

# AN ALTERNATIVE TO GRAPH MATCHING FOR LOCATING OBJECTS FROM THEIR SALIENT FEATURES

E.R. Davies

Machine Vision Group, Department of Physics  
Royal Holloway & Bedford New College  
Egham Hill, Egham, Surrey, TW20 0EX, UK

---

*The local-feature-focus method has become a standard means for robustly locating objects in two dimensions. Yet it is not without its difficulties, since the maximal clique approach to graph matching which it employs is excessively computation intensive, belonging to the class of NP-complete problems. Here we explore whether similar results can be obtained using other approaches, and in particular with the generalised Hough transform. The latter approach is found to be essentially equivalent to graph matching, while permitting objects to be located in polynomial ( $O(n^4)$ ) time.*

---

## 1. INTRODUCTION

In many branches of image analysis, including particularly that of automated visual inspection, it is necessary to locate objects both rapidly and robustly in digital images. During the past decade or so it has been found that the Hough transform (HT)<sup>1</sup> provides a sound basis for achieving these aims. A significant breakthrough occurred when it was found that edge orientation information could be used to cut down the computation requirements of the HT<sup>2</sup>. The Ballard version of the HT achieved this for objects of general shape, even in cases where the latter cannot be described analytically<sup>3</sup>. The method is not ideal, since computational load can still be considerable when object orientations are unknown. This fact has induced many workers either to improve on the generalised Hough transform (GHT)<sup>4</sup> or to turn towards other techniques.

One such technique is the local-feature-focus (LFF) method of Bolles and Cain<sup>5</sup>. This method is particularly useful in cases where objects have certain strong features such as corners and holes by which they may be located and identified. Indeed, if objects possess high contrast features such as these, then it should be highly efficient to use them for object detection - the reason being that the computation involved in searching an image decreases with the size of the template used. However, this assumes that it is trivial deducing the presence of an object once the set of image features is known. Such an assumption is far from the whole truth, though it is a useful starting approximation. Early on, a graph-theoretic approach seemed appropriate for matching observed features to idealised object features<sup>6,7</sup>: later this developed into the more sophisticated LFF method<sup>5</sup>.

The LFF method is designed to work well even in industrial applications where parts frequently overlap one another, or where they may be defective or distorted: in fact the method has become an oft-quoted and standard

means for locating objects in two dimensions. However, the technique has certain problems, since the maximal clique approach to graph matching which it employs can be excessively computation intensive. This raises the question of whether better results could be obtained by other means. Here we attempt to answer this question, and in particular to compare the LFF and GHT schemas. The actual comparison is carried out in section 4, sections 2 and 3 being devoted to respective preliminary studies of the two methods.

## 2. GRAPH-THEORETIC APPROACHES TO OBJECT LOCATION

If objects appear on a worktable or conveyor at a known distance from the camera, and if also they are flat or can appear in only a restricted number of stances in three dimensions, it will be apparent that they may be identified and located from a minimum number of small features. Clearly, one such feature is insufficient, but two features can in principle lead to positive identification if the features are distinguishable and their distance apart is known<sup>8</sup>. Suitable distinguishable features would be holes of different sizes, or a hole and a corner. Even if the features are indistinguishable, two of them may be sufficient to locate an object if they are anisotropic - for example, corners. However, two identical isotropic features will leave a single ambiguity about the location of an object unless it possesses 180° rotation symmetry.

In some ways it is more satisfactory to imagine that three features are necessary to lift the ambiguity referred to above, when objects are to be located and identified at a known range. Unfortunately, this scheme of things is too simplistic in many applications as it is insufficiently robust against distortions and occlusions. In particular, camera and other optical (e.g. perspective) distortions may arise, or the objects themselves may be distorted, or by resting partly on other objects they may not be quite in the assumed stance: such problems mean that distances between features may not be exactly as expected. Furthermore, objects may be partly obscured by other objects resting on them, or else they may be broken or otherwise defective. All these factors mean that as many features as possible should be taken into account in locating and identifying objects. The maximal clique approach<sup>6,7</sup> is intended to achieve this.

