

Indexing Two-Dimensional Objects Using Parametrised Geometric Features

C. C. Hand
I.T. Research Institute, Brighton Polytechnic
Brighton, U.K.

Abstract

A new approach to indexing in model-based vision is introduced; using indexing features whose deformed instances are present in a number of different object models. Hypotheses are generated by matching an indexing feature with an image using a viewing transformation incorporating the deformation; the parameter values of the deformational component provide specific indices to those models related to the indexing feature.

1 Introduction

Given a model-based vision (MBV) system which has a number of representations of different objects, how can it access the most appropriate model to match with a scene given that there is no *a priori* knowledge of the objects within it? This is the problem of indexing. One solution is to use indexing features, subsets of model descriptions which are computationally less expensive to match with an image, to generate hypotheses. Two types of indexing feature have been used; 'critical' features - features unique to specific objects [3, 4, 5], and 'similar' features - features common to a number of objects [6, 7]. This paper advocates using deformable indexing features whereby deformed instances are present in a number of quite different object models. As with similar features this has the advantage of representational parsimony. Furthermore, matching a feature with an image using a viewing transformation which has a deformational component can provide different indices for each model related to that feature via the specific parameter values of the deformation. This retains the specificity of the critical features approach.

The structure of this paper is as follows. The relationship between critical and similar features is first clarified and then extended to indexing via the structural deformations of parametrised features using anisotropic scaling. A specific implementation in the domain of 2-D, wire-frame, drawings is then discussed and a representative example is given which demonstrates the system's good performance.

2 Indexing by Critical and Similar Features

Traditionally, object recognition has been regarded as finding the parameters of some viewing transformation between some known object and an image description e.g. [1, 2].

Let I be an image description and M be an object model description, both being comprised of a set of geometrical primitives i.e. $I = \{p_{i1}, p_{i2}, \dots, p_{ir}\}$, and $M = \{p_{m1}, p_{m2}, \dots, p_{ms}\}$ where p_{ij} and p_{mk} are the j th and k th image and model primitives respectively. Object recognition is then characterised by the relation:

$$M \times I \supseteq C = \{(p_{ij}, pm_k) : T(p_{ij}) \approx pm_k\} \quad (1)$$

C is a correspondence between M and I ; it is a non-empty subset of their Cartesian Product. T is a geometrical transformation which superimposes a model primitive onto an image primitive. Thus, C is a relation between the two representations such that elements from M and I are related only if they are *geometrically compatible* with each other under some common viewing transformation. The parameters of T must be similar for every element of C so that each pair mutually supports and constrains the solution spaces of every other pair. Though suitable for the 'bin-of-parts' scenario it presents problems when there is a database of models $D = \{M_1, M_2, \dots\}$. If there is no *a priori* knowledge of what object is present in a scene then one must exhaustively search across D ; looking for the model which gives the best measure of fit. This is a major problem for MBV.

Two different types of model-driven indexing schemes have been suggested to reduce this search space. Both use indexing features which are related to particular models but which are simpler structures than the models themselves. They are generally used in hypothesis-verification schemes where a good match with a feature constitutes a good hypothesis which must be verified by a full model match. Such a feature match is less computationally expensive to match with the image than the full model description. The cost of hypothesis verification will also be reduced due to the information provided by the parameters of the viewing transformation of the hypothesis.

The first uses critically distinguishing features for each different object class i.e. there is a subset of each model description which is unique to it e.g. [3, 4, 5]. This gives a set of indexing features $F = \{f_1, f_2, \dots\}$ where $M_k \supseteq f_k$ and $\forall M_i \in D, i \neq k, M_i \cap T(f_k) = \emptyset$ and T is usually a rigid or a similarity transformation.

The second method uses 'similar' indexing features; features common to a number of object classes e.g. [6, 7]. Each similar feature is a subset of a number of different models; it represents the shared structure between them. Thus a similar feature is a correspondence, as defined in (1) above, but between two or more models i.e.

$$M_i \times M_m \supseteq f_k = \{(pm_{ij}, pm_{mn}) : pm_{ij} \in M_i, pm_{mn} \in M_m, T(pm_{ij}) \approx pm_{mn}\} \quad (2)$$

where pm_{mn} is the n th primitive of model M_m . Again T acts as a normalisation process. This can be extended to more than two models by recursively applying (2) but with the correspondence being expressed over the indexing feature and some other model. (The only restriction on this process is that the resulting description of the indexing feature should carry enough information such that it can be detected in some reasonably complex image.) A good match with a similar feature indicates a set of possible model hypotheses. Each hypothesis must then be tested by a full model match with each model associated with that feature; though each match will be highly constrained by the parameters of the viewing transformation for the indexing feature.

