the R-table. Note that i is determined as follows

$$\begin{aligned} i &= \text{gradient of } P'_1 - \text{gradient of } P'_2 \\ &= (\theta_1 + \theta) - (\theta_2 + \theta) \\ &= \theta_1 - \theta_2 \end{aligned}$$

Note also that P'_1 makes us possible to access the set $S_{(\theta_1 - \theta_2)}$ which stores the data for P_1 to A . Since the angle of rotation of the object is equal to the relative angle of rotation between the edge pixels P'_1 and P_1 , we can determine the relative angle of rotation between P_1 and P'_1 with the application of the sum of gradient angle. $(\theta_1 + \theta_2)$ stored in the S_i can now be applied to eqn. 2

$$\begin{aligned} \text{Angle of rotation} &= (\text{gradient of } P'_1 + \text{gradient of } P'_2 - (\theta_1 + \theta_2))/2 \\ &= ((\theta_1 + \theta) + (\theta_2 + \theta) - (\theta_1 + \theta_2))/2 \\ &= \theta \end{aligned} \quad (2)$$

where the sum of $\theta_1 + \theta_2$ can be retrieved from the set S_i .

As the angle of rotation and the scaling parameter of the object in question are obtained, the corresponding reference position A' of the image edge pixel P'_1 in the Hough space can be defined as follows

$$A' = P'_1 + k_i * \text{vector } R \text{ with a rotation of } \theta \text{ degree}$$

In the Hough domain, the cell which represents the location A' of the object is increased. The two-dimensional Hough space only indicates the location of the object in the image space. The contour sequence stored in the R-table is applied to determine the orientation of the detected object in an image and to eliminate false peaks from a cluster of possible locations of the object in the Hough space.

2.4 Peak searching process

Peak searching of the Hough space is an inherent problem for the application of the Hough transform. In our peak searching algorithm, we suggest a generalised approach to find an optimum position from a cluster of noisy peaks in the Hough domain to represent the parameters of the object. Note that this generalised peak searching approach is also applicable for other GHT algorithms as a method to verify their performance.

In the Hough space, the peak searching process is a difficult problem to be resolved in the final stage of the algorithm. As shown in the Fig. 3, a cluster of peaks is usually formed in the Hough space to represent the possible locations of object 1. The conventional approach to eliminate false peaks around the cluster of peaks is to find the maximum within the cluster of peaks. However, the effect of this method is usually to filter the cluster of peaks to become many nearby local maximum peaks as shown in Fig. 3b. Since the difference in values of cells in local clusters is usually small, the method of setting a

threshold to further eliminate the false peaks is not suitable in this situation. In this paper, we suggest a further process which is based on some observations of local maximum techniques.

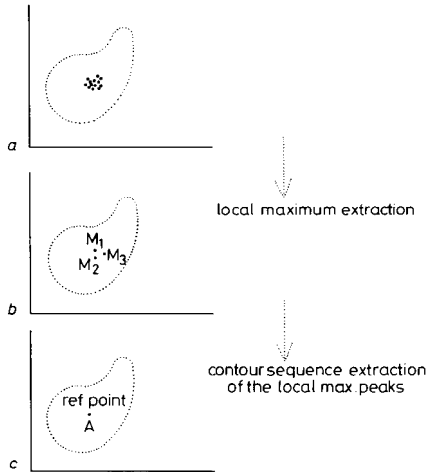


Fig. 3 Processes on peak searching

Because of the error of digital quantisation in an image, local peaks in the Hough space may not be easily used to locate an actual object in the image space. We select an optimum location M_{iopt} to locate the corresponding object instead of the conventional selection of using the original local maximum peak M_i as shown in Fig. 4. The criterion for the selection of the optimum

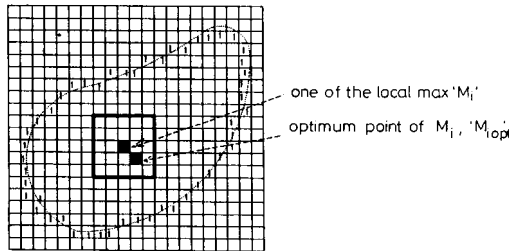


Fig. 4 Location of the optimum point of the object with class no. 1

point does not constrain it to its cell value. In order to determine the location of the optimum position, a 5×5 window of cells with M_i as the centre is extracted from the Hough space. The error function in eqn. 3 (see below) of the 25 neighbour cells will be calculated and stored for further process. M_{iopt} is selected in such a way that it possesses the minimum contour error among the 25 cells in the square window. Similarly, other local maximum peaks will be replaced by their optimum points with the same approach. The cluster of peaks (M_1, M_2, \dots) is replaced by the cluster of $(M_{1opt}, M_{2opt}, \dots)$. By comparing $(M_{1opt}, M_{2opt}, \dots)$, the best location representing the reference position, A , of the detected object is at a location corresponding to the minimum of $(M_{1opt}, M_{2opt}, \dots)$.