As a start, as many features as possible are identified in the off-camera image, and these are numbered in some convenient order, e.g. the order of appearance in a normal TV raster scan. The numbers then have to be matched against the letters corresponding to the features on an ideal object. For simplicity we here assume that only one type of object is being sought though several instances of it may appear in the input field.

The next step is to attempt to match each observed feature (number) against each ideal feature (letter), each such assignment then having to be checked for consistency, e.g. according to the distance between the two features, and the relative orientation of the feature and the line joining the features (the latter situation being applicable for corners but not for holes). A systematic way of achieving this is by constructing a *match graph* in which the nodes represent feature assignments, and arcs joining nodes represent pairwise compatibilities between assignments. To find the best match it is then necessary to find regions of the match graph where the cross-linkages are maximised.

To this end *cliques* are sought within the match graph. A clique is a *complete subgraph* - i.e. one for which all pairs of nodes are connected by arcs. However, if one clique is completely included within another clique, it is likely that the larger clique represents a better match. Hence *maximal cliques* are taken as leading to the most reliable matches between the observed image and the object model<sup>6,7</sup>.

Bolles and Cain used essentially this technique to locate hinges in digital images<sup>5</sup>. Hinges are characterised by 12 corners (8 convex and 4 concave) and four holes (of equal size). In this case the algorithm for finding maximal cliques became a computational bottleneck in the system, so they took the short cut of using certain starting features known as focus features: only a limited number of features near any given focus feature were then sought. We illustrate the situation for a general triangle (Figure 1). For simplicity of illustration we assume that the observed image contains only one triangle, that lengths match exactly, and that no occlusions occur. The match graph in this example is shown in Figure 2. There are 9 possible feature assignments, 6 valid compatibilities and 4 maximal cliques of sizes 3,1,1,1, only the largest corresponding to an exact match. (Here we define the *size* of a clique as the number of its compatibilities.) If occlusion of even one feature had occurred, there would have been only 6 possible feature assignments, 2 valid compatibilities and each of these would have been maximal cliques: the result would have been an ambiguity in the location of triangle - as remarked earlier. In such an example the checking of which subgraphs are maximal cliques is a trivial problem. However, in real matching tasks it can quickly become unmanageable. Another example is provided by Bolles<sup>7</sup>: in this case an engine cover is being sought in an image containing a whole engine cover and another engine cover, only part of which appears within the confines of the image (this is one type of partial occlusion). Here 17 holes are visible within the image, an ideal engine cover possessing 14 holes. In addition some of the holes are large and some small. There are 2x3 possible assignments of large holes and 12x14 possible assignments for the small holes; hence the match graph contains 172 nodes\*, and  $\binom{172}{2} = 14,706$  pairwise consistency checks must be made to build the match graph. Bolles estimates that at best it will take tens of seconds on a DEC KL-10 class computer to build the graph and locate the largest maximal clique<sup>7</sup>.

Several algorithms have been written for finding maximal cliques<sup>6,7</sup>. It seems highly likely that the best of these is close to the optimum speed. The reason for the time taken being large is not the inefficiency of the particular algorithm but the fact that it is not possible to devise a maximal clique algorithm that runs in polynomial time for a given number of nodes. Indeed, it has been shown that

the maximal cliques problem is "NP-complete", which implies that it runs in exponential time<sup>9</sup>. Hence short cuts of the type mentioned above are required to help identify maximal cliques in most real applications.

In the next section we take a preliminary look at the GHT before going on in section 4 to examine it as an alternative to the maximal clique approach.

