Pattern Recognition, Vol. 24, No. 11, pp. 1037–1051, 1991
Printed in Great Britain

0031-3203/91 $3.00 + .00
Pergamon Press plc

SCALE- AND ORIENTATION-INVARIANT GENERALIZED HOUGH TRANSFORM—A NEW APPROACH

SHENG-CHING JENG*† and WEN-HSIANG TSAI‡§

*Institute of Computer Science and Information Engineering; ‡Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan 30050, Republic of China

(*Received* 31 *August* 1990; *received for publication* 1 *May* 1991)

Abstract—The conventional generalized Hough transform (GHT) is useful for detecting or locating translated 2-dimensional (2D) object shapes. However, a weakness of the conventional GHT is that a brute force approach is usually required to handle shape scaling and rotation, resulting in the use of a 4D Hough counting space (HCS). A new version of the GHT, called scale- and orientation-invariant GHT (SOIGHT), is proposed to remove this weakness. The improvement is based on the use of half lines and circles to replace the displacement vectors used in the conventional GHT for cell value incrementation. The required dimensionality of the HCS for the SOIGHT is reduced to 2D so that the storage and computation requirements for cell value incrementation and maximum detection in the HCS can be reduced effectively. Some experimental results are included to demonstrate the applicability of the proposed SOIGHT.

Generalized Hough transform   Hough counting space   Cell value incrementation
Point spread function   Scale and rotation invariant   Shapes detection and location

1. INTRODUCTION

The Hough transform is useful for detecting or locating straight lines or analytic curves.[1–3] In reference (4), a generalized version of the Hough transform, called the generalized Hough transform (GHT), was proposed for detecting arbitrary 2-dimensional (2D) object shapes. It has been used in many applications of computer vision, such as shape detection and recognition, image registration, etc.

The GHT is good for detecting 2D object shapes with specific orientations and scales. In the GHT process, a 2D Hough counting space (HCS) consisting of an array of cells is required for cell value incrementation and maximum cell value detection. However, when the orientation and scale of an input shape is variant and unknown in advance, brute force is usually employed to enumerate all possible orientations and scales of the input shape in the GHT process. This requires two more dimensions for the HCS, resulting in the use of a 4D HCS and so the requirements of excessive storage and long computation time.

To overcome these problems, Davis[5] proposed a line-segment based GHT in which line segments instead of edge points are employed as the basic processing units in the GHT. In reference (6), line segments are also used as the basic units for multiclass object recognition. In a similar way, local classification of the instances of detected contours is performed before the GHT process in reference (7). The implementations of these approaches are complicated since local classifications of subpatterns are not easy. It is also difficult to obtain the desired subpatterns in an input image accurately, especially when noise is present. When the segmentation of the extracted subpatterns is poor, the object shapes cannot be detected or located correctly.

In this paper, two new cell value incrementation strategies, one scale-invariant (SI) and the other orientation-invariant (OI), are proposed first to overcome the problems caused by the scale and the orientation variations of input shapes. And by a combination of these two incrementation strategies, a scale- and orientation-invariant (SOI) cell value incrementation strategy which can be used to handle both scale and orientation variations is proposed. By using these incrementation strategies, the brute force approach to handling the scale and orientation variations of input shapes can be avoided. Only two parameters of the x- and y-translations are required in the new strategies so that the HCS is reduced to 2D. The storage and computation requirements can thus be reduced effectively. Furthermore, for the proposed incrementation strategies, point spread functions (PSFs)[8,9] are employed to replace the reference tables used in the conventional GHT. The

† Sheng-Ching Jeng is also with Advanced Technology Center, Computer & Communication Research Laboratories, Industrial Technology Research Institute, Chutung, Hsinchu, Taiwan 31015, Republic of China.

§ Author to whom correspondence should be addressed.

use of the PSF can also shorten the amount of processing time for cell value incrementation.

More specifically, in the proposed SI cell value incrementation strategy, instead of incrementing the values of the cells which are pointed to by the displacement vectors of the reference table as in the conventional GHT, the value of each cell passed through by the half line along a displacement vector is incremented. And in the proposed OI cell value incrementation strategy, circles instead of half lines are used for overcoming the effects caused by changes of input shape orientations. Finally, the proposed SOI cell value incrementation strategy includes two stages. First, the SI version of the PSF (SI-PSF) is generated. Next, by applying the OI cell value incrementation strategy to each of the nonzero votes in the SI-PSF, the SI-PSF can be transformed into the desired SOI-PSF.

By using the proposed SOI cell value incrementation strategy, a new version of the GHT, called scale- and orientation-invariant GHT (SOIGHT), is proposed. Since the required HCS is reduced to 2D, the computation requirement of the maximum cell value detection step in the SOIGHT can also be reduced. In addition, for each edge point in an input image, the maximum amount of required processing time for cell value incrementation operations is proportional to the size of the input image and is less than that of the conventional GHT. Thus the total computation complexity of the proposed SOIGHT is reduced greatly when compared with that of the conventional GHT.

Like the conventional GHT, the location of the cell with the maximum value in the HCS can be found out by the proposed SOIGHT as the desired location of the object shape. This location result is enough for use in certain shape detection applications. However, the scale and orientation of the detected object shape cannot be found directly by the proposed SOI cell value incrementation strategy. For cases where the shape scale and orientation are also desired, one method of solving this problem is also proposed.

This paper is organized as follows. In Section 2, the conventional GHT is reviewed. In Section 3, the proposed SI, OI and SOI cell value incrementation strategies are described. Section 4 includes the proposed SOIGHT. Experimental results are given in Section 5. Concluding remarks are given in Section 6.

2. A REVIEW ON CONVENTIONAL GHT

2.1. *Hough counting space*

To use the conventional GHT for detecting an arbitrary object shape (also called a template shape in the sequel) in an input image, it is necessary to set up a 4D HCS $H(X_t, Y_t, S, \Theta)$, where (X_t, Y_t) is a translation vector with respect to a reference point of the object shape and describes the location of the object shape in the input image, S describes the scale factor of the shape, and Θ specifies its orientation. In the proposed SOIGHT, only a 2D HCS $T(\mathbf{X}_t, \mathbf{Y}_t)$ is required. In this study, the centroid of an object shape is always used as the reference point of the shape.

Generally, an array of cells constitutes the HCS, in which each cell has a value specifying the possibility that the reference point of the object shape to be detected is located at the cell. When there exists a cell whose value exceeds a certain threshold and is the maximum in the HCS, then the object shape is said to be detected at the location of the cell.

2.2. *Reference tables*

To perform the GHT, a reference table for the template shape is built up in advance. For this, a basic reference table for the template shape for a specific scale $\bar{S}$ and a specific orientation $\bar{\Theta}$ is built up first by the following three steps: (1) select a reference point R in the given template shape; (2) rotate the shape through 180° with respect to R; (3) trace all the boundary points of the template shape and construct the basic reference table to consist of the displacement vectors between all the boundary points and point R, with each displacement vector represented by its length r and direction θ, augmented by the specific scale $\bar{S}$ and orientation $\bar{\Theta}$ of the template shape. Then, the desired reference table is constructed to consist of the basic reference tables for all the possible template scales and orientations.

Let $\#(S)$ denote the number of all the possible quantized scales of the given template shape, and $\#(\Theta)$ the number of the quantized directions of the shape around 360°. Also, let $\#(\mathrm{BRT})$ denote the size of the basic reference table. Then, the size of the reference table of the given template is obviously equal to $\#(S) \times \#(\Theta) \times \#(\mathrm{BRT})$. The magnitudes of the $\#(S)$ and $\#(\theta)$ values are determined by the required accuracy of object shape detection. The range of the shape scale is constrained by the scales of all possible shapes in input images.

2.3. *Conventional cell value incrementation strategy*

The cell value incrementation strategy is the kernel of the GHT and specifies a method to increment the cell values of the HCS. A common cell value incrementation strategy for the conventional GHT is to superimpose all the displacement vectors in the reference table on each edge point in the input image and increment by one each of the values of those cells which are pointed to by the displacement vectors. That is,

for each edge point (x_e, y_e),
for each scale $\bar{S}$,
for each orientation $\bar{\Theta}$, and
for each displacement vector (r, θ),
compute $(x, y) = (x_e, y_e) + (r\cos\theta, r\sin\theta)$, and
set $H(x, y, \bar{S}, \bar{\Theta}) = H(x, y, \bar{S}, \bar{\Theta}) + 1$. (1)

This incrementation strategy handles the scale and orientation variations of input shapes by brute force.