Both schemes have the disadvantage of exhaustive search across the set of indexing features during hypothesis generation. Though there will be fewer 'similar' indexing features than 'critical' features (due to the sharing of features between different models) this is at the cost of exhaustive search during hypothesis verification. Therefore, one must restrict the number of models that each similar feature indexes; one must find some

balance between the generality and uniqueness of a feature [6, 7]. Furthermore, a system must make active use of both the similarities and differences holding between objects. Similar features model similarities for indexing but 'passively' use differences during hypothesis verification and vice-versa for critical features. However, it would be helpful to make use of structural differences and similarities during indexing simultaneously.

3 Indexing via Structural Deformations

In 2D MBV it is generally assumed that the viewing transformation between two representations is either a rigid or a similarity transformation. However, one can use a 'non-similarity' transformation, such as anisotropic scaling or shear, as part of the viewing transformation between two representations. Features can be formed as in (2) above but with a deformational component as part of T . Thus the correspondence between models yields an indexing feature which can map onto different phenomenal forms via the different values of the deformational parameters of T i.e. an indexing feature can be shared across quite different object classes. This 'canonical feature' will be related to each deformed model instance and the index from the feature to each indexed model is given by the parameter values of the deformation. Thus, like 'similar' features, the same indexing feature could index a number of different models. However, the parameter values of the non-similarity part of T can be used to provide critically distinguishing indices. In general there is no need to exhaustively search all of the models associated with an indexing feature. In those cases where the phenomenally same feature exists in more than one model the indexing feature behaves as if it were a similar indexing feature type, but only for those models sharing the same parameter values for the deformation. Thus the similar indexing feature approach is a special case of indexing via structural deformations.

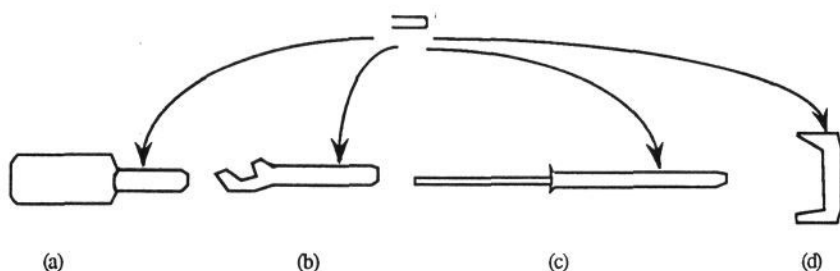


Figure 1: Deformed instances of the canonical indexing feature 'handle' appear in the object models (a) to (d) which are the 'bottle', 'c-spanner', 'screwdriver', and 'bracket' respectively. The specific parameter values of the four different deformations of the 'handle' provide the indices from that feature to the four models.

Consider the example of the different shapes in Figure 1; there are four different models and a feature. Each model contains an anisotropically scaled instance of the 'handle' - the five line segments furthest to the right in each model. Thus the 'handle' can be thought of as a stretchable feature. Each instance of the 'handle' within the individual models is different, both with respect to the canonical indexing feature and from each other. However, each instance is related to every other as any instance can map onto any other. Thus the 'handle' represents the shared structure between the different object classes

whilst the different phenomenal instances resulting from the different parameters used in the deformation represents the differences between the shared structures. This satisfies the condition that an indexing feature simultaneously reflects both the structural similarities and differences existing between forms.

4 Anisotropic Scaling

The 'non-similarity' transformation considered here is that of anisotropic scaling in the context of 2-D wire-frame drawings. The primitives are directed line segments, $\overrightarrow{(x_1, y_1)(x_2, y_2)}$. Let the transformation $T_{\alpha\beta}$ be the mapping:

$T_{\alpha\beta}: \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ such that

$$T_{\alpha\beta}(\overrightarrow{(x_1, y_1)(x_2, y_2)}) = \overrightarrow{(\alpha x_1, \beta y_1)(\alpha x_2, \beta y_2)} \quad \alpha, \beta \in \mathfrak{R} \quad (3)$$

A transformed feature is simply the set of all directed line segments comprising the structural description of the indexing feature i.e. $f_i = \left\{ \overrightarrow{(\alpha x_1, \beta y_1)(\alpha x_2, \beta y_2)} \right\}_k$.