### 3. THE GENERALISED HOUGH TRANSFORM APPROACH TO OBJECT LOCATION

We begin this section by outlining the standard Hough method for circle detection<sup>2</sup>. In this method all edge fragments in an image are first located; then votes are accumulated in a parameter space congruent to image space at positions which are a distance R equal to the expected circle radius along the local edge normals; parameter space is then found to contain clusters of candidate centre points, and peak location is used to find the most likely positions of circle centres. Locating peaks in parameter space is not a trivial task as the peak is often fragmented by noise - especially that arising from aliasing round the periphery of the object. The technique requires local edge orientation to be estimated quite accurately<sup>10</sup>: this can be achieved to within  $\sim 1^\circ$  using the Sobel operator<sup>11</sup>. Note that the amount of computation is minimised since the way edge orientation information is taken into account cuts down the numbers of votes cast in parameter space. This approach to circle location is important in that it is highly robust against occlusions, defects, distortions and noise.

The GHT aims to apply the same basic techniques when objects have arbitrary shapes. In this case votes are accumulated at an object localisation point L, which is found not at a fixed distance R along the local edge normal, but at a variable distance R in a variable direction  $\phi$ , where the values of R and  $\phi$  are functions of local edge orientation  $\theta$ : their values are obtained either analytically or from a suitable lookup table called the R-table<sup>3</sup>. Details of how the R-table is constructed are beyond the scope of this paper. However, the method is in principle capable of locating objects whose shapes are analytically defined (such as ellipses) or arbitrary shapes (cams, gaskets, etc.). The method is also applicable to polygons including squares, rectangles, and so on. The main disadvantage of the method is that if the object orientation is unknown, it has to be introduced as an additional parameter (i.e. an extra dimension) in parameter space: this significantly increases the numbers of votes cast in parameter space and the size of the space that has to be searched for peaks. Means of alleviating this problem are still being devised, though it is now known how to optimise the situation in the case of polygon detection<sup>4</sup>.

---

\* According to the author's calculations this number (from<sup>7</sup>) should read 174 (= 2x3 + 12x14), with a consequent change in the number of consistency checks.

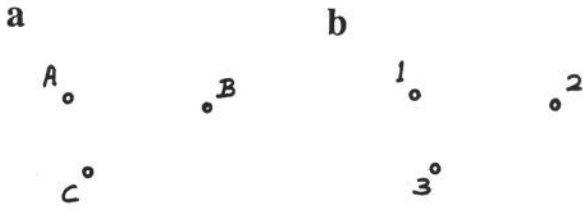


Figure 1. A simple matching problem - a general triangle.  
 a - model template;  
 b - image.

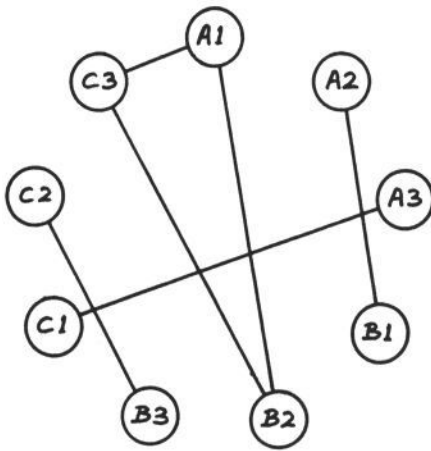


Figure 2. Match graph for a general triangle.  
 The maximal cliques are: (1) A1, B2, C3; (2) A2, B1;  
 (3) B3, C2; and (4) C1, A3. Although the respective  
 cliques have 3,2,2,2 nodes, we here measure the sizes of  
 cliques by the numbers of arcs; in this case they are  
 3,1,1,1. In general for a clique of  $v$  nodes there are  $\binom{v}{2}$   
 arcs.

