

# Colour image detection and matching using modified generalised Hough transform

R.-C. Lo  
W.H. Tsai

*Indexing terms: Colour-image matching, Hough transform, Normalised colour values, Feature vectors*

**Abstract:** A new approach to 2-dimensional (2D) colour-image detection and matching using a modified version of the generalised Hough transform (GHT) is proposed. In the conventional GHT, the useful colour information existing in the input image and the relationship between each pixel and its neighbourhood are not used. Furthermore, lighting changes in the image are not usually considered. Therefore, the conventional GHT is seldom applied to colour images. In the proposed approach, lighting changes are removed using normalised colour values. Next, certain critical pixels of an input colour image whose neighbourhoods have larger variances of normalised colour values are extracted. For each critical pixel, a feature vector, which includes the normalised colour values of the pixel as well as those of the pixel's neighbours, is then constructed. A modified voting rule for the GHT is therefore proposed which is based on a similarity-measure function of the feature vectors. High maximum peaks in the cell array are searched finally as the result. The proposed method is robust for colour-image detection and matching in noisy, occlusive, and lighting-change environments, as demonstrated by experimental results.

## 1 Introduction

The Hough transform is a well known method for detection and location of straight lines or analytic curves [1, 2]. A generalised version of the Hough transform, called the generalised Hough transform (GHT), was proposed by Ballard [3] for detecting arbitrary two-dimensional (2D) shapes. It has been used successfully in many applications of image processing and computer vision such as shape detection, image registration, object matching etc.

In the conventional GHT process, a 'template shape'

is detected in an input image. Before the transform can be performed, it is usually necessary to perform the preprocessing step of boundary detection or shape thinning on both the input image and the template shape to transform them into binary images consisting of boundary or skeletal pixels. Each boundary or skeletal pixel of the template shape is then represented by a displacement vector relative to a reference point in the shape image. All the displacement vectors constitute a 'reference table' (R-table) [3]. To perform the GHT, a 2D Hough counting space (HCS) consisting of a 2D array of accumulators is constructed. The GHT superimposes all the displacement vectors of the R-table on each extracted boundary or skeletal pixel of the input image. The value of each accumulator that is pointed to by a displacement vector is then incremented by one. Finally, the location of the template shape is detected by searching the HCS for the local maximum accumulator value.

The GHT has received a lot of improvements [4-12]. However, for colour-image detection and matching, the GHT can be modified further. In the conventional GHT, the useful colour information existing in the image and the relationship between each pixel and its neighbours are not utilised. Furthermore, lighting changes in images are seldom considered. To include these considerations, a modified version of the conventional GHT is proposed in this study.

First, the RGB colour values of each image pixel are 'normalised' to reduce the influence of lighting changes. Next, certain 'critical pixels' in the input colour image are extracted. A pixel is said to be critical in this study if the variances of the normalised colour values in the neighbourhood of the pixel are sufficiently large. A feature vector then is constructed for each critical pixel. The feature vector includes the normalised colour values of the pixel as well as those of its neighbours in the image. A modified voting rule for the GHT is proposed accordingly, which is based on a similarity-measure function of the feature vector. Each vote for a cell increments the cell value by the fractional value of the similarity measure instead of the fixed value of 1. The location of the cell with the maximum value exceeding a preselected threshold is searched finally as the result of image detection or matching. Advantages of the proposed approach include at least the following.

(i) The extraction of critical pixels transforms the input colour image into a sparse point-type image to which the GHT becomes applicable.

(ii) The use of critical pixels helps avoiding the preprocessing step of boundary detection or shape thinning,

© IEE, 1996

*IEE Proceedings* online no. 19960448

Paper first received 17th July 1995 and in revised form 1st March 1996

The authors are with the Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan 300, Republic of China

R.-C. Lo is also with the Department of Electronic Engineering, National Taipei Institute of Technology, Taipei, Taiwan 106, Republic of China

which usually is necessary before applying the GHT to binary images but is not easy to perform on colour images.

(iii) The normalisation of image colour values reduces the influence of lighting changes in environments to input images and matching results.

(iv) The inclusion of the normalised colour values of each critical pixel and its neighbours in the feature vector makes use of not only the colour information in the image but also the relationship of colour information among the pixel neighbourhood.

(v) The use of the feature vectors of the proposed critical pixels, as well as the corresponding voting rule for fractional-cell-value incrementation, makes the resulting peaks in the HCS more prominent and easier to detect.

As a result, the proposed method is robust for colour-image detection and matching in noisy, occlusive, and lighting-change environments, as shown by the experimental results.

## 2 Review of conventional generalised Hough transform

To use the conventional GHT to detect an arbitrary template shape, it is necessary to set up a 2D HCS( $X_i, Y_i$ ) where ( $X_i, Y_i$ ) is a translation vector with respect to a reference point of the template shape and describes the location of the template shape. Each cell in the HCS has a value specifying the possibility that the reference point of the template shape to be detected is located at the cell.

Before the GHT is performed, an R-table for the template shape is built up by the following steps:

- (i) select a suitable point R in the given template shape as the reference point;
- (ii) rotate the shape 180° with respect to R;
- (iii) trace all the boundary or skeletal pixels of the template shape and construct an R-table consisting of the displacement vectors between all the boundary or skeletal pixels and R.

In the GHT process, all the displacement vectors of the R-table are superimposed on each pixel in the input image. The value of each cell pointed to by a displacement vector is incremented by one. If there exists any cell with its value exceeding a preselected threshold value and being the maximum in HCS, then it is determined that the template shape is detected at the location of the cell.

## 3 Proposed approach for colour-image detection and matching using modified generalised Hough transform

The proposed method extracts critical pixels from an input image and utilises the normalised colour information of each critical pixel and its neighbours to perform the shape detection and matching. By normalising the colour values of the image, the unknown effects of common incident lighting illumination can be removed. Then, each critical pixel is represented by a feature vector which includes the normalised colour values of the pixel itself as well as its neighbouring pixels. To perform the modified GHT process, a 'colour reference table' (abbreviated as CR-table) and a modified voting rule are proposed. The details are described in the following.