2.4. *Conventional GHT for detecting a scaled and rotated object shape*

In the following, we describe the conventional GHT algorithm using brute force[10] to detect or locate a given object shape with its scale and orientation different from the reference shape.

Algorithm 1. Conventional GHT using brute force.

Input. A given template shape and an input image containing an object shape which is a scaled and rotated version of the template.

Output. The location in the input image where the object shape appears, together with the scale and orientation of the object shape with respect to the template shape.

Steps.

(1) Form a 4D HCS $H(X_t, Y_t, S, \Theta)$ and set all the values of the cells in H to zero.

(2) Trace all the edge points in the input image.

(3) Cell value incrementation: increment the values of the cells in $H(X_t, Y_t, S, \Theta)$ according to the cell value incrementation strategy specified by (1).

(4) Maximum cell value detection: find out the location of the cell with its value exceeding a threshold t and being the maximum in H.

(5) Exit with the corresponding image location, scale and orientation as the output.

2.5. *Complexity analysis of the conventional GHT*

In the above conventional GHT, a 4D HCS is required, and the computational complexity is focused on Steps (3) and (4). In Step (3), for each edge point the amount of processing time for cell value incrementation is proportional to the size of the reference table of the given template. Let $\#(E)$ denote the number of edge points in the input image. Then the total amount of processing time for cell value incrementation is proportional to $\#(E) \times \#(S) \times \#(\Theta) \times \#(\text{BRT})$. On the other hand, the process of maximum cell value detection in Step (4) is performed in the 4D HCS and so is time consuming.

2.6. *Point spread function*

In the GHT, each cell value in the HCS serves as evidence for the existence of the template shape at the corresponding location in the input image, and is contributed by a set of edge points of the object shape in the input image. To measure such a type of evidence, the notion of PSF[8,9] is employed in this study. The PSF, whose function is similar to the above-mentioned reference table and whose space is of the same size as that of the HCS, can be used for incrementing the cell values in the HCS. It is also constructed in advance to shorten the amount of processing time of the GHT.

For the case of the conventional GHT, the PSF is constructed by superimposing all the displacement vectors of the reference table on the central point P_c of the PSF (for convenience we regard P_c as corresponding to the reference point of the object shape) and collecting all the vector ends falling at each location in the space of the PSF as the vote value of the location. Let $V_{\text{PSF}}(x, y, \bar{S}, \bar{\Theta})$ be defined as the vote value at coordinates (x, y) of the PSF for the scale $\bar{S}$ and orientation $\bar{\Theta}$ of the template. Let (x_c, y_c) denote the coordinates of point P_c. Then, V_{PSF} for the conventional GHT can be described by:

$$V_{\text{PSF}}(x, y, \bar{S}, \bar{\Theta}) = \begin{cases} 1, & \text{if } (x, y) = (x_c, y_c) + (r_i\cos\theta_i, r_i\sin\theta_i) \text{ for some } i = 1, 2, \ldots, k, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where r_i and θ_i are the same as those in (1) and k is the total number of the displacement vectors in the reference table. As a result, every non-zero vote value is equal to one in the resulting PSF for the conventional GHT. Construction of the PSF for the proposed SOIGHT is more complicated and will be described in the next section.

In the mean time, a PSF table (PSFT) which corresponds to the PSF is also constructed in this study to provide the information for determining the scale and orientation of the detected object shape in the proposed SOIGHT. The details will be described in Section 4. In the PSFT, the sources of the votes which are contributed by different orientations and scales of the template are recorded. Specifically, associated with each non-zero vote value V_{nz} in the PSF, there is a list of vote sources stored as a set of scale-orientation pairs $\{(S_i, \Theta_i) | i = 1, 2, \ldots, V_{nz}\}$ in the PSFT. For the case of the conventional GHT, for each non-zero vote $V_{nz} = 1$ of the V_{PSF} in (2), the associated list in the PSFT is just the scale-orientation pair $(\bar{S}, \bar{\Theta})$ in (2).

To use the PSF in the cell value incrementation step, for each edge point P_e in the input image, let its corresponding cell in the HCS be denoted as C_e. Then, for each cell C in the HCS, if there exists a cell P in the space of the PSF with a non-zero vote value V_{nz} such that the displacement vector from P to the center P_c of the PSF is equal to the displacement vector from C_e to C, then the value of cell C in the HCS is incremented by the value V_{nz}. Let (x_e, y_e), (x, y) and (x_c, y_c) denote the coordinates of the cell C_e, cell C and point P_c, respectively. Then the displacement from cell C_e to cell C is $(x_e - x, y_e - y)$, so that the coordinates of cell P in the space of the PSF whose non-zero vote value is used to increment the value of cell C are just $((x_e - x) + x_c, (y_e - y) + y_c)$. Let EP denote an edge point marix with its size equal to that of the input image, in which each element $EP(x', y')$ indicates whether an edge point exists at the corresponding location (x', y') in the input image: *if* an edge point exists at location (x', y'), *then* $EP(x', y') = 1$; *else* $EP(x', y') = 0$.

Then for the conventional GHT, the step of cell value incrementation can be described simply by:

$$H(x, y, \bar{S}, \bar{\Theta}) = \sum_{x'=1}^{N} \sum_{y'=1}^{M} EP(x', y') \cdot V_{PSF}((x' - x) + x_c, (y' - y) + y_c, \bar{S}, \bar{\Theta}), \quad (3)$$

where $N \times M$ is the size of the input image. For the proposed incrementation strategies, a similar equation for the cell value incrementation step will be given later.

By using these PSFs, only the non-zero votes contribute to form the HCS, so the amount of processing time for cell value incrementation for each edge point in the input image is just proportional to the number of the non-zero votes in the PSF (denoted as #(PSF) in the sequel).

2.7. *Illustrative example*

In this paper, we use point patterns to illustrate the algorithms. A given template shape is shown in Fig. 1(a) which contains five corner points (the lines between the corner points are shown to make the shape clearer; they are not part of the template), and whose centroid R_1 is selected as the reference point. Another template shape identical to that of Fig. 1(a) but with a 1/2 scale is shown in Fig. 1(b) whose centroid R_2 again is selected as the reference point. In Fig. 1(c), the template shape of Fig. 1(a) is rotated through 180°, and the five displacement vectors between all the pattern points and the reference point R_1 constitute the reference table. In Fig. 1(d), the displacement vectors of the template shape of Fig. 1(b) are shown. Figures 1(e) and (f) show the corresponding PSFs of the template shapes of Figs 1(a) and (b), respectively, in which the vote values of the cells pointed to by the displacement vectors superimposed on the central points of the PSFs are just one and the other vote values are zero (shown as blank squares). Furthermore, two corresponding PSFTs are also constructed. In the PSFT for the template shape of Fig. 1(a), the list of vote sources associated with each non-zero vote value includes just a single and identical scale-orientation pair (1.0, 0°). And in the PSFT for the template shape of Fig. 1(b), the lists of vote sources all have the same scale-orientation pair (0.5, 0°).

A shape shown in Fig. 1(g), which is identical to the template shape of Fig. 1(b), is used as the input image and Algorithm 1, or equivalently Equation (3), is performed to superimpose all the displacement vectors shown in Fig. 1(d) of the template shape of Fig. 1(b) on the five corner points of the input shape. There exist exactly five displacement vectors pointing to a specific cell R'_2 which corresponds to the reference point R_2 of the template. This means that detection of the template is successful.

In Fig. 1(h), all the displacement vectors shown in Fig. 1(c) of the given template shown in Fig. 1(a) are superimposed on the five corner points of the input image. In this case, the scale of the superimposed displacement vectors is different from that of the input shape; as a result, the cell with the maximum value 5 cannot be found out.