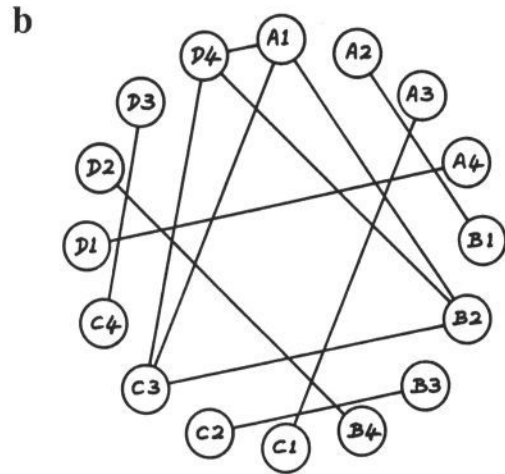
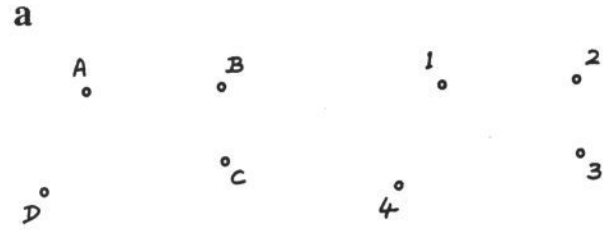
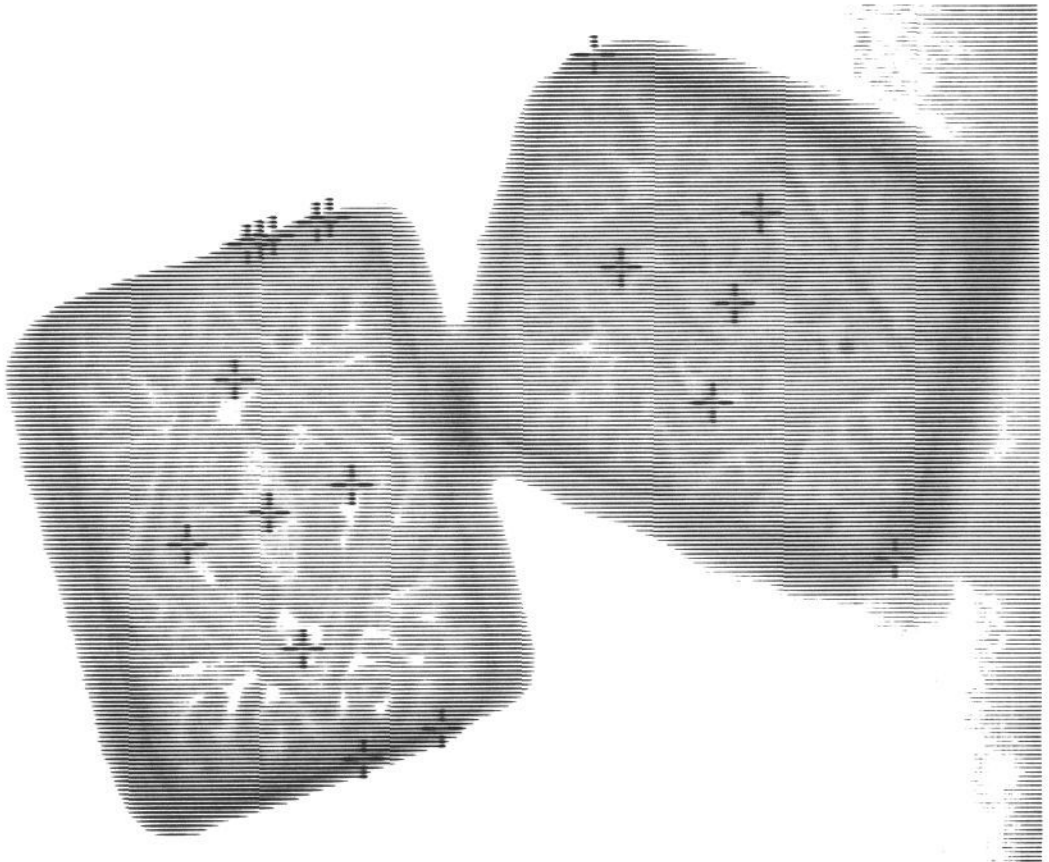


Figure 4. Another matching problem - a general quadrilateral.  
 a - basic labelling of model (left) and image (right)  
 b - match graph  
 c - placement of votes in parameter space (notation as  
 in Figure 3).