Consider the special case $T_{\alpha\beta}$ where $\alpha = \beta$. Here anisotropic scaling mimics the similarity transformation of uniform scaling. To avoid confusion it is best to define this transformation separately:

$T_\lambda = T_{\lambda\lambda}: \mathfrak{R}^2 \rightarrow \mathfrak{R}^2$ such that

$$T_\lambda(\overrightarrow{(x_1, y_1)(x_2, y_2)}) = \overrightarrow{(\lambda x_1, \lambda y_1)(\lambda x_2, \lambda y_2)} \quad \lambda \in \mathfrak{R} \quad (4)$$

Both $T_{\alpha\beta}$ and T_λ can map into the same parameter space P as T_λ is merely a special case of $T_{\alpha\beta}$. Let a feature be deformed according to (3) and then two different uniformly scaled versions be produced with the scaling factors, λ_1 and λ_2 . The parameters of the composite transformations give the points $(\lambda_1\alpha, \lambda_1\beta)$ and $(\lambda_2\alpha, \lambda_2\beta)$ in P . There is an equivalence between the two forms as defined by the ratio of the scale factors:

$$\frac{\lambda_1\beta}{\lambda_1\alpha} = \frac{\beta}{\alpha} = \frac{\lambda_2\beta}{\lambda_2\alpha} \quad (5)$$

Formally, let \sim be an equivalence relation on the set P . We define the equivalence class $\overline{p_1}$ of an element $p_1 \in P$ to be the subset

$$\overline{p_1} = \{ p_2 \in P: p_1 \sim p_2 \} \quad (6)$$

where $p_1 = (\alpha_1, \beta_1)$, $p_2 = (\alpha_2, \beta_2)$, α_1 and $\alpha_2 \neq 0$, and \sim is defined as the ratio in (5).

Thus all uniformly scaled instances of some deformed structure form an equivalence class and one can choose a particular instance to act as the class representative. The equivalence class for each deformed instance of a feature forms a line through the origin and the parameter values of the class representative in P . The equivalence classes of the models in Figure 1 are shown in Figure 2.

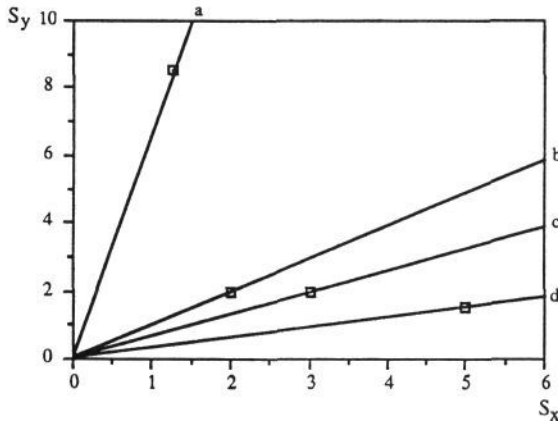
To index one finds the distance between the point in P defined by the correspondence between an indexing feature and an image, and each point in P defined by the correspondence between the indexing feature and the class representative of each deformed instance of that feature present in some model. This gives a partial ordering of the equivalence classes and, consequently, the models. One can also use a threshold, ϵ , such that hypothesis verification is conducted only when $d(\mathbf{p}_1, \mathbf{p}_2) \leq \epsilon$. This reduces unwarranted verification when the parameters of the transformation between an indexing feature and an image are far from those given by the matches between that indexing feature and the subset of models to which it is related. One metric for P , which uses (6), is the absolute value of the angular difference between two points:

$$d(\mathbf{p}_1, \mathbf{p}_2) = \left| \cos^{-1} \left(\frac{\mathbf{p}_1^t \mathbf{p}_2}{\|\mathbf{p}_1\| \|\mathbf{p}_2\|} \right) \right| \quad (7)$$