3. PROPOSED NEW CELL VALUE INCREMENTATION STRATEGIES FOR THE GHT

3.1. *Proposed scale-invariant cell value incrementation strategy for detecting a scaled shape*

The proposed SI cell value incrementation strategy is useful for the case where the shape to be detected is known to be in a specific orientation $\bar{\Theta}$ but with variable scales. In the proposed strategy, instead of incrementing the values of the cells which are pointed to by the displacement vectors as in the conventional GHT, the value of each cell passed through by the half line along each displacement vector in the basic reference table is incremented by one. That is,

for the specific orientation $\bar{\Theta}$,
for each edge point (x_e, y_e), and
for each displacement vector (r_i, θ_i) in the basic reference table,
set $T(x, y) = T(x, y) + 1$ if $\mathrm{TAN}^{-1}(x - x_e, y - y_e) = \theta_i + \bar{\Theta}$
(i.e. if (x, y) falls on the half line from (x_e, y_e) with direction $\theta_i + \bar{\Theta}$), (4)

where T is a 2D Hough counting space, and function

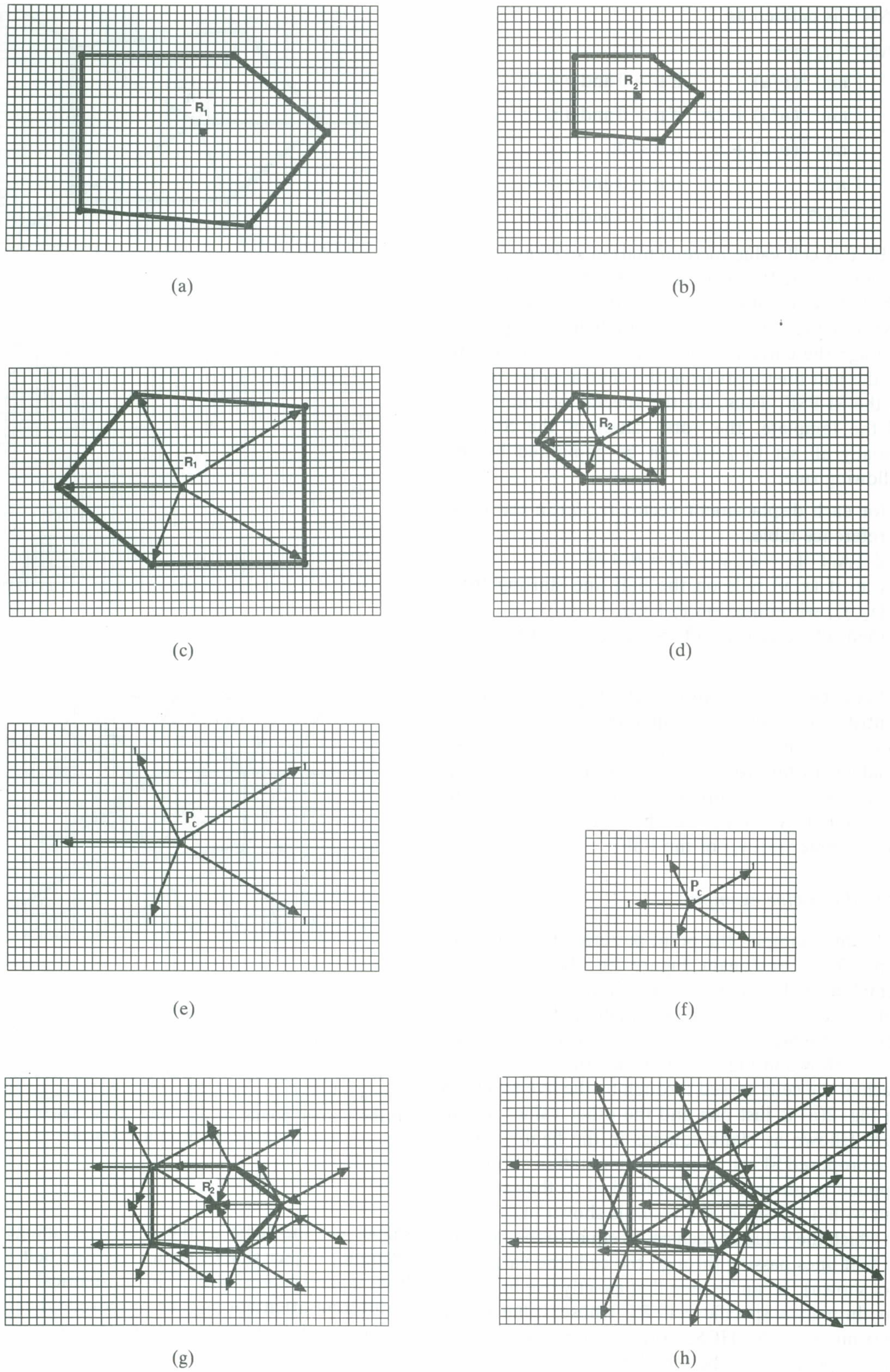


Fig. 1. Illustration of the conventional GHT: (a) point patterns of the template (five points); (b) point patterns of the template with a 1/2 size; (c) five displacement vectors of the template in (a); (d) five displacement vectors of the template in (b); (e) the PSF of the template in (a); (f) the PSF of the template in (b); (g) all the displacement vectors in (d) superimposed on the corner points of the input shape; (h) all the displacement vectors in (c) superimposed on the corner points of the input shape.

TAN^{-1} is defined as

$$\mathrm{TAN}^{-1}(u, v) = \begin{cases} \tan^{-1}(v/u) & \text{if } u > 0 \text{ and } v \geq 0; \\ \tan^{-1}(v/u) + 2\pi & \text{if } u > 0 \text{ and } v \leq 0; \\ \tan^{-1}(v/u) + \pi & \text{if } u < 0; \\ \pi/2 & \text{if } u = 0 \text{ and } v > 0; \\ 3\pi/2 & \text{if } u = 0 \text{ and } v < 0. \end{cases}$$

By this cell value incrementation strategy, a corresponding SI-PSF can be constructed in advance. The vote value of each point in the space of the SI-PSF is equal to the number of half lines passing through the corresponding cell of the point, where all the half lines start from the central point (x_c, y_c) of the SI-PSF. Thus, for the specific orientation $\bar{\Theta}$ of the template, the vote values of the SI-PSF, initially all set to zero, can be computed by the following rule:

for each displacement vector (r_i, θ_i) in the basic reference table,
if $\mathrm{TAN}^{-1}(x - x_c, y - y_c) = \theta_i + \bar{\Theta}$
(i.e. if (x, y) falls on the half line starting from (x_c, y_c) with direction $\theta_i + \bar{\Theta}$),
then set $V_{\text{SI-PSF}}(x, y, \bar{\Theta}) = V_{\text{SI-PSF}}(x, y, \bar{\Theta}) + 1$. (5)

For each incrementation of the $V_{\text{SI-PSF}}$ value in (5) contributed by displacement vector (r_i, θ_i), a scale-orientation pair $(((x - x_c)^2 + (y - y_c)^2)^{1/2}/r_i, \bar{\Theta})$ is added to the list of the vote sources at location (x, y) in the corresponding PSFT, where $((x - x_c)^2 + (y - y_c)^2)^{1/2}$ is the distance from (x, y) to the center (x_c, y_c) of the PSF.

3.2. *Illustrative example*

In this example, the point patterns shown in Section 2.7 are used again. Figure 2(a) shows the given template with five half lines all starting from the reference point R_1 of the template and going along the five displacement vectors. The corresponding SI-PSF is shown in Fig. 2(b) in which the vote value of each of the cells passed through by the five half lines in Fig. 2(a) is just one and the other values are all zero (shown as blank squares in the figure). After the proposed SI cell value incrementation strategy is applied to an input shape identical to the template but with a 1/2 scale, the result is shown in Fig. 2(c) in which the value of each cell is equal to the number of half lines passing through the cell (the values are not shown). It can be seen that exactly five half lines go through a specific cell R_1' whose value is the maximum in the HCS, corresponding to the reference point R_1' of the template.

3.3. *Proposed orientation-invariant (OI) cell value incrementation strategy for detecting a rotated shape*

In many applications, the scales of template object shapes are the same as those of input images. For example, object shapes on a conveyor belt viewed from a fixed camera are all in an identical scale but in different orientations. The proposed OI cell value incrementation strategy is useful for such cases. It can be used to handle orientation variations of input shapes.