### 3.1 Removal of lighting changes

In an RGB colour image, the tristimulus colour values of each pixel at position ( $j, k$ ), denoted as  $R(j, k)$ ,  $G(j, k)$  and  $B(j, k)$  (abbreviated as  $R, G, B$ ), can be approximately modelled [13–17] by integration of products of object-reflectivity function  $F_i(j, k, \lambda)$  with  $i = R$  (red),  $G$  (green), and  $B$  (blue), spectra  $\lambda$  from 380nm to 760nm, light-source illumination function  $\Phi(j, k, \lambda)$ , and a sensor responsivity and gain function  $S_i(j, k, \lambda)$ :

$$R(j, k) = \int_{\lambda} F_R(j, k, \lambda) \Phi(j, k, \lambda) S_R(j, k, \lambda) d\lambda$$

$$G(j, k) = \int_{\lambda} F_G(j, k, \lambda) \Phi(j, k, \lambda) S_G(j, k, \lambda) d\lambda$$

$$B(j, k) = \int_{\lambda} F_B(j, k, \lambda) \Phi(j, k, \lambda) S_B(j, k, \lambda) d\lambda$$

This is a convenient physical model but unfortunately it is affected considerably by lighting changes which are usually unknown. Therefore, it is not suitable for use in the colour-image detection and matching directly. On the other hand, it is known that light sources coming from the lighting changes in normal environments usually are composed of nearly white daylight [13–15]. Let  $\Phi(j, k, \lambda)$  be a daylight sources. According to the theory of Thomas Young [14],  $\Phi(j, k, \lambda)$  can be perceived as  $\sum_{i=1}^3 \beta_i(j, k) P_i(\lambda)$ , where  $P_i(\lambda)$ ,  $i = 1, 2, 3$ , are the values of the three primary-colour spectra with a certain spectral energy distribution, and  $\beta_1(j, k)$ ,  $\beta_2(j, k)$  and  $\beta_3(j, k)$  are the corresponding mixture proportions. For white daylight sources, the  $\beta_1(j, k) : \beta_2(j, k) : \beta_3(j, k)$  are approximately equal. After lighting changes, a certain white daylight source  $\Phi(j, k, \lambda)$  is changed to another one  $\Phi'(j, k, \lambda)$ , which may be characterised with a different spectral distribution, but is usually still a white daylight source in a normal environment, i.e.  $\Phi'(j, k, \lambda)$  can be perceived as  $\sum_{i=1}^3 \beta'_i(j, k) P_i(\lambda)$  too or, equivalently,

$$\beta'_1(j, k) : \beta'_2(j, k) : \beta'_3(j, k) \cong \beta_1(j, k) : \beta_2(j, k) : \beta_3(j, k)$$

Let  $\beta'_i/\beta_i = \eta$ ,  $i = 1, 2, 3$ . Then we have

$$\begin{aligned} R'(j, k) &= \int_{\lambda} F_R(j, k, \lambda) \Phi'(j, k, \lambda) S_R(j, k, \lambda) d\lambda \\ &= \eta \int_{\lambda} F_R(j, k, \lambda) \Phi(j, k, \lambda) S_R(j, k, \lambda) d\lambda \\ &= \eta R(j, k) \end{aligned}$$

$$\begin{aligned} G'(j, k) &= \int_{\lambda} F_G(j, k, \lambda) \Phi'(j, k, \lambda) S_G(j, k, \lambda) d\lambda \\ &= \eta \int_{\lambda} F_G(j, k, \lambda) \Phi(j, k, \lambda) S_G(j, k, \lambda) d\lambda \\ &= \eta G(j, k) \end{aligned}$$

$$\begin{aligned} B'(j, k) &= \int_{\lambda} F_B(j, k, \lambda) \Phi'(j, k, \lambda) S_B(j, k, \lambda) d\lambda \\ &= \eta \int_{\lambda} F_B(j, k, \lambda) \Phi(j, k, \lambda) S_B(j, k, \lambda) d\lambda \\ &= \eta B(j, k) \end{aligned}$$

Note that, for each pixel,  $\eta$  is unknown but identical for all the three colour bands [14, 16]. To remove the influence of  $\eta$ , the normalised colour values  $r(j, k)$ ,  $g(j, k)$ , and  $b(j, k)$  are defined as follows:

$$\begin{aligned}
r(j, k) &= \frac{255R(j, k)}{R(j, k) + G(j, k) + B(j, k)} \\
&= \frac{255R'(j, k)}{R'(j, k) + G'(j, k) + B'(j, k)} \\
g(j, k) &= \frac{255G(j, k)}{R(j, k) + G(j, k) + B(j, k)} \\
&= \frac{255G'(j, k)}{R'(j, k) + G'(j, k) + B'(j, k)} \\
b(j, k) &= \frac{255B(j, k)}{R(j, k) + G(j, k) + B(j, k)} \\
&= \frac{255B'(j, k)}{R'(j, k) + G'(j, k) + B'(j, k)}
\end{aligned} \tag{1}$$

When the light sources are limited to daylight or to those with fixed ratios of mixture proportions of three primary-colour sources, from eqn. 1, as is assumed in this study, we see that the normalised colour values  $[r(j, k), g(j, k), b(j, k)]$  (abbreviated as  $(r, g, b)$  henceforth) only include the surface-reflective properties of the imaged object and a sensor responsivity and gain function. They are not susceptible to environmental lighting changes. Therefore, it is possible to match object images using normalised colour values to avoid the influence of lighting changes, as is done in this study.

### 3.2 Extracting critical pixels from normalised colour image

To extract meaningful critical pixels, we check the variance  $U$  of the  $r, g, b$  values of the neighbourhood of each pixel in the template shape or input image: if  $U$  is larger than a threshold value, then the pixel is defined to be critical and is extracted.

$P_{i1}$	$P_{i2}$	$P_{i3}$	$R_{i1}$	$R_{i2}$	$R_{i3}$	$r_{i1}$	$r_{i2}$	$r_{i3}$
$P_{i8}$	$P_{i1}$	$P_{i4}$	$G_{i1} B_{i1}$	$G_{i2} B_{i2}$	$G_{i3} B_{i3}$	$g_{i1} b_{i1}$	$g_{i2} b_{i2}$	$g_{i3} b_{i3}$
$P_{i7}$	$P_{i6}$	$P_{i5}$	$R_{i8}$	$R_{i1}$	$R_{i4}$	$r_{i8}$	$r_{i1}$	$r_{i4}$
			$G_{i8} B_{i8}$	$G_{i1} B_{i1}$	$G_{i4} B_{i4}$	$g_{i8} b_{i8}$	$g_{i1} b_{i1}$	$g_{i4} b_{i4}$
			$R_{i7}$	$R_{i6}$	$R_{i5}$	$r_{i7}$	$r_{i6}$	$r_{i5}$
			$G_{i7} B_{i7}$	$G_{i6} B_{i6}$	$G_{i5} B_{i5}$	$g_{i7} b_{i7}$	$g_{i6} b_{i6}$	$g_{i5} b_{i5}$

**Fig. 1** Colour information of the neighbourhood of a pixel  $P_i$   
a Pixel  $P_i$  and its 8-neighbours  $P_{ij}$  for  $j = 1, 2, \dots, 8$   
b RGB colour values  $(R_i, G_i, B_i)$  of  $P_i$  and  $(R_{ij}, G_{ij}, B_{ij})$  for  $j = 1, 2, \dots, 8$  of 8-neighbours  
c Corresponding normalised colour values of b