In other words, the process of peak searching in the Hough space consists of two stages. First, local maximum peaks (M_1, M_2, \dots) are extracted to indicate the approximate locations of suspected objects. For the fine filtering of false peaks process, a comparison of error functions of the optimum positions $(M_{1opt}, M_{2opt}, \dots)$ is exploited to select the desired locations.

3 Determination of the orientation and scale of the candidate objects

During the process of peak searching, the parameters of orientation and scale are extracted at the same time. The contour sequences of the detected object with reference to possible local maximum peaks are used to compare with that of the pre-defined sequence in the third part of the R-table. Referring to Fig. 2b, $C_1(n)$ is the contour sequence of object 1. It is known that $C_1(n)$ is a scaled cyclic shift version of the prototype sequence $C_{pro}(n)$. The shift between these sequences is actually equal to the angle of orientation of the object.

$$C_{obj1}(n) = k \times (\text{cyclic shift of the sequence } C_{pro}(n) \text{ with } \theta \text{ degree})$$

$$\text{Error}_{obj1}(\phi) = \sum_{n=0^\circ}^{360^\circ} |C_{obj1}(n) - k \times (\text{cyclic shift of } C_{pro}(n) \text{ with } \phi \text{ degree})| \quad (3)$$

From eqn. 3 it is found that the $\text{Error}_{obj1}(\phi)$ is the minimum as ϕ is equal to θ . The desired orientation of the object is to be detected by searching the minimum of the contour error function, $\text{Error}_{obj1}(\phi)$.

From the peaks indicated in the Hough space, the scaling parameter for the detected object is found according to the class of the object. As extracting from the contour profile in the above process, the class of the contour pixel is stored in order to indicate the corresponding scaling parameter, k_i . Hence, to get the class number of the corresponding object from the image, we can directly retrieve the scaling parameter k from the array for the suspected object.

4 Comparison of our proposed GHT and the standard approach

A comparison of our suggested algorithm and the standard approach is shown in Table 1, where we compare

Table 1: Complexity of the proposed GHT and the conventional GHT algorithms

	Proposed GHT algorithm	Conventional GHT algorithm
Memory requirement	$N_x \times N_y$	$N_x \times N_y \times N_{scale} \times N_{ang}$
Order of processing time	$O(n)$	$O(n) \times N_{scale} \times N_{ang}$
Computation effort on peak searching (condition for local maximum peaks)	Two-dimensional Hough space	Four-dimensional Hough space

N_x, N_y = resolution of the image space in x -axis and y -axis, respectively
 N_{scale}, N_{ang} = resolution of the scale and orientation parameters, respectively
 n = number of edge pixels in the image

the efficiency and the memory requirement of the approaches to detect objects. As we have mentioned, one of the advantages of the proposed algorithm is to replace the four-dimensional Hough space with a two-dimensional one. Furthermore, the order of computation effort of the recognition process is independent of the resolution of N_{scale} and N_{ang} .

5 Results

In Figs. 5 to 7, the results of image recognition of the proposed algorithm are shown. The Figures are divided into four parts:

- (i) upper left corner: original image

- (ii) upper right corner: binary level of the original image
- (iii) lower left corner: corresponding location of the detected object
- (iv) lower right corner: local maximum peaks in the Hough space



Fig. 5 Result of the proposed Hough algorithm to detect 'key 256'



Fig. 6 Result of the proposed Hough algorithm to detect 'screw 256'

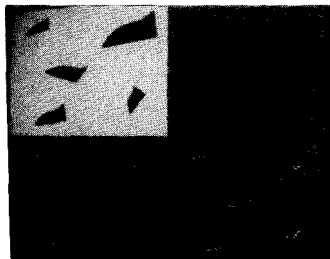


Fig. 7 Result of the proposed Hough algorithm to detect 'irregular 256'

The peaks of local maximum in the Hough space are shown in part of each Figure. It is found that the cluster of peaks is around the reference positions of the detected objects. We applied our suggested peak searching algorithm to extract the best position from the peaks. In part (iii) of each Figure, the cross marks represent the best reference positions of the detected objects.

We divide the processing time of our recognition algorithm into two parts (see Table 2). The computation times for the Hough transform and the peak searching process are counted independently.

The Hough processing time is relatively shorter than that for peak searching. The computation of the Hough algorithm mainly depends on the number of edge pixels.

Table 2: Processing time of our suggested Hough algorithm for objects of different shapes

Image	Hough processing time, s	Peak searching processing time, s	No. of local maximum peaks
key 256	6.3	5.3	1
screw 256	1.9	82.9	14
irregular 256	7.5	103.6	20

When we have a larger number of boundary pixels in the images, the processing time is increased. For the comparison of the conventional GHT algorithm, we can expect that its processing time will be much longer than that of our proposed one. Taking N_{scale} and N_{ang} to be 15 and 180, respectively, we have found that the approximate computation effort of the conventional approach will be about 2700 times longer than that of our algorithm. The fourth column of the Table shows the number of local peaks from the Hough space. It is found that the peak searching processing time is proportional to the number of the local maximum peaks in the Hough space. Apparently, a possible method to reduce the searching time is to increase the threshold for local peaks, but the undesired loss of information of the Hough space may become a severe cost for this high threshold approach.