Where $\mathbf{p}_1, \mathbf{p}_2 \in P$ and $\mathbf{p}_1^t \mathbf{p}_2$ is the Euclidean inner product. (7) gives the distance around the unit circle and is multi-valued which complicates the metric, but it is well known that:

$$\begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} \cos\pi & -\sin\pi \\ \sin\pi & \cos\pi \end{bmatrix} = \begin{bmatrix} -S_x & 0 \\ 0 & -S_y \end{bmatrix} \text{ and } \begin{bmatrix} -S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} \cos\pi & -\sin\pi \\ \sin\pi & \cos\pi \end{bmatrix} = \begin{bmatrix} S_x & 0 \\ 0 & -S_y \end{bmatrix} \quad (8)$$

Thus one need only consider the upper half of the plane specified by P to account for all phenomenally different anisotropically scaled instances. This restricts the range of values to $[0, \pi]$ thus avoiding the problem. Also, if an indexing feature is mirror symmetrical, as it is with the 'handle', then only the positive quadrant of P need be considered.



□ Parameter mapping of the class representative for each model equivalence class

Figure 2: The equivalence classes for the four models related to the indexing feature 'handle'. The parameters for the class representatives are (a) 'bracket' (1.25,8.5), (b) 'bottle' (2,2), (c) 'c-spanner' (3,2), and (d) 'screwdriver' (5,1.5).

Anisotropic scaling also deals with reflections. This is a desirable property in certain contexts e.g. the inspection of 2-D machine components which could lie on either side. Separate representations of the indexing feature are not required. Instead the indexing feature will have two indices into the same model, (S_x, S_y) and $(-S_x, S_y)$.

Finally, although the viewing transformation above deals solely with anisotropic scaling, the current work uses a 2-D viewing transformation specified by five parameters; two for scale (S_x, S_y) , one for rotation (ψ) , and two for translation (T_x, T_y) :

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \quad (9)$$

This presents no problem for the current implementation as the length of a line segment, and hence the scale of an object, is unaffected by rotation and translation.

5 Implementation Details and Performance

The current implementation uses a model-driven hypothesis-verification cycle. It attempts to determine a correspondence between an indexing feature and an image and, if one exists, the scale parameters of the transformation are compared with those of the models associated with the indexing feature as described above. (The current threshold used for (7) is set at 3°.) Each model indexed must then be verified by a full model match. If verified those image primitives forming the correspondence with the *model* primitives are then removed from the image description. This continues recursively until the image is too small to be matched further or until all of the indexing features have been considered.

Because the system is concerned with the parameter values of viewing transformations a natural choice of matching algorithm for both hypothesis generation and verification is the Generalised Hough Transform (GHT) [1, 2]. However, the GHT makes exponential demands upon memory with respect to the number of parameters of the viewing transformation. Two techniques are used to overcome this problem:

1. Parameter space decomposition - Because there is a natural dominance of parameters the 5-D parameter space for the viewing transformation can be decomposed into a linear hierarchy of three sub-spaces; a 2-D scale, 1-D rotation, and 2-D translation parameter space in that order [1, 2]. Possible scale parameters for the viewing transformation are determined and then passed on to determine the orientation parameter and so on. Because the constraining information of the primitive measurements are de-coupled a large number of false local maxima are generated in the scale parameter space. Many of these are eventually disconfirmed by using two simple constraints. The first constraint is to place a threshold on the metric as suggested above. The second constraint involves decreasing the number of false maxima in *subsequent* parameter spaces by reducing the image description. Only those image line segments consistent with at least one indexing feature (or model) line segment under the specific parameter values found in previous parameter spaces are used in determining the parameters in the subsequent parameter spaces.

2. Coarse-fine search within parameter spaces - Hierarchical coarse-fine search is also conducted within each parameter space; this is loosely based upon the Adaptive Hough Transform [8]. It involves using a coarsely quantised accumulator array, incrementing the array according to the normal GHT procedure, and clustering any cells whose count passes a threshold. The parameter values of the *bounds* of each cluster are then dynamically mapped onto the coarse accumulator array and the process continues recursively. The process terminates when the parameter values mapped onto the bounds of the accumulator array reach some pre-defined minimal resolution. The centroids of the clusters of cells found at minimal resolution are taken as possible solutions for the transformation defined by that parameter space.