More specifically, let  $P_i$  be a pixel in the processed image and  $P_{ij}$  with  $j = 1, 2, \dots, 8$  be its 8-neighbours, as shown in Fig. 1a. Let  $(R_i, G_i, B_i)$  and  $(R_{ij}, G_{ij}, B_{ij})$  be the  $R, G, B$  colour values of  $P_i$  and  $P_{ij}$ , respectively, as shown in Fig. 1b, and their normalised  $r, g, b$  colour values are  $(r_i, g_i, b_i)$  and  $(r_{ij}, g_{ij}, b_{ij})$ , as shown in Fig. 1c. The critical pixels are extracted by the following steps:

- (i) for each pixel  $P_i$  in the input image, compute the variance  $U_i$  of the  $r, g, b$  values between  $P_i$  and its 8-neighbours, i.e. compute  $U_i = \sum_{j=1}^8 (|r_{ij} - r_i| + |g_{ij} - g_i| + |b_{ij} - b_i|)$ ;
- (ii) if  $U_i$  is greater than a threshold value, then pixel  $P_i$  is defined to be a critical pixel.

### 3.3 Colour reference table

For each colour template shape, we have to constitute a CR-table. The CR-table not only records the dis-

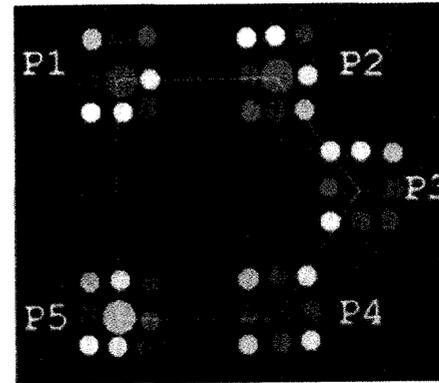
placement vector  $V_i$  between each extracted critical pixel  $P_i$  and the reference point  $R$  but also preserves the normalised colour values of both  $P_i$  and its neighbouring pixels which form a feature vector, as shown in Fig. 4a. The feature vector  $f_i$  of pixel  $P_i$  includes the normalised colour values of  $P_i$  and all of its 8-neighbours  $P_{ij}$  ( $j = 1, 2, \dots, 8$ ), i.e.

$$\begin{aligned}
f_i &= (f_{i0}, f_{inr}, f_{ing}, f_{inb}) \\
&= ((r_{i0}, g_{i0}, b_{i0}), \\
&\quad (r_{i1}, r_{i2}, r_{i3}, r_{i4}, r_{i5}, r_{i6}, r_{i7}, r_{i8}), \\
&\quad (g_{i1}, g_{i2}, g_{i3}, g_{i4}, g_{i5}, g_{i6}, g_{i7}, g_{i8}), \\
&\quad (b_{i1}, b_{i2}, b_{i3}, b_{i4}, b_{i5}, b_{i6}, b_{i7}, b_{i8}))
\end{aligned}$$

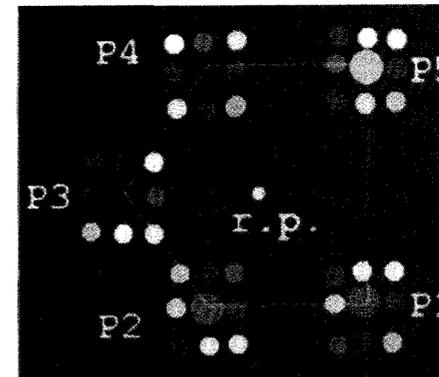
where  $f_{i0} = (r_{i0}, g_{i0}, b_{i0})$  is the normalised colour values of  $P_i$  and  $(f_{inr}, f_{ing}, f_{inb})$  includes the normalised  $r, g, b$  values of the 8-neighbours. All extracted critical pixels together with their feature vectors are traced to constitute the CR-table by the following steps:

- (a) select a suitable point  $R$  in the given template shape as the reference point;
- (b) rotate the shape  $180^\circ$  with respect to  $R$ ;
- (c) trace all the critical pixels of the template shape and construct the CR-table by including the displacement vector between each critical pixel and  $R$ , as well as the feature vector of the pixel.

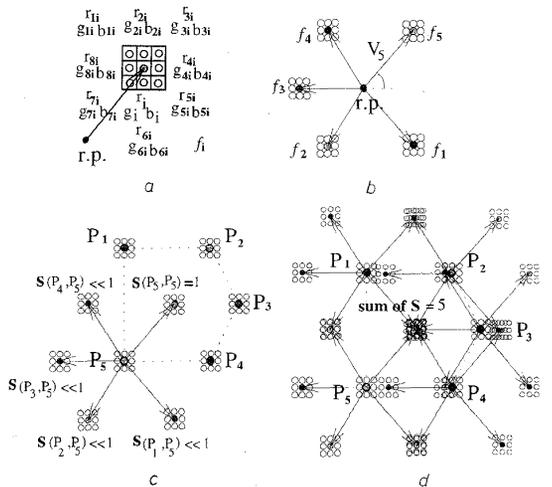
The steps are illustrated by Figs. 2, 3, 4a and 4b.



**Fig. 2** Illustration of building CR-table: extracted template shape consisting of five colour pixels associated with their 8-neighbours



**Fig. 3** Illustration of building CR-table: template shape rotated through  $180^\circ$  with respect to the reference point



**Fig. 4** Illustration of building CR-table  
 a Feature vector of pixel  $P_i$  constructed using normalised colour  $r, g, b$  values  
 b CR-table  
 c Superposition of CR-table on an extracted critical pixel  $P_5$   
 d Superposition of CR-table on all extracted critical pixels

### 3.4 Voting rule

The similarity measure  $S(P_i, P_j)$  between two critical pixels  $P_i$  and  $P_j$  is first defined as

$$S(P_i, P_j) = S_1 S_2$$

$$= \left( \frac{1}{|f_{i0} - f_{j0}| + 1} \right) \left\{ \frac{1}{w_o (|f_{inr} - f_{jnr}| + |f_{ing} - f_{jng}| + |f_{inb} - f_{jnb}|) + 1} \right\}$$

with

$S_1$  = similarity measure between the two points  $P_i$  and  $P_j$  themselves

$$= \frac{1}{|f_{i0} - f_{j0}| + 1} = \frac{1}{|r_{i0} - r_{j0}| + |g_{i0} - g_{j0}| + |b_{i0} - b_{j0}| + 1}$$

$S_2$  = similarity measure between the two sets of 8-neighbours of two points

$$= \left\{ \frac{1}{w_o (|f_{inr} - f_{jnr}| + |f_{ing} - f_{jng}| + |f_{inb} - f_{jnb}|) + 1} \right\}$$

$$= \frac{1}{\left( \sum_{k=1}^8 w_k |r_{ik} - r_{jk}| \right) + \left( \sum_{k=1}^8 w_k |g_{ik} - g_{jk}| \right) + \left( \sum_{k=1}^8 w_k |b_{ik} - b_{jk}| \right) + 1} \quad (2)$$

where

(i)  $\circ$  is the vector-product operator;