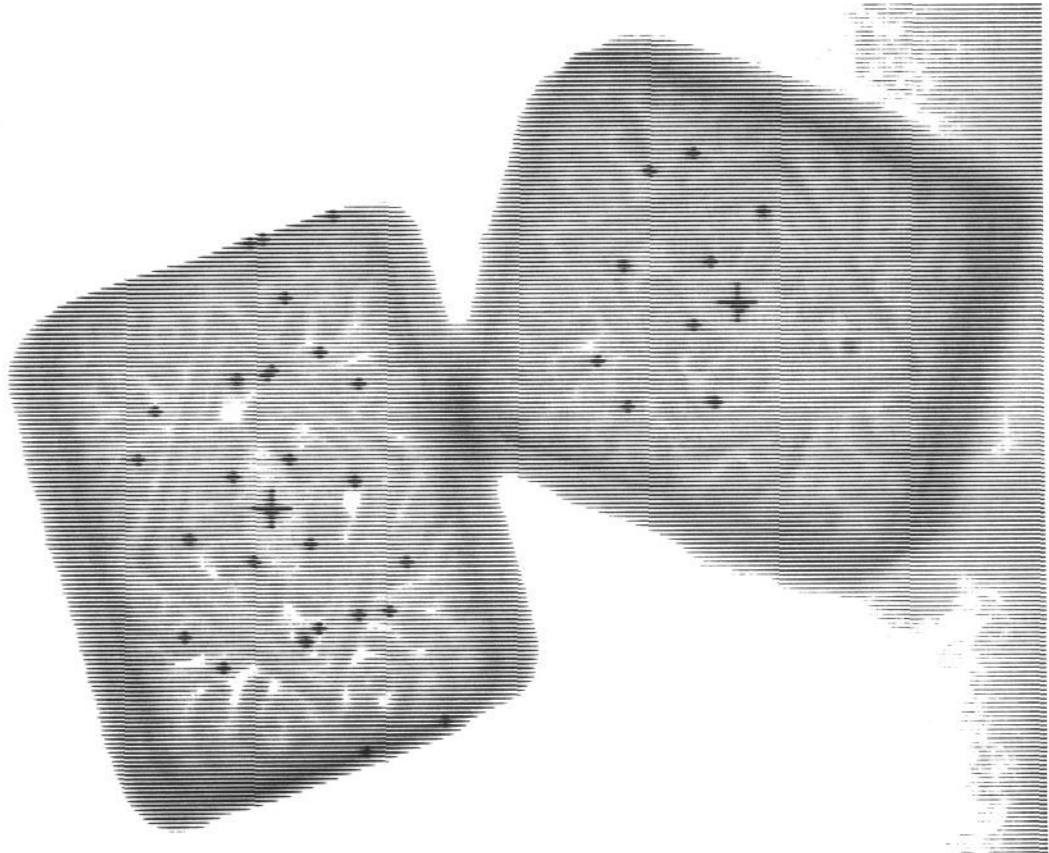
Figure 3. Placement of votes in parameter space for a  
 general triangle.

- o - positions of observed features
- . - positions of votes
- - position of main voting peak.

**a**



**b**



*Figure 5. Cream biscuits located using the GHT.*

(a) shows an off-camera 256x256 image of two cream sandwich biscuits with crosses indicating the result of applying a simple hole detection routine.

(b) shows the two biscuits reliably located by the GHT from the hole data in (a). The isolated small crosses indicate the positions of single votes.



The GHT retains the advantages of the HT circle detector described above in being highly robust against occlusions and object defects. Indeed, it is known to be a type of spatial matched filter, so it is in this sense optimal in its noise rejection characteristics<sup>12</sup>. These facts make it attractive to try out the GHT as a means of collating information from small features on objects, as a possible alternative to graph-theoretic approaches.