The current number of intervals used for each parameter is 15. This results in 225 cells for the accumulator array of the scale and translation parameter spaces and only 15 cells for the rotation space, as compared to 15^5 for a single 5-D parameter space.

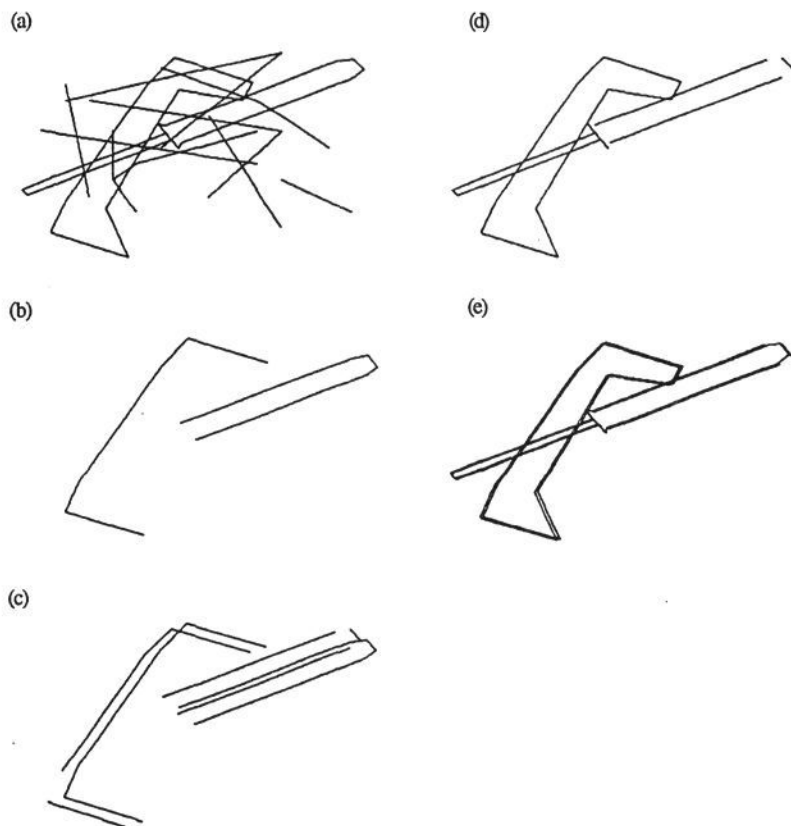


Figure 3: Matching an image containing two different object types.

Hypothesis Generation: (a) The image. (b) The indexing feature hypotheses. (c) The hypothesised instances superimposed upon the reduced image description. Hypothesis Verification: (d) The reduced image description after both hypothesised model matches. (e) The hypothesised models superimposed on the reduced image description of (d).

A typical example of the system at work is given in Figure 3. The 'screwdriver' and 'bracket' were transformed according to the parameters given in Table 1. Noise segments were added giving a total number of 35 line segments, resulting in a signal/noise ratio of 0.1666 for each instance of the indexing feature. (The noise was generated by finding the minimal enclosing rectangle about the objects of interest, and using a uniformly distributed random number generator to set the endpoints of a segment to fall within it).

Table 1: The parameter values of the model transformations used in generating the image.

Model	Scale	Ψ	T_x	T_y
Screwdriver	1.1	30°	20	10
Bracket	1.9	155°	18	9

Table 2 gives the number of hypotheses generated by matching the 'handle' with the image for each parameter space in the hierarchy of sub-spaces. Note the power of the threshold constraint in reducing the number of scale hypotheses and the general reduction in hypotheses as one traverses the hierarchy due to both the constraining nature of the successive transformations and the use of image reduction.

Table 2: The results of matching the indexing feature with the image.

No. Scale Hypotheses	Reduced No. Scale Hypotheses	Rotation Hypotheses	Translation Hypotheses
37	15	3	2

The two translation hypotheses represent the number of correspondences between the image and the 'handle' once scale, orientation, and translation are taken into account. The parameter values of these hypotheses are given in Table 3.