(ii)  $w = (w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8)$  is a weighting vector computed by:  $w_k = 1/n$  if neighbouring pixel  $P_k (k = 1, 2, \dots, 8)$  exists;  $= 0$  otherwise

where  $n$  is the total number of the neighbouring pixels (note that the number of the 8-neighbours of a pixel on the image boundary is not necessarily 8);

(iii)  $f_{i0} = (r_{i0}, g_{i0}, b_{i0})$  is the normalised colour values of  $P_i$ ;

(iv)  $f_{j0} = (r_{j0}, g_{j0}, b_{j0})$  is the normalised colour values of  $P_j$ ;

(v)  $f_{inr} = (r_{i1}, r_{i2}, r_{i3}, r_{i4}, r_{i5}, r_{i6}, r_{i7}, r_{i8})$  is the neighbouring red information of pixel  $P_i$ ;

(vi)  $f_{jnr} = (r_{j1}, r_{j2}, r_{j3}, r_{j4}, r_{j5}, r_{j6}, r_{j7}, r_{j8})$  is the neighbouring red information of pixel  $P_j$ ;

(vii)  $f_{ing} = (g_{i1}, g_{i2}, g_{i3}, g_{i4}, g_{i5}, g_{i6}, g_{i7}, g_{i8})$  is the neighbouring green information of pixel  $P_i$ ;

(viii)  $f_{jng} = (g_{j1}, g_{j2}, g_{j3}, g_{j4}, g_{j5}, g_{j6}, g_{j7}, g_{j8})$  is the neighbouring green information of pixel  $P_j$ ;

(ix)  $f_{inb} = (b_{i1}, b_{i2}, b_{i3}, b_{i4}, b_{i5}, b_{i6}, b_{i7}, b_{i8})$  is the neighbouring blue information of pixel  $P_i$ ; and

(x)  $f_{jnb} = (b_{j1}, b_{j2}, b_{j3}, b_{j4}, b_{j5}, b_{j6}, b_{j7}, b_{j8})$  is the neighbouring blue information of pixel  $P_j$ .

Note that owing to consideration of the influence of the pixel itself as well as the neighbouring pixels, the measure  $S(P_i, P_j)$  includes two parts  $S_1$  and  $S_2$ . From the definition of  $S(P_i, P_j)$ , we see that when the feature vectors  $f_i$  and  $f_j$  of two pixels are identical,  $S(P_i, P_j) = 1$ ; on the contrary, the larger is the difference between the feature vectors of the two pixels, the smaller the value of  $S(P_i, P_j)$  becomes. As  $f_i$  and  $f_j$  are totally different,  $S(P_i, P_j) = 1/[255 \times 255] \approx 0$ . The proposed voting rule for colour images is just to use the value of  $S(P_i, P_j)$ , which is fractional, as the increment value instead of the value of 1 in the cell-value-incrementation stage of the GHT. Because  $r_i + g_i + b_i = 255$  and  $r_j + g_j + b_j = 255$ , the  $|r_i - r_j| + |g_i - g_j| + |b_i - b_j|$  in  $S(P_i, P_j)$  is equal to  $|r_i - r_j| + |g_i - g_j| + |(r_j + g_j) - (r_i + g_i)|$ . The feature vector therefore has nine redundant items out of 27 numbers. In this way, the dimensionality of the feature vector can be reduced from 27 items to 18 items.

### 3.5 Modified GHT for detecting and matching colour-object shapes

In the following, we describe the algorithms of the proposed modified GHT. Algorithm 1 is used to build a CR-table and algorithm 2 is used to perform shape matching by detecting a given colour-template shape from an input colour image.

**Algorithm 1:** Building CR-table using a given colour-template shape.

**Input:**

A given RGB colour-template shape.

**Output:**

A CR-table.

**Steps:**

(i) Initialisation: form a 2D array  $B$  as CR-table; set all values of the cells in  $B$  to zero and regard the centre location  $R$  of  $B$  as a common reference point.

(ii) Removing lighting changes: compute the normalised colour images of the template shape using eqn. 1.

(iii) Extracting critical pixels and building feature vectors: extract critical pixels from the normalised template shape by the processes as described in Section 3.2.

(iv) Building CR-table: select the centroid point  $C$  of the template shape as the reference point; rotate the template shape through  $180^\circ$  with respect to  $C$  and translate it in such a way that  $C$  coincides with  $R$ ; trace all extracted critical pixels  $P_i$  which includes the displacement vector  $V_i$  between  $P_i$  and  $C$  as well as its feature vectors; increment by 1 the value of each cell of  $B$  which is pointed to by  $V_i$ .

(v) Output  $B$  associated with the feature vectors of the critical pixels as the CR-table.

(vi) End.

**Algorithm 2:** Modified GHT for colour-image detection or matching.

**Input:**

(i) An input colour image containing a translated object shape partially or fully identical to the given template shape, taken under lighting change environments.

- (ii) A CR-table.
- (iii) A threshold value  $t_v$

**Output:**

A location in the input image where the template shape appears (or more specifically, where the reference point of the template shape is located).

**Steps:**

- (i) Initialisation: form a 2D HCS  $H(X_i, Y_i)$  and set all the values of the cells in  $H$  to zero.
- (ii) Removing lighting changes: compute the normalised colour values of the input image using eqn. 1.
- (iii) Extracting critical pixels and building feature vectors: extract critical pixels from the normalised input image by the processes as described in Section 3.2.
- (iv) Cell-value incrementation: for each displacement vector  $V_i$  of the CR-table which is formed by pixel  $P_i$  in the template shape, perform the following steps:
  - (a) superimpose  $V_i$  on each extracted critical pixel  $P_j$  of the input image;
  - (b) calculate the similarity measure value  $S(P_i, P_j)$  by eqn. 2;
  - (c) increment by  $S(P_i, P_j)$  the value of the cell in  $H$  which is pointed to by the displacement vector  $V_i$
- (v) Maximum cell-value detection: find out as output the location of the cell with its value exceeding  $t_v$  and being the local maximum in  $H$ .
- (vi) End.