The OI cell value incrementation strategy for detecting a rotated shape is similar to the SI cell value incrementation strategy for detecting a scaled shape, except that circles instead of half lines are

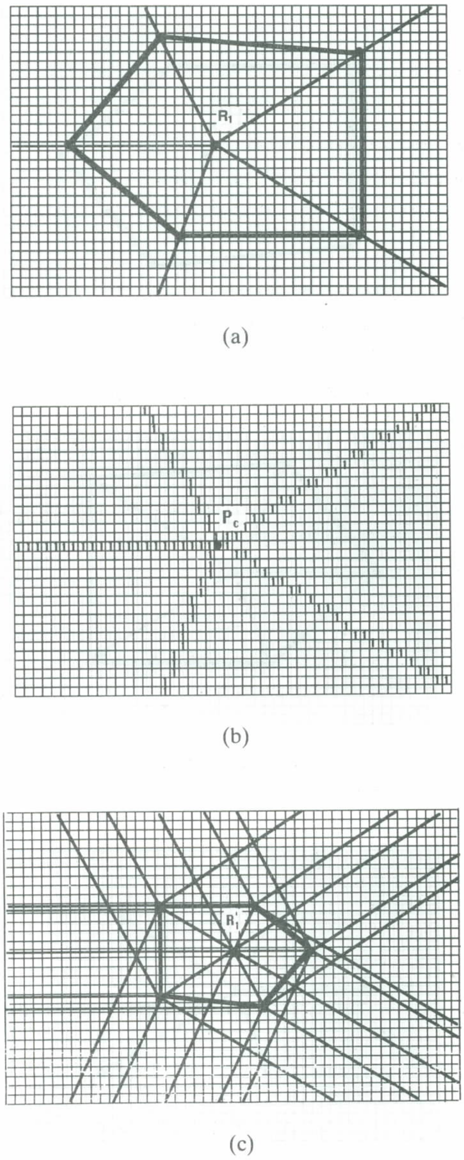


Fig. 2. Illustration of the proposed SI incrementation strategy for handling scale variations in the GHT: (a) the template shape and the five half lines of the reference table for the SI incrementation strategy; (b) the SI-PSF of the template in (a); (c) all the half lines in (a) superimposed on the corner points of an input shape identical to the template but with a 1/2 scale.

superimposed on the edge points. Let input shapes be all in a specific scale $\bar{S}$. Similar to (4) in Section 3.1, the following statement describes the proposed OI cell value incrementation strategy:

for the specific scale $\bar{S}$,
for each edge point (x_e, y_e), and
for each displacement vector (r_i, θ_i) in the basic reference table,
set $T(x,y) = T(x,y) + 1$ if $((x - x_e)^2 + (y - y_e)^2)^{1/2} = \bar{S} \times r_i$
(i.e. if (x, y) falls on the circle with radius $\bar{S} \times r_i$ and center (x_e, y_e)), (6)

where T is a 2D Hough counting space.

By this incrementation strategy, the corresponding OI-PSF includes a set of votes with each vote value being just the number of circles passing through the corresponding cell of the vote value, where the centers of all the circles are located identically at the central point (x_c, y_c) of the OI-PSF. In this OI-PSF, again only two parameters, namely, the x- and y-translations, are necessary. Thus corresponding to the specific scale $\bar{S}$ of the template, the vote values of the OI-PSF can be computed by the following rule:

for each displacement vector (r_i, θ_i) in the basic reference table,
if $((x - x_c)^2 + (y - y_c)^2)^{1/2} = \bar{S} \times r_i$
(i.e. if (x, y) falls on the circle with radius $\bar{S} \times r_i$ and center (x_c, y_c)),
then set $V_{\text{OI-PSF}}(x, y, \bar{S}) = V_{\text{OI-PSF}}(x, y, \bar{S}) + 1$. (7)

For each incrementation of the $V_{\text{OI-PSF}}$ value in (7) contributed by displacement vector (r_i, θ_i), a scale-orientation pair $(\bar{S}, \text{TAN}^{-1}(x - x_c, y - y_c) - \theta_i)$ is added to the list of the vote sources at location (x, y) in the corresponding PSFT, where $\text{TAN}^{-1}(x - x_c, y - y_c)$ is the direction angle of the vector from the center (x_c, y_c) of the PSF to (x, y).

3.4. *Illustrative example*

In this example, point patterns are used again. Figure 3(a) shows a template with five circles, each taking the reference point R of the given template as its center and passing through one of the five corner points. The corresponding OI-PSF is shown in Fig. 3(b), in which the vote value of each cell is just the number of circles passing through the cell. After the proposed OI cell value incrementation strategy is applied to an input shape identical to the

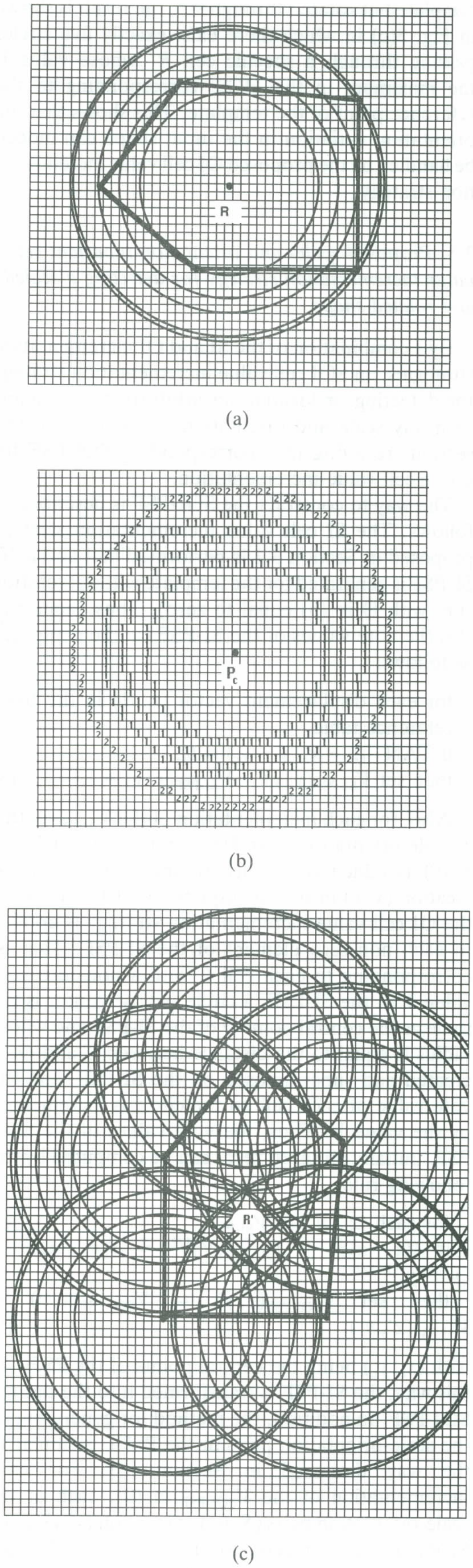


Fig. 3. Illustration of the proposed OI incrementation strategy for handling orientation variations of input shapes in GHT: (a) the template and five circles for the incrementation strategy; (b) the OI-PSF of the template in (a); (c) all the circles in (a) superimposed on the corner points of an input shape identical to the template but rotated through an angle of 90°.

template but rotated through 90°, the result is shown in Fig. 3(c) in which there exist exactly five circles passing through a specific cell R' whose value is the maximum in the HCS, corresponding to the reference point of the template. This shows that the orientation variation in the input shape can indeed be handled by the proposed OI cell value incrementation strategy.

3.5. *Proposed scale- and orientation-invariant cell value incrementation strategy for detecting a scaled and rotated shape*

By a combination of the SI and OI incrementation strategies, an SOI cell value incrementation strategy for detecting or locating an arbitrary object shape with any scale and orientation is proposed in this section, resulting in a corresponding SOI-PSF for use in cell value incrementation.

The way to construct the SOI-PSF is described as follows. The SI cell value incrementation strategy proposed in Section 3.1 is performed first and a 2D SI-PSF is generated for the template without rotation (i.e. with $\bar{\Theta}$ in (5) equal to zero). Thus, similar to (5), the vote values of the SI-PSF can be computed as follows:

for each displacement vector (r_i, θ_i) in the basic reference table,
if $\mathrm{TAN}^{-1}(x - x_c, y - y_c) = \theta_i$,
then set $V_{\text{SI-PSF}}(x, y) = V_{\text{SI-PSF}}(x, y) + 1$. (8)

Also, for each incrementation of the $V_{\text{SI-PSF}}$ in (8), a scale-orientation pair $(((x - x_c)^2 + (y - y_c)^2)^{1/2}/r_i, 0°)$ is added to the list of the vote sources at location (x, y) in the corresponding SI-PSFT. Next, the OI cell value incrementation strategy is applied to each non-zero element of the SI-PSF. That is, corresponding to each cell C with a non-zero vote value V in the SI-PSF, a circle, whose center is the same as the central point O of the SI-PSF and whose radius is equal to the distance between C and O, is created and superimposed on the SI-PSF, and the vote value of each cell on the circle is incremented by the value V. Thus, the vote values of the SOI-PSF, initially all set to zero, can be computed by the following rule:

for each (x', y') in $V_{\text{SI-PSF}}$
if $((x - x_c)^2 + (y - y_c)^2)^{1/2} = ((x' - x_c)^2 + (y' - y_c)^2)^{1/2}$
then set $V_{\text{SOI-PSF}}(x, y) = V_{\text{SOI-PSF}}(x, y) + V_{\text{SI-PSF}}(x', y')$, (9)

with (x_c, y_c) being the central points of the PSFs. In this way, the SI-PSF is transformed into the desired SOI-PSF. In addition, the method for generating all the scale-orientation pairs in the PSFT is similar to the way discussed in Section 3.3. For each incrementation of the $V_{\text{SOI-PSF}}$ in (9), each of the scale-orientation pairs $(S, \theta°)$ at location (x, y) in the SI-PSFT is changed to be $(S, \mathrm{TAN}^{-1}(x - x_c, y - y_c) - \theta_i)$ which is then added to the list of the vote sources at location (x, y) in the desired SOI-PSFT, where θ_i is the direction of the ith displacement vector.