Table 3: The viewing transformation parameter values of the two hypotheses

	S_x	S_y	Ψ	T_x	T_y
First Hypothesis	5.50131	1.63356	30.0267°	20.6667	9.33320
Second Hypothesis	2.38418	16.17490	155.04°	18.6667	9.33332

The first hypothesis indexes the 'screwdriver' whilst the second indexes the 'bracket'. These are the required hypotheses; no others have been made. The distance between the scale parameters and the parameters of the appropriate equivalence classes are 0.16° and 0.13° respectively and fall well within the threshold. Each hypothesis is verified by a match with the appropriate model and the parameters are given in Table 4 (c.f Table 1).

Table 4: The viewing transformation parameter values of the two verified hypotheses

	S_x	S_y	Ψ	T_x	T_y
Screwdriver Hypothesis Verification	1.09927	1.10163	29.9982°	19.9316	10.05130
Bracket Hypothesis Verification	1.90118	1.90018	154.997°	18.1333	9.06666

Several hundred experiments using the database of Figure 1 have been conducted using images containing from one to three objects with varying degrees of noise. The pattern of the results given above is typical of the system's performance. It fails to recognise an object approximately 8% of the time but most of these failures can be avoided by modifications to the algorithms. Furthermore, when the system generates more hypotheses than instances actually present this is often due to slightly different scale parameters. These are still consistent under rotation and translation due to error introduced by quantisation of the accumulator arrays; they effectively represent the same index and this is easily resolved by verification of any one of the similar indexing hypotheses.

6 Conclusions

A new approach to indexing has been presented; indexing by the parameter values of deformable features. This unifies the advantages of both critical and similar indexing features. Like those schemes it is model-driven and thus necessitates exhaustive search at the level of the indexing features. However, unlike those schemes, it does not necessarily entail exhaustive search of those models related to a particular indexing feature. As with similar features a number of different objects can share the same indexing feature; different models can be associated with one another if a deformed instance of an indexing feature is present within them. Furthermore, like critical features, the system gives highly specific indices for each model related to an indexing feature via the parameter values of the deformational component of the viewing transformation.

Finally, further work on the current system includes a larger database, a more detailed analysis of the effect of noise on the system's performance, and an investigation of the effect of error on the image primitives themselves. The latter is of special interest as currently the images are generated by transforming the models and adding noise. Related to this last point is that systems requiring the accurate segmentation of line segments, such as this paper and e.g. [1, 2], may prove to be too 'brittle'. Therefore, some theoretical work has been conducted using collections of vertices for indexing features [9]. The segmentation of vertices may prove to be more reliable than that of line segments. [9] shows that one can recover the ratio of the anisotropic scale factors from the angles of vertices thus satisfying (5) and (6) above. This will be the subject of a future study.

References

- [1] D.H. Ballard, and D. Sabbah, On Shapes. In Proc. 7th IJCAI, Morgan Kaufmann, Los Altos, 1981, pp 607-612.
- [2] D.H. Ballard, and D. Sabbah, Viewer Independent Shape Recognition. PAMI 1983; 5(6): 653-660.
- [3] R.C. Bolles, and R.A. Cain, Recognizing and locating partially visible objects: The local feature focus method. Int. J. of Robotics Research 1982; 1(3): 57-82.
- [4] R.C. Bolles, P. Horaud, and M.J. Hannah, 3DPO: A Three-Dimensional Part Orientation System'. In Fischler M.A., and Firschein O, (Eds.) Readings in

Computer Vision: Issues, Problems, Principles, and Paradigms. Morgan Kaufmann, Los Altos , 1987, pp 355-359.

- [5] F.P. Sykes, S.B. Pollard, and J.E.W. Mayhew, Hypothesis and Verification in 3-D Model Matching. In Proc. 5th Alvey Vision Conference, University of Sheffield, 1989, pp 7-12.
- [6] T.F. Knoll, and R.C. Jain, Recognizing Partially Visible Objects Using Feature Indexed Hypotheses. IEEE J. of Robotics and Automation 1986, RA-2(1): 3-13.
- [7] T.F. Knoll, and R.C. Jain, Learning to recognize objects using feature indexed hypotheses. In Proc. First Int. Conf. on Computer Vision, 1987, pp. 552-556.
- [8] J. Illingworth, and J. Kittler, The Adaptive Hough Transform. PAMI 1987; 9(5): 690-698.
- [9] C.C. Hand, Model Indexing Using Parametrised Features. Phd Thesis (submitted for examination), University of Sussex, 1992.