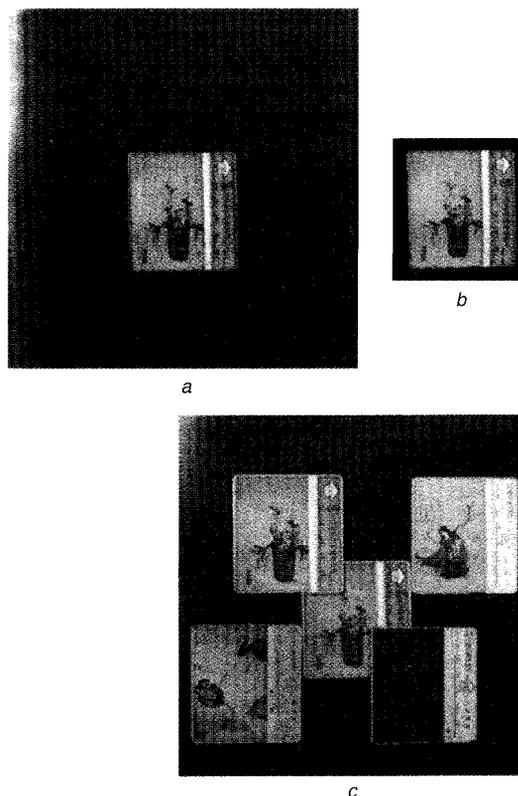
Fig. 4c illustrates the superimposition of all the displacement vectors of a CR-table on an extracted critical pixel, in which the value of each cell pointed to by the displacement vectors of the CR-table is incremented by the similarity measure  $S(P_i, P_j)$  where  $P_i$  is in the CR-table and  $P_j$  is an extracted critical pixel. Fig. 4d illustrates the superimposition of the displacement vectors of the CR-table on all the extracted critical points.

As a result of including the colour information of the pixel neighbourhood for computing the similarity measure  $S$ , the peaks formed in the  $H$  become sharper and higher than those of not including the colour information of the neighbourhood and are easier to detect. This can be seen from the comparative experimental results shown in Figs. 5–20.

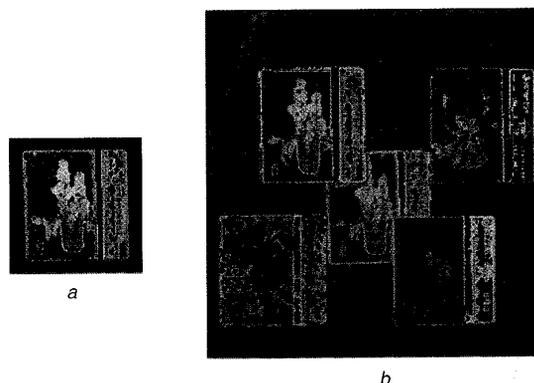
### 4 Experimental results

The proposed modified GHT algorithm has been implemented on a Sun Sparc 10 workstation and several images have been tested. The size of each input image is  $256 \times 256$  pixels. Some experimental results are shown in Figs. 5–20.

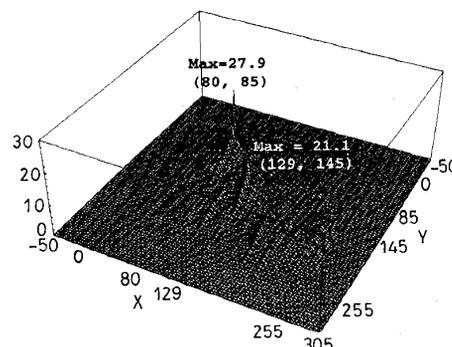
Fig. 5a is a colour image containing a telephone card to be used as a template. Fig. 5b is the telephone-card template, whose size is  $100 \times 100$  pixels, segmented from Fig. 5a. Fig. 5c is an input image containing two telephone cards fully or partially identical to the telephone-card template and several other kinds of card under different lighting intensities. Figs. 6a and 6b show the extracted critical pixels of Figs. 5b and 5c. After the proposed modified GHT is performed, two obvious peaks exist in the resulting HCS, as shown in Figs. 7 and 8, which locate the positions of two detected telephone cards. Fig. 7 shows the resulting HCS with the neighbouring colour information being considered, and Fig. 8 shows the resulting



**Fig. 5** Illustration of practical telephone-card detection  
 a Image containing a telephone-card template  
 b Telephone-card template segmented from a  
 c Input image containing two cards fully or partially identical to the card template and several other kinds of card under different lighting intensities

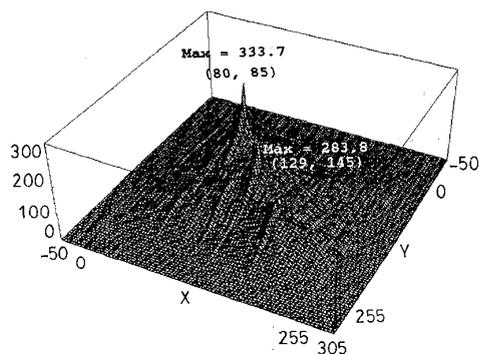


**Fig. 6** Illustration of practical telephone-card detection  
 a Extracted pixels from Fig. 5b  
 b Extracted pixels from Fig. 5c

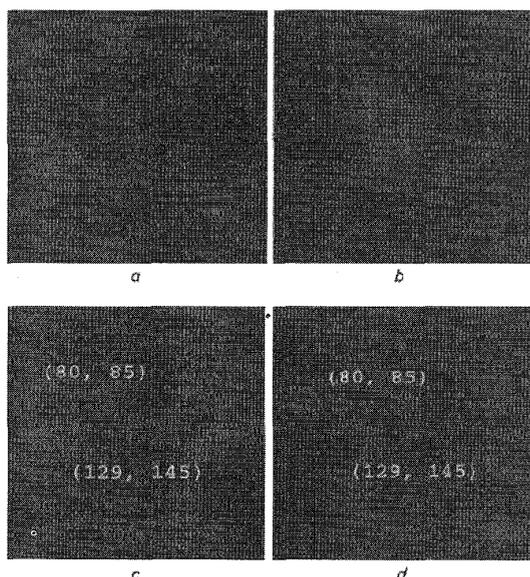


**Fig. 7** Illustration of practical telephone-card detection: resulting HCSs after performing the modified GHT while considering the neighbourhood colour information  
 The maximum value of the obvious peak, designated max, is also shown

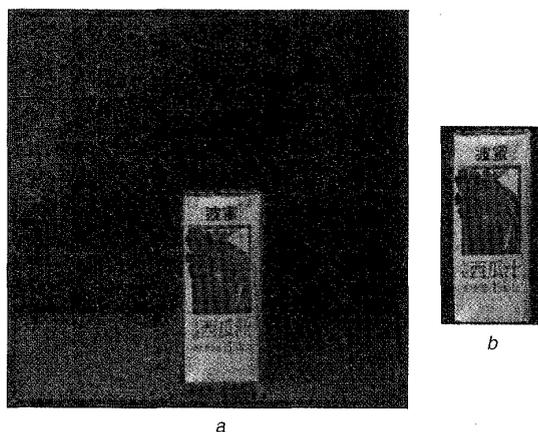
HCS without considering the neighbouring colour information, i.e. with  $S(P_i, P_j) = S_1$ . The HCS in Fig. 7 has two more obvious peaks than those in Fig. 8.