The way to perform cell value incrementation operations using the SOI-PSF is similar to (3). That is, the HCS T now can be computed by:

$$T(x, y) = \sum_{x'=1}^{N} \sum_{y'=1}^{M} EP(x', y') \cdot V_{\text{SOI-PSF}}((x' - x) + x_c, (y' - y) + y_c), \quad (10)$$

where EP, N and M are as defined in (3). In the SOI-PSF, again only x- and y-translations are necessary. It is the same as the cases of the SI and OI incrementation strategies. That is, the HCS for the three proposed strategies are all 2D, in contrast with the 4D HCS required by the conventional GHT using the brute force approach.

4. PROPOSED SCALE- AND ORIENTATION-INVARIANT GHT

4.1. *Algorithm of proposed SOIGHT*

In the proposed SOIGHT, the 2D SOI-PSF is used for cell value incrementation in the 2D HCS $T(X_t, Y_t)$. After the SOI cell value incrementation strategy is applied, a maximum cell value detection procedure is invoked to find out the location of the cell whose value is the maximum in the HCS, which is then taken to be corresponding to the reference point of the detected shape. In certain shape detection applications, only existence of the template shape in an input image need be checked and the specific scale and orientation of the shape may not be required. For such cases, the SOIGHT described so far in this paper is sufficient. For cases where shape scales and orientations are also desired, a certain means to determine the scale and orientation of the detected shape is required. One method for this purpose is proposed as follows.

First, for each cell C in T, a corresponding vote source record table (VSRT) for recording the locations of the sources of the votes for C is constructed. Each record in the VSRT for C indicates a location L in the point spread function space whose non-zero vote value is used to increment the value of C. The VSRT for each C is set empty initially. Then, in the cell value incrementation step, corresponding to each incrementation of a cell value in T, the location of the vote source in the space of $V_{\text{SOI-PSF}}$, i.e. the coordinates $(x' - x + x_c, y' - y + y_c)$ in (10), is recorded in the VSRTs. After the location (x_o, y_o) of the reference point in the detected shape is found out, the set of the records of the corresponding VSRT, denoted as V_o, is examined to obtain the locations of the vote sources of (x_o, y_o). And according to each record in V_o, the lists of the vote sources in the PSFT mentioned in Section 2.6 are examined in turn and collected as a set S.

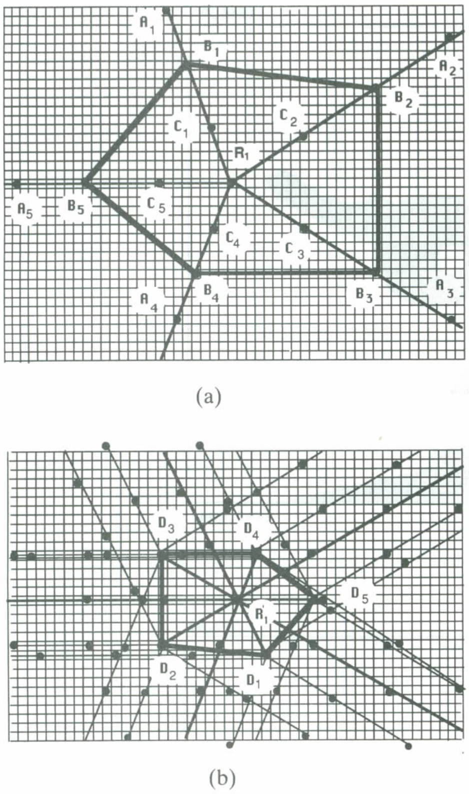


Fig. 4. Illustration of the method of determining the scale and orientation of the detected shape: (a) three groups of sample points on the five half lines of the reference table of the template shape; (b) all the half lines in (a) superimposed on the corner points of an input shape identical to the template but with a different size.

Each vote source in S is a scale-orientation pair. Finally, the pair (S_o, Θ_o) which occurs for the maximum number of times in S is found out as the desired scale and orientation of the detected object shape. (S_o, Θ_o) will be called the most frequent scale-orientation pair. Thus the location, scale and orientation of the input shape are all determined now. A detailed algorithm is described in the following.

Algorithm 2. SOIGHT using the proposed SOI cell value incrementation strategy.

Input. A given template shape and an input image containing a scaled and rotated version of the template.

Output. The location (x_o, y_o) in the input image where the template shape appears, and the scale S_o and orientation Θ_o of the object.

Steps.

(1) Form a 2D Hough counting space $T(X_t, Y_t)$, set all the values of the cells in T to zero, and build up an empty VSRT for each cell in T.

(2) Trace all the edge points in the input image.

(3) Cell value incrementation: for each edge point in the input image, use all the non-zero votes of the SOI-PSF to increment the values of the cells in T according to the SOI cell value incrementation strategy mentioned in Section 3.5, and make the corresponding records in the VSRTs of the cells in T.

(4) Maximum cell value detection: find out the cell C_o with its value exceeding a threshold t and being the maximum in T, and regard the location (x_o, y_o) of C_o as corresponding to the reference point of the given template.

(5) Determination of the scale and orientation: check the records in the VSRT of C_o, retrieve accordingly the lists of the vote sources in the PSFT, and find the most frequent scale-orientation pair (S_o, Θ_o) from the retrieved vote sources.

(6) Exit with (x_o, y_o), S_o and Θ_o as the output.

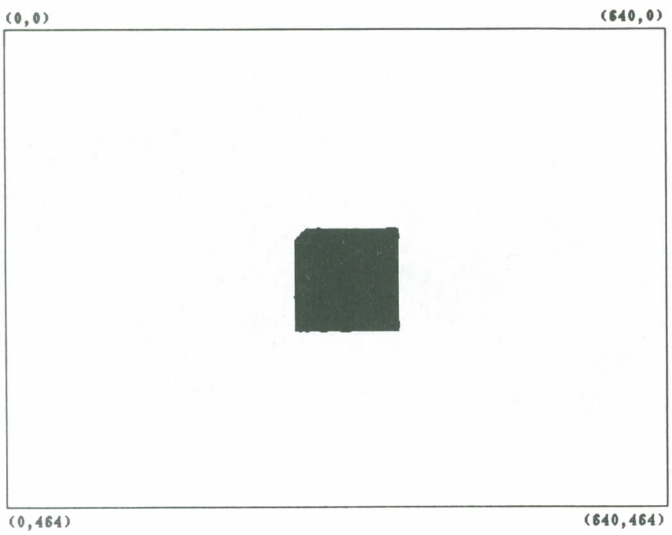
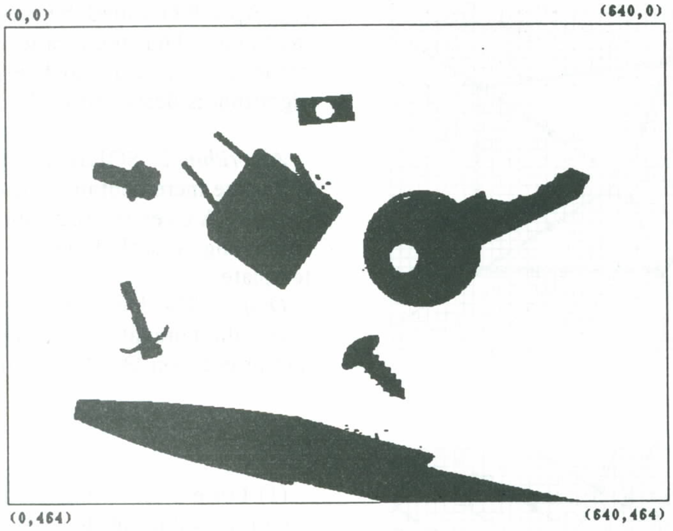
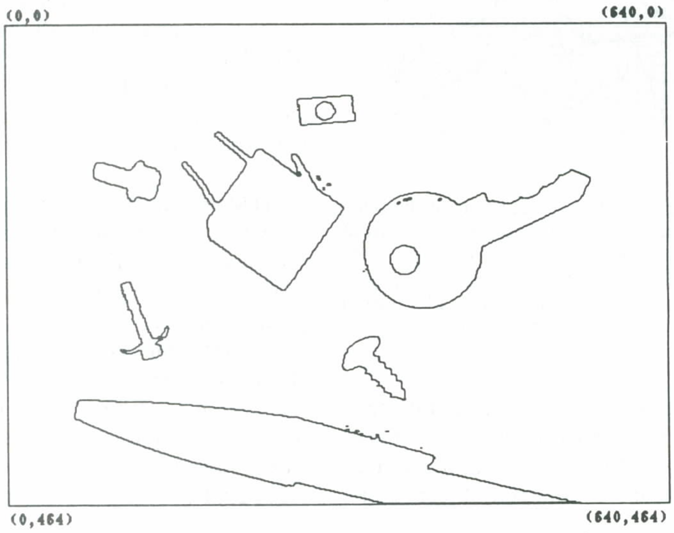