#### 4. USING THE GHT TO COLLATE INFORMATION FROM SMALL FEATURES

In this section we analyse how the GHT can be used as an alternative to the maximal clique approach, to collate information from small features in order to locate objects robustly in digital images. In order to simplify the situation and the comparisons we shall make between the two approaches, we consider initially situations where objects have no symmetries and where the features are isotropic and indistinguishable. This situation would apply exactly when all features are holes of a specific size, and all inter-hole distances are different - as for the general triangle of Figure 1.

To apply the GHT for analysing feature information we adopt the strategy of listing all features and then accumulating votes in parameter space at every possible position of L consistent with each *pair* of features. This strategy is particularly suitable in the present context, as it corresponds to the pairwise assignments used in the maximal clique method, and allows us to compare like with like when the two methods are finally assessed. To proceed we have merely to use the inter-feature distance as a lookup parameter in the GHT R-table. For isotropic features this means that we have to have two entries for the position of L for each value of the inter-feature distance. Note that we simplified the analysis by assuming that no symmetries exist and that all pairs of features have different inter-feature distances. If this were not so, then more than two vectors would have to be stored in the R-table per inter-feature distance value.

We are now in a position to give a simple example of how the GHT can be applied to feature collation. We start with the triangle example of Figure 1. Figure 3 shows the positions at which votes are accumulated in parameter space. There are 4 peaks, with heights of 3,1,1,1, it being clear that, in the absence of complicating occlusions and defects, the object is locatable at the peak of maximum size.

Figure 4 shows the situation for a general quadrilateral, (a) giving the basic labelling, (b) giving the match graph and also showing the 12 valid compatibilities and the 7 maximal cliques of sizes 6,1,1,1,1,1, and (c) showing the result of applying a GHT. In the latter case there are 7 peaks of sizes 6,1,1,1,1,1. Close examination of Figures 1-4 indicates that every peak in parameter space corresponds to a maximal clique in the match graph. Indeed, there is a one-to-one relation between the two. In the uncomplicated situation we are examining here this is bound to be so for any general arrangement of features within an object, since every pairwise compatibility between features begets two object locations, one correct and one that can be correct only from the point of view of that pair of features. Hence the correct ones all add to give

a large maximal clique and a large peak in parameter space, whereas the incorrect ones give maximal cliques each containing two wrong assignments and each corresponding to a false peak of size 1 in parameter space.

We assumed earlier that the objects being detected possess no symmetries. If there are symmetries the situation becomes slightly more complex, and a modified form of the maximal clique approach is required. Space does not permit a full analysis of the situation to be carried out here, but we have concluded that the GHT and maximal clique approaches are still essentially equivalent.

When some of the features of an object are missing (e.g. because they are occluded), the correspondence between the two methods is retained: indeed, it is found that occlusion does not make object identification any more complicated or liable to error, for either method. Similarly, if any additional features (e.g. from other nearby or overlapping objects) are present in the image, they do not in general interfere with interpretation: if the largest maximal clique leads to a correct identification, then so does the highest peak in parameter space, and vice versa.

Finally, we note that each of the methods can readily be extended to situations where the features are corners or directed corners instead of holes. Even with this change the two methods remain essentially equivalent.

#### 4.1. Results obtained using the GHT

Figure 5(a) shows a pair of cream biscuits which are to be located from their "docker" holes - this strategy being advantageous since it has the potential for highly accurate product location prior to detailed inspection. The holes detected by a simple template matching routine are also indicated in Figure 5(a); the template used is rather small and as a result the routine is fairly fast but fails to locate all holes and in addition gives false alarms. Hence is it required to use an "intelligent" algorithm to analyse the hole location data.

Figure 5(b) shows the positions of candidate object centres as found by the GHT. The small isolated crosses indicate the positions of single votes, and those very close to the two large crosses lead to voting peaks of weights 10 and 6 at these respective positions. Hence object location is both accurate and robust as required.