**Fig. 8** Illustration of practical telephone-card detection: resulting HCSs after performing the modified GHT without considering the neighbourhood colour information  
The maximum value of the obvious peak, designated max, is also shown. The HCS in Fig. 7 has two more obvious peaks, which locate the positions of the template cards, than those in Fig. 8

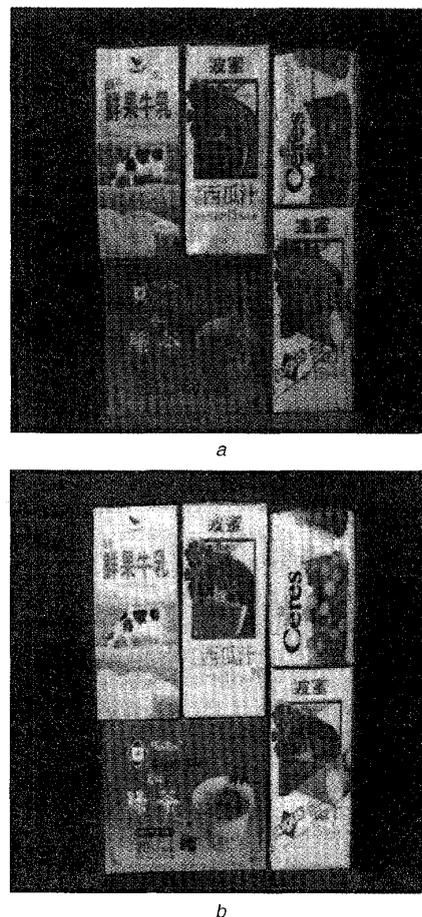


**Fig. 9** Illustration of practical telephone-card detection  
a Resulting HCSs in Fig. 7 as an image  
b Resulting HCSs in Fig. 8 as an image  
c Detected locations of template shape in a after thresholding  
d Detected locations of template shape in b after thresholding



**Fig. 10** Illustration of practical juice-box detection  
a Template image  
b Segmented template

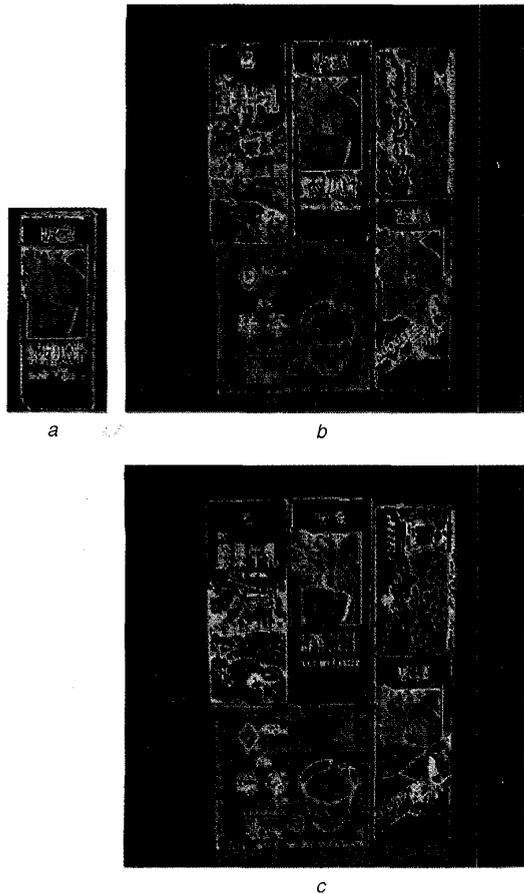
Owing to use of the different measure  $S(P_i, P_j)$ , the scales of counting values in Figs. 7 and 8 are different but the detected peaks in the HCSs are equally prominent. Figs. 9a and 9b show the resulting HCSs in Figs. 7 and 8 as images. Figs. 9c and 9d show the detected locations of the template shape in Figs. 9a and 9b after thresholding.



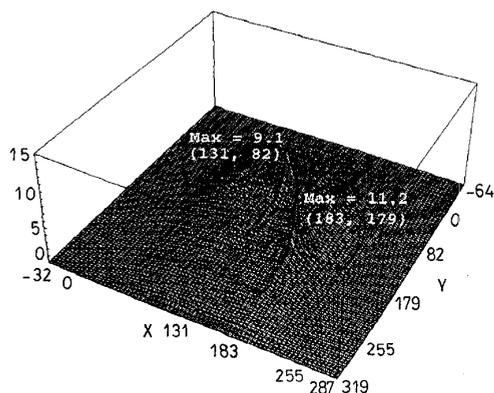
**Fig. 11** Illustration of practical juice-box detection  
a Input image containing two juice boxes fully or partially identical to the template  
b Input image containing two juice boxes fully or partially identical to the template but under a different lighting intensity to that used in b

Figs. 10–17 show another set of experimental results using juice-box images. Fig. 10a is a colour image containing a juice-box template. Fig. 10b is the template, whose size is  $64 \times 128$  pixels, segmented from Fig. 10a, and its extracted critical pixels are shown in Fig. 12a. Figs. 11a and 11b are two input images, each containing two boxes fully or partially identical to the template and several other kinds of juice boxes under different lighting intensities. Figs. 12b and 12c show the extracted critical pixels of Figs. 11a and 11b, respectively. After the modified GHT is performed on the images, Figs. 13 and 14 show the two resulting HCSs, each of which includes two obvious peaks indicating the positions of two detected templates. Figs. 15 and 16 show another set of resulting HCSs like Figs. 13 and 14 but without consideration of the neighbouring colour information. Figs. 17a and 17b show the candidates of the detected locations in Figs. 13 and 14, respectively, after thresholding using threshold value =  $0.55 * \text{global maximum value in HCS}$ . Similarly, Figs. 17c and 17d

show the candidates of the detected locations in Figs. 15 and 16, respectively, using the identical threshold value to that in Figs. 17a and 17b but more redundant or error candidates are detected because there is no consideration of the neighbouring colour information.