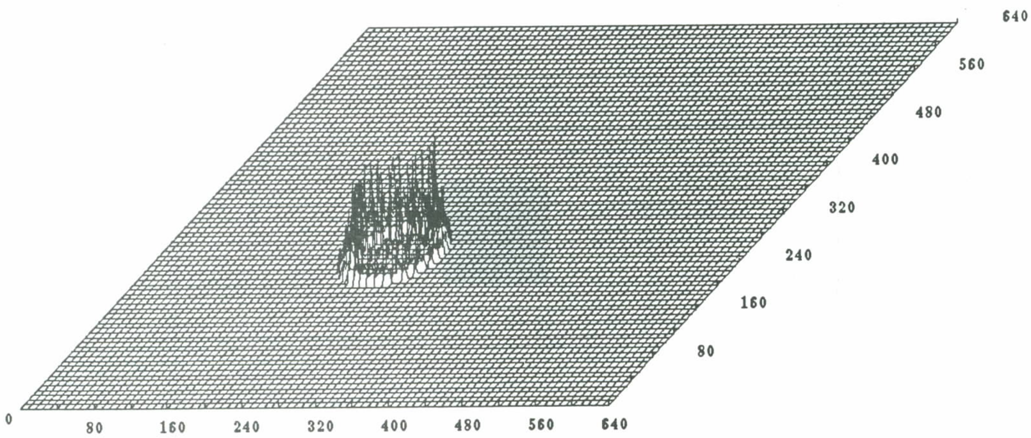
Fig. 5. The image of the template used in the experiments.



(a)

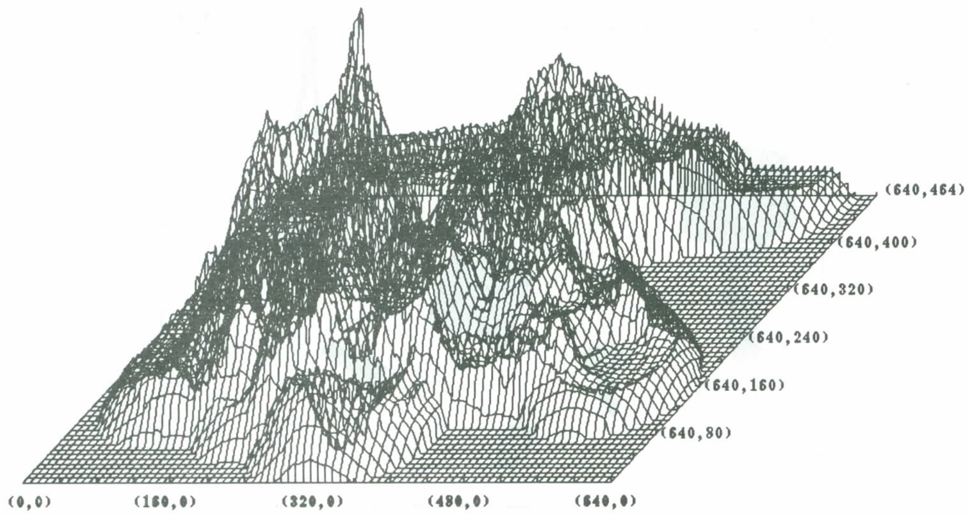


(b)

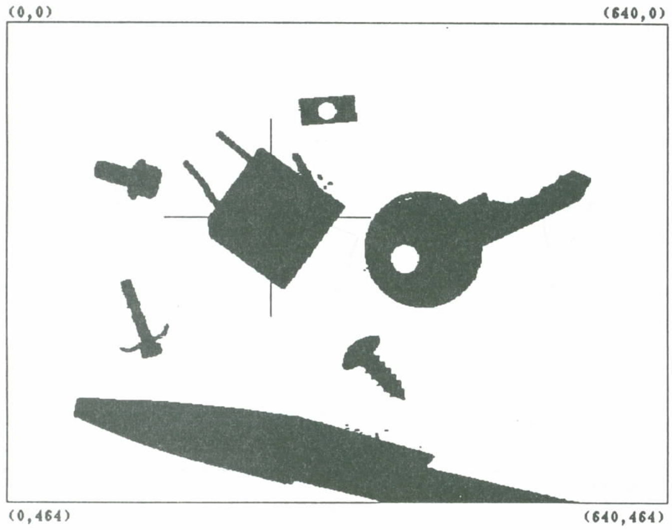


(c)

Fig. 6. (*Continued on following page*.)



(d)



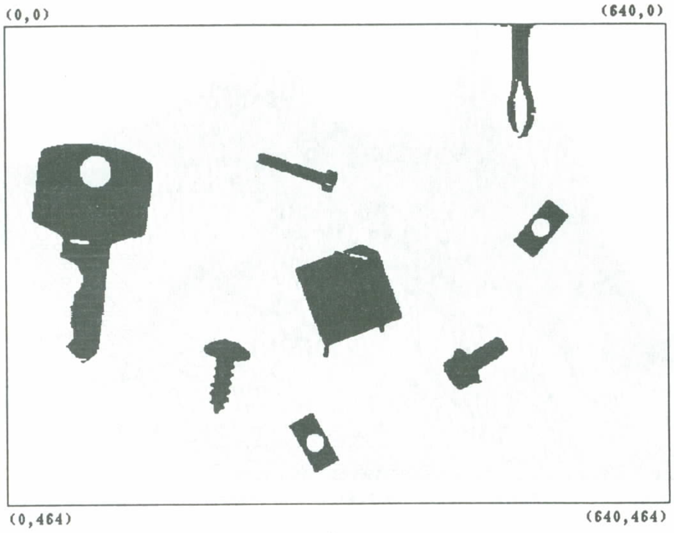
(e)

Fig. 6. Illustration of Experiment 1: (a) an input image; (b) the edge points in the input image; (c) point spread function of the template; (d) the cell values in the HCS for the template; (e) the detected shape with a cross symbol in the bilevel input image.

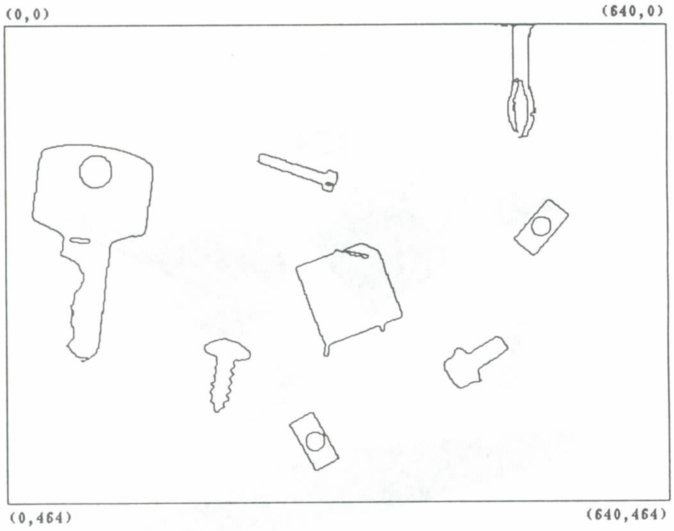
4.2. *Illustrative example*

In this simple example, the point patterns in Fig. 2 are used again to illustrate the step (Step 5) of determining the scale and orientation of the detected shape in Algorithm 2. For simplicity of illustration, only three sample points (shown as black dots in Fig. 4(a)) with different scales on each of the half lines of the reference table are used, and only the case of an unrotated template is considered. In Fig. 4(a), point R_1 is the reference point of the template, and the given template is rotated through 180° with respect to R_1 to construct the reference table. There exist three groups of points, A_i, B_i and C_i, $i = 1, \ldots, 5$, on the half lines constructed from the reference table. All the points A_i in the first group have an identical scale-orientation pair (1.5, 0°) in the PSFT; all the points B_i in the second group have another identical scale-orientation pair (1.0, 0°) in the PSFT; and all the points C_i in the third group have a third identical scale-orientation pair (0.5, 0°) in the PSFT.