#### 4.2. Computational load

In this sub-section we examine the computational load of the two approaches to object location. For simplicity we imagine an image that contains just one wholly visible example of the object being sought. Suppose the object possesses  $n$  features and that we try to recognise it by seeking all possible pairwise compatibilities, whatever their distance apart (as in all examples in the main part of section 4). (Note that short cuts of the type mentioned in section 2 can validly be applied in both approaches, with no major change in the final comparison.)

For an object possessing  $n$  features, the match graph contains  $n^2$  nodes (i.e. possible assignments), and there are  $n^2(n^2-1)/2$  possible pairwise compatibilities to be checked in building the graph. A possible assumption is that 50% of the pairwise compatibilities will hold<sup>5</sup>, but even if this

is not so, the amount of computation at this stage of the analysis is  $O(n^4)$ . To this must be added the cost of finding the maximal cliques. Many authorities assert that this problem is NP-complete and therefore is non-polynomial and probably exponential in  $n^{7,9}$ .

Now let us see the cost of getting the GHT to find objects via pairwise compatibilities. As we have seen, total height of all peaks in parameter space is always equal to the number of pairwise compatibilities in the match graph. Hence the computational load is of the same order,  $O(n^4)$ . Next comes the problem of locating all the peaks in parameter space. In this case parameter space is congruent to image space. Hence for an  $N \times N$  image only  $N^2$  points have to be visited in parameter space and computational load is  $O(N^2)$ .

## 5. CONCLUDING REMARKS

This paper has shown that, apart from certain complications arising from symmetry, the maximal clique approach to object identification and location is equivalent to the GHT approach. However, whereas the maximal clique approach takes  $O(n^4)$  time to build the match graph and exponential time to analyse it for maximal clique solutions, the GHT takes  $O(n^4)$  time to build parameter space, and then  $O(N^2)$  time to locate the object. We are therefore forced to conclude that the graph theory formalism is not so well matched to the relevant real-space template matching task as the GHT.

## Acknowledgement

The author is grateful to the UK Science & Engineering Research Council for financial support during the course of this research.

## REFERENCES

1. Hough, P.V.C. "Method and means for recognising complex patterns" U.S. Patent 3069654 (1962).
2. Kimme, C., Ballard, D. and Sklansky, J. "Finding circles by an array of accumulators", *Comm.Assoc.Comput.Mach.*, 18 No.2 (1975) pp 120-122.
3. Ballard, D.H. "Generalising the Hough transform to detect arbitrary shapes" *Pattern Recogn.* 13 No.2 (1981) pp 111-122.
4. Davies, E.R. "Reduced parameter spaces for polygon detection using the generalised Hough transform" *Proc. 8th Int.Conf. on Pattern Recogn. Paris* (1986) pp 495-497.
5. Bolles, R.C. and Cain, R.A. "Recognizing and locating partially visible objects: the local-feature-focus method" *Int.J.Robot.Res.*, 1 No.3 (1982) pp 57-82.
6. Ambler, A.P., Barrow, H.G., Brown, C.M., Burstall, R.M. and Popplestone, R.J. "A versatile system for computer-controlled assembly" *Artif.Intell.* 6 (1975) pp 129-156.
7. Bolles, R.C. "Robust feature matching via maximal cliques" *SPIE 182 Imaging Applications for Automated Industrial Inspection and Assembly* (1979) pp 140-149.
8. Davies, E.R. "A glance at image analysis - how the robot sees" *Chartered Mechanical Engineer* (1984) pp 32-35.
9. Gibbons, A. "Algorithmic graph theory" *Cambridge University Press* (1985).
10. Davies, E.R. "The performance of the generalised Hough transform: concavities, ambiguities and positional accuracy" *Proc. 3rd Alvey Vision Conference, Cambridge* (1987) pp 327-333.
11. Davies, E.R. "Circularity - a new principle underlying the design of accurate edge orientation operators" *Image and Vision Computing* 2 No.3 (1984) pp 134-142.
12. Davies, E.R. "A new framework for analysing the properties of the generalised Hough transform" *Pattern Recogn.Lett.*, 6 No.1 (1987) pp 1-7.