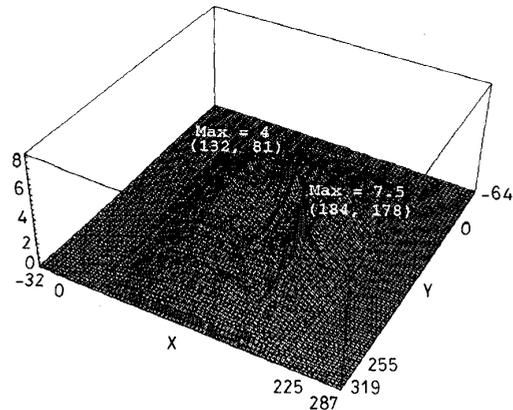
**Fig. 12** Illustration of practical juice-box detection  
*a* Extracted pixels from Fig. 10b  
*b* Extracted pixels from Fig. 11a  
*c* Extracted pixels from Fig. 11b



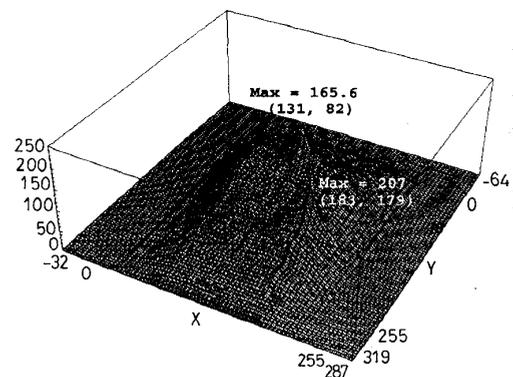
**Fig. 13** Illustration of practical juice-box detection: resulting HCSs after performing the modified GHT on Fig. 12a with consideration of neighbouring information

Figs. 18a and 18b are two other input images which are the same as Fig. 11a except that to the images are added some gaussian noise and spots as well and they are disturbed using some diffuse and wind filters. After

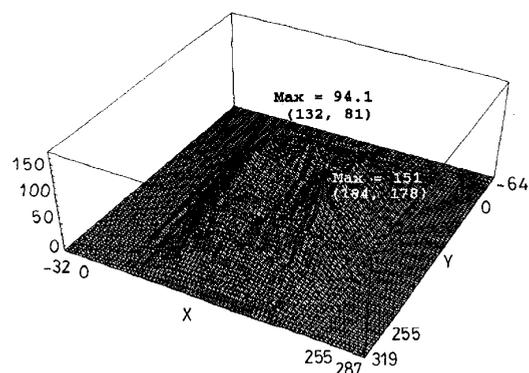
modified GHT has been performed on Fig. 18a with or without consideration of neighbouring colour information, the resulting HCSs are shown in Figs. 19a and 19b. After thresholding has been performed on Figs. 19a and 19b using threshold value =  $0.8 * \text{global maximum value in the HCS}$ , the candidates of the detected locations of template shape are shown in Figs. 19c and 19d, respectively. From the results, we see that more redundant candidates in Fig. 19d are detected than those in Fig. 19c because there is no consideration of neighbouring information. Figs. 20a-d are like Figs. 19a-19d except that they use image Fig. 18b and give the same conclusion.



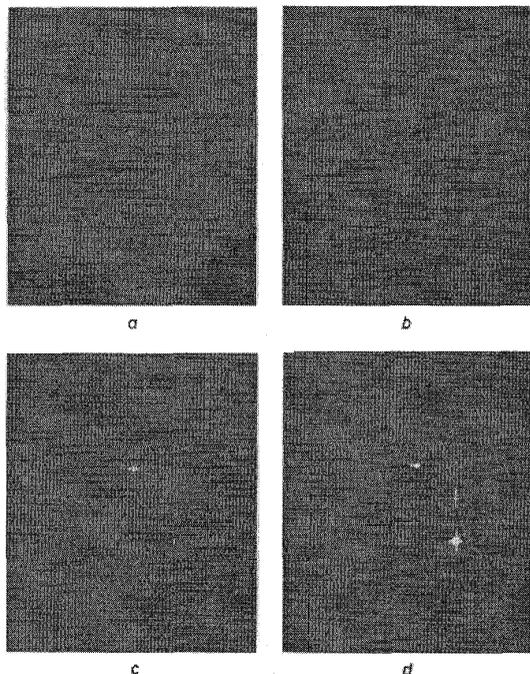
**Fig. 14** Illustration of practical juice-box detection: resulting HCSs after performing the modified GHT on Fig. 12b with consideration of neighbouring information  
 Both HCSs in Figs. 13 and 14 have two obvious peaks which locate the positions of two detected templates



**Fig. 15** Illustration of practical juice-box detection: resulting HCSs as in Fig. 13 but without considering neighbourhood information



**Fig. 16** Illustration of practical juice-box detection: resulting HCSs as in Fig. 14 but without considering neighbourhood information



**Fig. 17** Illustration of practical juice-box detection  
*a* Candidates of the detected locations in Fig. 13 after thresholding using threshold value =  $0.55 \times$  global maximum value in HCS  
*b* Candidates of the detected locations in Fig. 14 after thresholding using threshold value =  $0.55 \times$  global maximum value in HCS  
*c* Candidates of the detected locations in Fig. 15 using the identical threshold value to that in *a*; more redundant or error candidates are detected because there is no consideration of neighbourhood information  
*d* Candidates of the detected locations in Fig. 16 using the identical threshold value to that in *b*; more redundant or error candidates are detected because there is no consideration of neighbourhood information

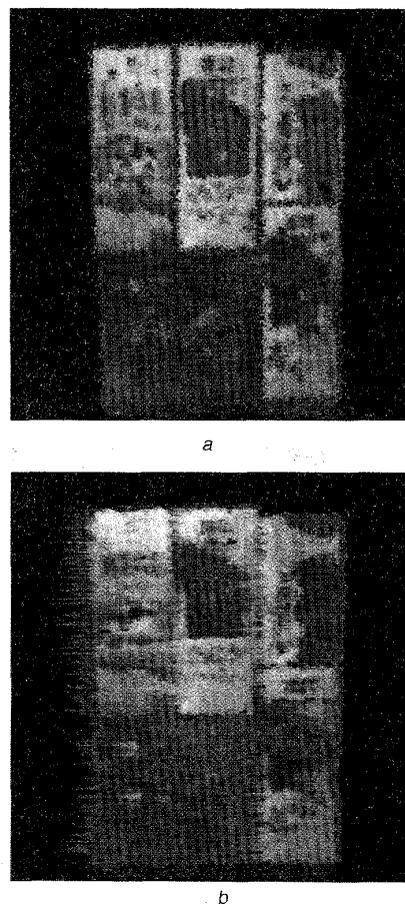
## 5 Conclusions and discussion

A new approach to colour-image detection and matching using the modified GHT in lighting-change environments has been proposed. Using normalised colour values, lighting changes in input images can be removed. Critical pixels rather than conventional boundary or skeletal pixels are extracted from colour images, and the colour information of each pixel and that of its neighbours is utilised. A modified voting rule in which cell-value incrementation is based on fractional similarity-measure values has therefore also been proposed. The experimental results showed that the proposed approach is robust for colour-image detection and matching in noisy, occlusive and lighting-change environments, and has high potential for practical applications.