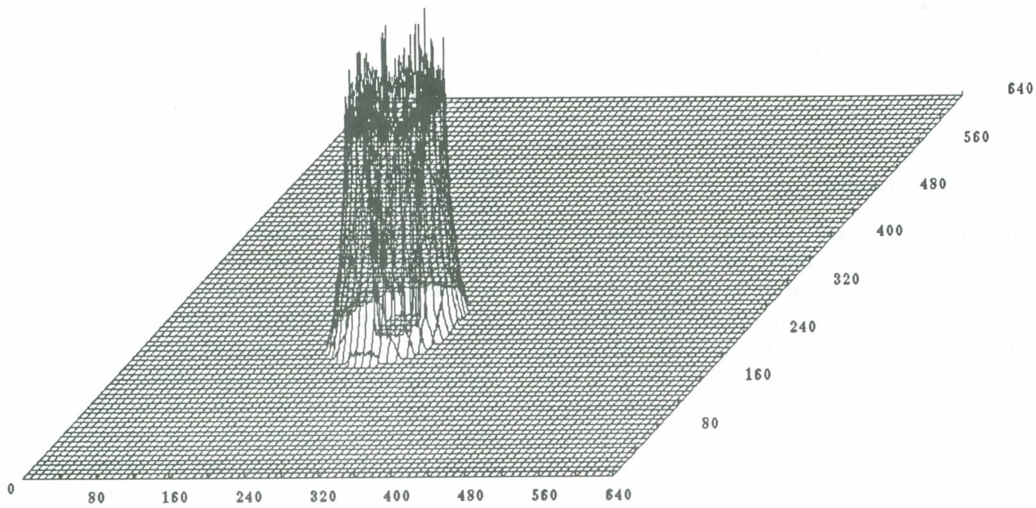
In Fig. 4(b), all the half lines in Fig. 4(a) are superimposed on the corner points of an input shape with a size different from that of the template in Fig. 4(a). After the steps of cell value incrementation and maximum cell value detection are done, the location R_1' of the cell with the maximum cell value in the HCS shown in Fig. 4(b) is found out. Exactly five half lines pass through R_1'. Each of the five half lines going through R_1' starts from one of the corner points



(a)

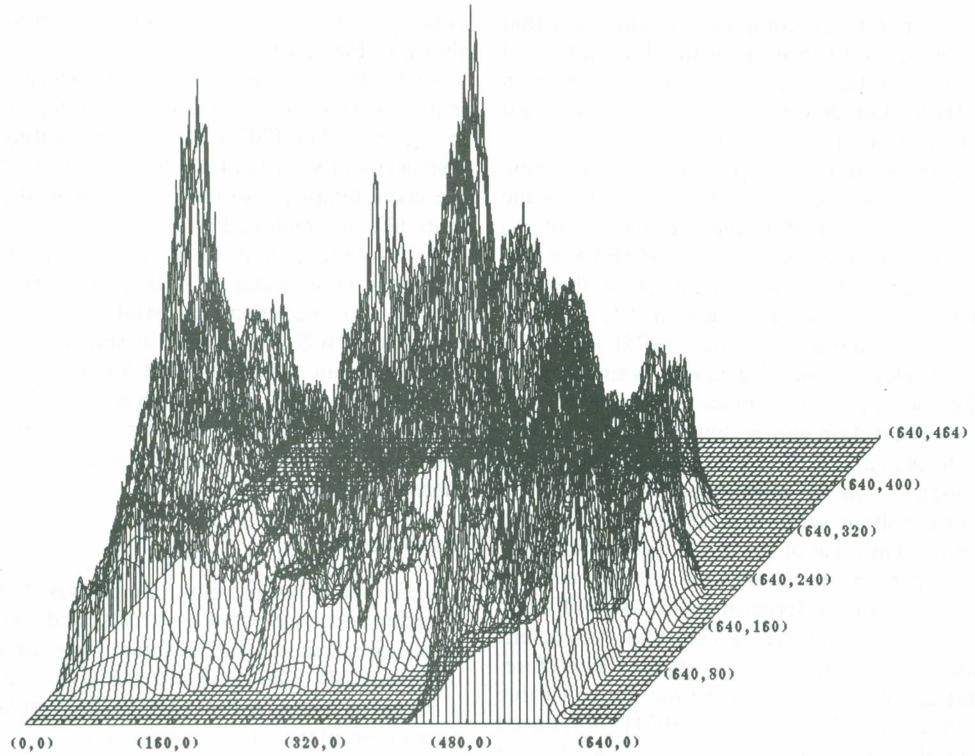


(b)

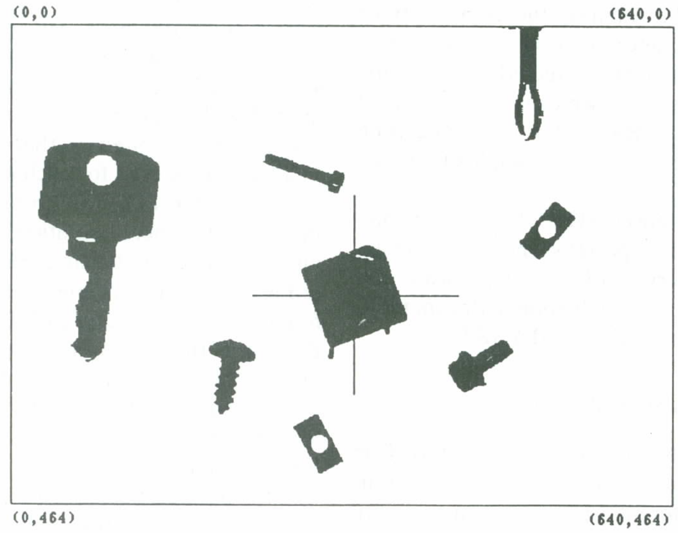


(c)

Fig. 7. (*Continued on following page.*)



(d)



(e)

Fig. 7. Illustration of Experiment 2: (a) an input image; (b) the edge points in the input image; (c) point spread function of the template by using the proposed SOI incrementation strategy; (d) the cell values in the HCS; (e) the detected shape with cross symbol in the bilevel image of the input.

D_i, $i = 1, \ldots, 5$, i.e. all the locations of the five points D_i are added into the VSRT for R'_1, denoted as $V_{R'_1}$. According to the location of each D_i in $V_{R'_1}$, the corresponding list of vote sources in the PSFT is retrieved, which only includes the scale-orientation pair $(0.5, 0°)$ associated with the points C_i of the third group shown in Fig. 4(a). After the scale-orientation pairs for the five points D_i are all collected, the most frequent scale-orientation pair is finally determined to be $(0.5, 0°)$.

4.3. *Complexity analysis for the SOIGHT*

In the proposed SOIGHT, a 2D HCS is required;

and the computational complexity of the algorithm can be divided into three considerable parts: cell value incrementation, maximum cell value detection in the HCS, and determination of the scale and orientation of the detected shapes.

The computational complexity for cell value incrementation is proportional to the number of the non-zero votes in the PSF and the number of the edge points in the input image, i.e. $\#(\text{PSF}) \times \#(E)$. When compared with the complexity of the conventional GHT discussed in Section 2.5, the ratio R can be computed to be $\#(\text{PSF})/(\#(S) \times \#(\Theta) \times \#(\text{BRT}))$. Since R generally is smaller than one, the total amount of processing time for cell value incrementation can be lowered down in proportion to the ratio R. For example, in our experiments discussed in the next section, the quantized values of the shape scales are 0.8, 0.9 and 1.0; and the quantized interval of the shape orientation is 1°, so $\#(S)$ and $\#(\Theta)$ are equal to 3 and 360, respectively. In the case of Experiment 2, #(BRT) and #(PSF) are 401 and 11,312, respectively. As a result, the ratio R is computed to be 0.026. Therefore, the total amount of processing time for cell value incrementation in the proposed SOIGHT is reduced to be in the ratio 0.026 of that required by the conventional GHT. Next, since the processed area in maximum cell value detection is less than that for the conventional GHT. Finally, the complexity of scale and orientation determination is proportional to the size of the part of the detected object shape which matches the template, which can be estimated to be of the order of #(BRT). So this item can be ignored when compared with the complexity of cell value incrementation.