6 Conclusion

In this paper, we have suggested combining the merits of the Hough transform and the contour sequence techniques for the recognition of non-analytic objects. Our modified GHT algorithm becomes a rotation and scale invariant approach. Heavy burden on memory space of the GHT is one of the main obstacles for the application of the technique to pattern recognition. Referring to the conventional generalised Hough transform, we can replace a four-dimensional memory requirement with a two-dimensional one. The reduction of Hough space not only saves the cost of implementation but also significantly simplifies the peak searching process.

For the conventional Hough approach, the peaks in the Hough space give the expected locations of the detected objects. However, false peaks in Hough algorithm often causes great trouble and produces spurious results. Besides the investigation on the Hough transform, we have also proposed a more accurate peak searching technique for the detection of the peaks in the Hough space. We also define the reference position of an object based on the minimum of the contour error function of the local maximum peaks cluster. In fact, the optimum locations of objects may not lie on the peak positions in the Hough space. The approach is capable of eliminating a log of false peaks from the optimum locations of the objects.

For the present investigation, we mainly concentrate on images with reasonably good contours. Our experimental work shows that the approach is robust under a Gaussian noise environment. For images with some broken boundaries, we might have to do some pre-processing work to link the broken lines to form complete contours. However, if there are long broken lines or for occluded objects, some further rearrangement of our scheme might be necessary. In this case, the scaling parameter k might be incorrectly estimated. One remedy is to make the voting process in the Hough domain for a range of values of k . This will undoubtedly increase the amount of processing time. However, the size of the problem still roughly remains as two dimensional and

application of the approach could possibly be extended to the detection of occluded objects. This gives a fruitful direction for further research.

7 References

- 1 HOUGH, V.C.: 'Method and means for recognizing complex patterns'. US Patent 3069654, 1962
- 2 DUDA, R.O., and HART, P.E.: 'Use of the Hough transform to detect lines and curves in pictures', *Commun. Ass. Comput. Mach.*, 1972, **15**, pp. 11-15
- 3 YE, Q.Z.: 'A preprocessing method for Hough transform to detect circles'. Proc. IEEE Conf. on Computer Vision and Pattern Recognition, 1986, pp. 652-653
- 4 CHAN, R., and SIU, W.C.: 'A new approach for efficient Hough transform for circles', *IEE Proc. E*, 1991, **138**, (5), pp. 335-344
- 5 MERLIN, P.M., and FARBER, D.J.: 'A parallel mechanism for detecting curves in pictures', *IEEE Trans.*, 1975, **C-24**, pp. 96-98
- 6 SKLANSKY, J.: 'On the Hough technique for curve detection', *IEEE Trans.*, 1978, **C-27**, (10), pp. 923-926
- 7 BALLARD, D.H.: 'Generalizing the Hough transform to detect arbitrary shapes', *Pattern Recognition*, 1981, **13**, (2), pp. 111-122
- 8 DAVIES, E.R.: 'Finding ellipses using the generalised Hough transform', *Pattern Recognition Letters*, 1989, **9**, pp. 87-96
- 9 DAVIES, E.R.: 'Application of the generalised Hough transform to corner detection', *IEE Proc. E*, 1988, **135**, (1), pp. 49-54
- 10 DAVIES, E.R.: 'Reduced parameter spaces for polygon detection using the generalised Hough transform'. Proc. 8th Internat. Conf. on Pattern Recognition, 1988, pp. 495-497
- 11 KANG, C.W., PARK, R.H., and LEE, K.H.: 'Extraction of straight line segments using rotation transformation: generalised Hough transformation', *Pattern Recognition*, 1991, **24**, (7), pp. 633-641
- 12 THOMAS, A.D.H.: 'Compressing the parameter space of the generalised Hough transform', *Pattern Recognition Letters*, 1992, **13**, pp. 107-112
- 13 MA, S.D., and CHEN, X.: 'Hough transform using slope and curvature as local properties to detect arbitrary 2D shapes'. 9th Internat. Conf. on Pattern Recognition, 1988, Rome, Italy, pp. 511-513
- 14 JENG, S.C., and TSAI, W.H.: 'Scale and orientation-invariant generalised Hough transform — a new approach', *Pattern Recognition*, 1991, **24**, (11), pp. 1037-1051
- 15 VAN VEEN, T.M., and GROEN, F.C.A.: 'Discretization errors in the Hough transform', *Pattern Recognition*, 1981, **14**, pp. 137-145
- 16 O'GORMAN, L., and SANDERSON, A.C.: 'The converging squares algorithm: an efficient multidimensional peak picking method'. Proc. IEEE Internat. Conf. on Acoustics, Speech, and Signal Processing, 1983, Boston, USA, pp. 112-115
- 17 GUPTA, L., and SRINATH, M.D.: 'Contour sequence moments for the classification of closed planar shapes', *Pattern Recognition*, 1987, **20**, pp. 267-272