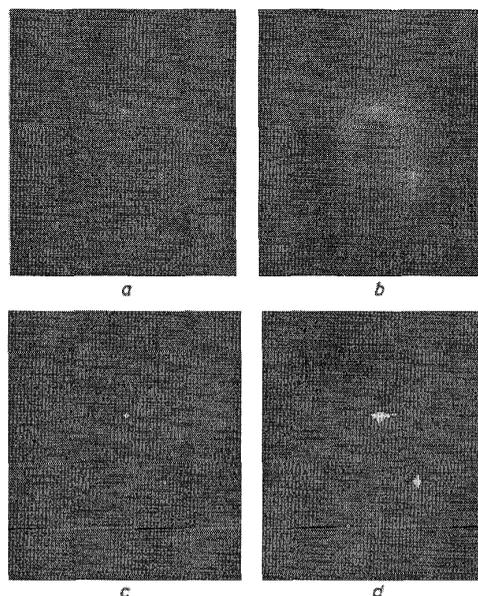
When the light sources are too dark or bright, the colour information will become unstable and the proposed method may fail. Likewise, as the light sources coming from lighting changes cannot be guaranteed to be white daylight, the proposed method may again fail. Although all the experiments in the proposed method involve only translation, it is possible to extend the method to scaling and rotation transformations using colour interpolation [18]. This will be a worthwhile topic for further study.

## 6 Acknowledgment

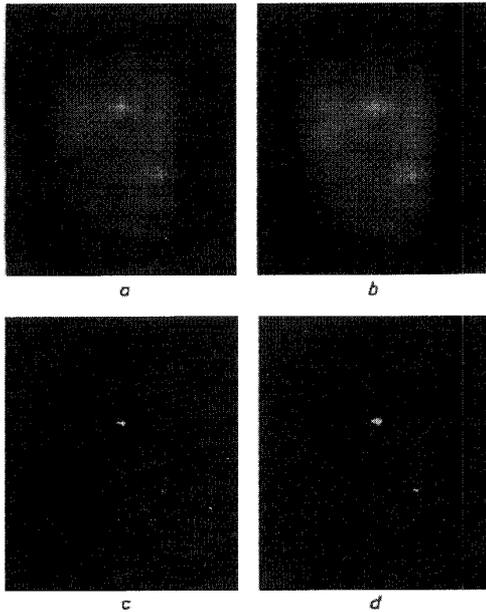
The work was partially supported by the National Science Council, Republic of China, under contract NSC82-0404-E009-157.



**Fig. 18** Comparing results of the proposed GHT for input-disturbed images  
*a* Noisy image with added Gaussian noise, spots and diffuse filter  
*b* Image distorted with wind and diffuse filter



**Fig. 19** Comparing results of the proposed GHT for input-disturbed images  
*a* Resulting HCSs shown as images after performing modified GHT on the image in Fig. 18*a* with consideration of neighbouring pixel information  
*b* Resulting HCSs shown as images after performing modified GHT on the image in Fig. 18*a* without consideration of neighbouring pixel information  
*c* Candidate of the detected locations of template shape in *a* after thresholding using threshold value =  $0.8 \times$  global maximum value in HCS  
*d* Candidate of the detected locations of template shape in *b* after thresholding using threshold value =  $0.8 \times$  global maximum value in HCS  
 More redundant candidates are detected in *d* than in *c* because there is no consideration of neighbouring information



**Fig.20** Comparing results of the proposed GHT for input-disturbed images

*a* Resulting HCSs shown as images after performing modified GHT on the image in Fig. 18*b* with consideration of neighbouring pixel information  
*b* Resulting HCSs shown as images after performing modified GHT on the image in Fig. 18*b* without consideration of neighbouring pixel information  
*c* Candidate of the detected locations of template shape in *a* after thresholding using threshold value =  $0.8 * \text{global maximum value in HCS}$   
*d* Candidate of the detected locations of template shape in *b* after thresholding using threshold value =  $0.8 * \text{global maximum value in HCS}$   
 More redundant candidates are detected in *d* than in *c* because there is no consideration of neighbouring information

## 7 References

1 HOUGH, P.V.C.: 'Method and means for recognizing complex patterns'. US Patent 3069654, 1962

2 DUDA, R.O., and HART, P.E.: 'Use the Hough transform to detect lines and curves in pictures', *Comm. Assoc. Comput.*, 1972, **15**, pp. 11-15

3 BALLARD, D.H.: 'Generalizing the Hough transform to detect arbitrary shapes', *Pattern Recognit.*, 1981, **13**, pp. 111-122

4 DAVIS, L.S.: 'Hierarchical generalized Hough transforms and line-segment based generalized Hough transform', *Pattern Recognit.*, 1982, **15**, pp. 277-285

5 LLLINGWORTH, J., and KITTLER, J.: 'A survey of the Hough transform', *Comput. Vis. Graph. Image Process.*, 1988, **44**, pp. 87-116

6 JENG, S.C., and TSAI, W.H.: 'Fast generalized Hough transform', *Pattern Recognit. Lett.*, 1990, **11**, pp. 725-733

7 WANG, H.L., and REEVES, A.P.: 'Three-dimensional generalized Hough transform for object identification', *J. Soc. Photo-Opt. Instrum. Eng.*, 1990, **1192**, pp. 363-374

8 JENG, S.C., and TSAI, W.H.: 'Scale- and orientation-invariant generalized Hough transform - a new approach', *Pattern Recognit.*, 1991, **24**, pp. 1037-1051

9 SANO, M., MEGURO, S., and ISHII, A.: 'Gray-level image recognition based on multiple cell-features', *Syst. Comput. Jpn.*, 1991, **22**, pp. 81-93

10 LEAVERS, V.F.: 'The dynamic generalized Hough transform: its relationship to the probabilistic Hough transforms and an application to the concurrent detection of circles and ellipses', *CVGIP, Image Underst.*, 1992, **56**, pp. 381-398

11 LI, Z.N., YAO, B., and TONG, F.: 'Linear generalized Hough transform and its parallelization', *Image and Vision Computing*, 1993, **11**, pp. 11-24

12 BHANDARKAR, S.M.: 'A fuzzy probabilistic model for the generalized Hough transform', *IEEE Trans.*, 1994, **SMC-24**, pp. 745-759

13 LEE, H.C., BRENEMAN, E.J., and SCHULTE, C.P.: 'Modeling light reflection for computer colour vision', *IEEE Trans. Pattern Anal. and Mach. Intell.*, 1990, **12**, pp. 402-409

14 YOUNG, T.: 'On the theory of light and colors', *Phil. Trans. Royal Soc. Lond.*, 1802, **92**, pp. 20-71

15 BORGES, C.E.: 'Trichromatic approximation method for surface illumination', *J. Opt. Soc. Am. A*, 1991, **8**, pp. 1319-1323

16 PRATT, W.K.: 'Digital image processing' (Wiley Interscience, New York, 1991), 2nd ed., pp. 316-320

17 HARALICK, R.M., and SHAPIRO, L.G.: 'Computer and robot vision, vol. II' (Addison Wesley, New York, 1993), pp. 1-36

18 WATT, A.: 'Fundamentals of three-dimensional computer graphics' (Addison Wesley, New York, 1990), pp. 309-354