To sum up, the proposed SOIGHT is faster than the conventional GHT, especially when the numbers of the quantized intervals of the shape scale and orientation are large, although some extra memory space must be used for PSFTs and VSRTs.

5. EXPERIMENTAL RESULTS

In our experiments, the proposed SOIGHT is applied to a case of using a computer vision system to detect or locate different capacitors in an assembly line. One type of capacitor is used as the template as shown in Fig. 5, in which the pins of the capacitor are not included.

Experiment 1

In this experiment, the proposed GHT with the OI cell value incrementation strategy is performed to detect a rotated capacitor shape in an input image according to a given template shape shown in Fig. 5. There is no scale variation here. A bilevel input image of the size 640 × 464, including a capacitor as shown in Fig. 5 but with a different orientation, is shown in Fig. 6(a). Then by performing the Sobel edge operations,[8] the edge points are produced as shown in Fig. 6(b).

An OI-PSF for the template is constructed for use in the OI cell value incrementation strategy as shown in Fig. 6(c). The PSF is built up by rotating all the displacement vectors of the basic reference table of the given template, so that the shape of the PSF is quite like an annulus. For convenience to show the PSF, the central point of the PSF is selected to be located at coordinates (256,256). After performing the cell value incrementation strategy on all the edge points, the HCS is generated as shown in Fig. 6(d). The maximum cell value in HCS is found out to be at the location (255,187), which is marked by a cross symbol "+" in Fig. 6(e). The detected orientation of the capacitor is 306° when compared with the template in the clockwise direction.

Experiment 2

In the second experiment, the proposed SOIGHT is performed to detect a distorted scaled and rotated capacitor shape in an input image. The input image of the size 640 × 464 is shown in Fig. 7(a), including a capacitor whose scale is 4/5 of the template. The edge points of the input image are produced as shown in Fig. 7(b).

By using the SOIGHT, an SOI-PSF of the template as shown in Fig. 7(c) is constructed for incrementing the HCS. The vote values in the PSF are mostly greater than those of the OI-PSF constructed in Experiment 1. The shape of the PSF is limited within an area because the scaling range is assumed to be limited (so that a line segment instead of a half line is used for each displacement vector in generating the SOI-PSF from the OI-PSF). After the cell value incrementation strategy is performed on all the edge points, the generated HCS is as shown in Fig. 7(d). The maximum cell value in the HCS is found out at coordinates (330,261). After performing the step of the determination of the scale and orientation (Step 5 of Algorithm 2), the most frequent scale-orientation pair (0.8, 69°) is found out. Therefore, the detected scale of the shape is in a scale of 0.8 of the template and the detected orientation is 69° when compared with the template in the clockwise direction. The location (330,261) is marked by a cross symbol "+" as shown in Fig. 7(e). By this experiment, it is seen that the proposed SOIGHT can be effectively applied to detect or locate a distorted, scaled and rotated object shape.

6. CONCLUDING REMARKS

Two new incrementation strategies, one SI and the other OI, for the GHT have been proposed for handling shape scale and orientation variations, respectively, which the conventional GHT is usually considered to be unable to handle unless by brute force. And by a combination of these incrementation strategies, an SOI cell value incrementation strategy handling both scale and orientation variations is formed. For these incrementation strategies, a new version of the GHT, called SOIGHT, has been proposed for detecting or locating arbitrary 2D object shapes with variable scales and orientations. The required HCS is reduced to 2D so that the computation requirement of maximum cell value detection can be reduced effectively. The amount of processing time for cell value incrementation by the proposed SOIGHT is also less than that of the conventional GHT. Thus the total computational complexity of the SOIGHT is reduced greatly. Further research may be directed to the analysis of the behavior of the HCS when more than one shape occurs in an input image and the exploration of a more effective method for detecting or recognizing multiclass object shapes in more complicated images.

REFERENCES

1. P. V. C. Hough, *Method and Means for Recognizing Complex Patterns*. U.S. Patent 3069654 (1962).
2. R. O. Duda and P. E. Hart, Use of the Hough transform to detect lines and curves in pictures, *Communs. Ass. comput. Mach.* **15**, 11–15 (1972).
3. P. M. Merlin and D. J. Farber, A parallel mechanism for detecting curves in pictures, *IEEE Trans. Comput.* **24**, 96–98 (1975).
4. D. H. Ballard, Generalizing the Hough transform to detect arbitrary shapes, *Pattern Recognition* **13**, 111–122 (1981).
5. L. S. Davis, Hierarchical generalized Hough transforms and line-segment based generalized Hough transforms, *Pattern Recognition* **15**, 277–285 (1982).
6. R. Krichnapuram and D. Casasent, Hough space transformations for discrimination and distortion estimation, *Comput. Vision Graphics Image Process.* **38**, 299–316 (1987).
7. V. Cantoni and L. Carrioli, Structural shape recognition in a multi-resolution environment, *Signal Processing* **12**, 267–276 (1987).
8. A. Rosenfeld and A. C. Kak, *Digital Picture Processing*. Academic Press, New York (1982).
9. C. M. Brown, Inherent bias and noise in the Hough transform, *IEEE Trans. Pattern Anal. Mach. Intell.* **5**, 493–505 (1983).
10. D. H. Ballard and C. M. Brown, *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ (1982).

About the Author—WEN-HSIANG TSAI was born on 10 May 1951 in Tainan, Taiwan, Republic of China. He received the B.S. degree from National Taiwan University in 1973, the M.S. degree from Brown University in 1977, and his Ph.D. degree from Purdue University in 1979, all in Electrical Engineering. From 1973 to 1975, he served in the Chinese Navy as an electronics officer. Since November 1979, he has been on the faculty of the Institute of Computer Science and Information Engineering at National Chiao Tung University, Hsinchu, Taiwan. In the academic year of 1983–1984, he was a visiting scholar at the University of Illinois at Chicago. From 1984 to 1986, he was an Assistant Director and later an Associate Director of the Microelectronics and Information Science and Technology Research Center at National Chiao Tung University. He joined the Department of Computer and Information Science at National Chiao Tung University in August 1984, acted as the Head of the Department from 1984 to 1988, and is currently a Professor there. He is also a Researcher in the Institute of Information Science at Academia Sinica, Taipei, Taiwan. He serves as a consultant to several research institutes and industrial companies. His current research interests include pattern recognition, computer vision, image processing and Chinese information processing.

Dr Tsai is an associate editor of *Pattern Recognition*, *International Journal of Pattern Recognition and Artificial Intelligence*, *Journal of the Chinese Institute of Engineers*, *Journal of Information Science and Engineering* and *Proceedings of National Science Council* of the Republic of China. He was elected as an Outstanding Talent of Information Science of the Republic of China in 1986. He was the winner of the 13th Annual Best Paper Award of the Pattern Recognition Society of U.S.A. He obtained the 1987 Outstanding Research Award and the 1988 Distinguished Research Award, both of National Science Council of the Republic of China. He also obtained the 1989 Distinguished Teaching Award of the Ministry of Education of the Republic of China. He was the winner of the 1989 Acer Long Term Award for Outstanding Ph.D. Dissertation Supervision and the winner of 1990 Outstanding Paper Award of the Computer Society of the Republic of China. Dr Tsai has published more than 45 papers in well-known international journals.

About the Author—SHENG-CHING JENG was born on 4 January 1958 in Taipei, Taiwan, Republic of China. He received the B.S. degree in Computer Engineering from National Chiao Tung University in 1980, and the M.S. degree in Electrical Engineering from National Taiwan University in 1982. From 1982, he began to work in the Electronics Research & Service Organization (ERSO) of the Industrial Technology Research Institute (ITRI) in Chutung, Hsinchu, Taiwan. From 1988 to 1990, he has been in the Advanced Technology Center (ATC) of ERSO, ITRI. He is currently a researcher of the ATC, Computer & Communication Research Laboratories, ITRI. In 1986, he entered the Institute of Computer Science and Information Engineering at National Chiao Tung University as a part-time student, and is now working toward his Ph.D. degree. His current research interests include computer vision, pattern recognition and image processing.