



PERGAMON

Pattern Recognition 35 (2002) 1895–1915

**PATTERN
RECOGNITION**

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Relational object recognition from large structural libraries

Benoit Huet, Edwin R. Hancock*

Department of Computer Science, University of York, York, Y010 5DD, UK

Received 18 August 2000; accepted 2 August 2001

Abstract

This paper presents a probabilistic similarity measure for object recognition from large libraries of line-patterns. We commence from a structural pattern representation which uses a nearest neighbour graph to establish the adjacency of line-segments. Associated with each pair of line-segments connected in this way is a vector of Euclidean invariant relative angle and distance ratio attributes. The relational similarity measure uses robust error kernels to compare sets of pairwise attributes on the edges of a nearest neighbour graph. We use the relational similarity measure in a series of recognition experiments which involve a library of over 2500 line-patterns. A sensitivity study reveals that the method is capable of delivering a recognition accuracy of 94%. A comparative study reveals that the method is most effective when either a Gaussian kernel or Huber's robust kernel is used to weight the attribute relations. Moreover, the method consistently outperforms the standard and the quantile Hausdorff distance. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Image retrieval; Relational graphs; Hausdorff distance; Robust statistics

1. Introduction

Object recognition from large libraries of images holds the key to the automatic manipulation of massive volumes of visual information. The overall goal is the rapid recognition of images according to their contents. Most of the research literature on object recognition from large image libraries has focused on the use of low-level image summaries. One of the most popular object recognition schemes is to use the attribute histogram. This idea was originally popularised by Swain and Ballard [1] for retrieving colour images from data-bases. The idea underpinning this representation is to bin suitably chosen image attributes in a fairly coarse histogram and to effect recognition on the basis of minimum histogram

distance. The retrieval method has been extended to both texture [2] and orientation representations [3,4]. Histograms of topographic shape index information have also been used for recognising objects from range images [5]. For line pattern recognition, a closely related representation is the pairwise geometric histogram [6]. Rather than using raw image attributes, this representation uses relative attributes defined over pairs of lines segmented from edge-maps.

Despite this interest in developing compact object representations for rapid image recognition and retrieval, there has been little work aimed at using higher level relational descriptions for large-scale object recognition problems. For instance, in the domain of CAD-based vision Costa and Shapiro [7] and Sengupta and Boyer [8] have used relational matching techniques to recognise arrangements of lines. However, these two studies focus on relatively simple and noise free imagery. The aim in this paper is to focus in more detail on how large line patterns can be recognised when there are significant levels of noise and segmentation error.

*Corresponding author. Tel.: +44-1904-43-3374; fax: +44-1904-43-2767.

E-mail addresses: huetb@cs.york.ac.uk (B. Huet), erh@cs.york.ac.uk (E.R. Hancock).

1.1. Literature review

In this paper, we will be concerned with recognising line-patterns. In this section we review the available methods reported in the literature. The review is structured according to the representational complexity of the recognition strategy.

One of the simplest attribute representations for line-patterns is the pairwise geometric histogram [6]. Rather than using raw absolute image attributes, this representation uses relative attributes defined over line-pairs. Several alternative attribute sets have been suggested, but most revolve around the use of angle difference and relative perpendicular distance [9]. Each of these different histogram-based recognition techniques can be viewed as accumulator methods that avoid the problems of storage and search which limit the use of techniques such as the generalised Hough-transform [10]. Recently, we have reported a more sophisticated histogram-based recognition strategy which uses an N -nearest neighbour graph to gate the contributions to a pairwise geometric histogram [11]. The use of relational structure is demonstrated to offer significant performance improvements when noisy line-patterns are being matched.

Although histograms provide succinct attribute summaries, they can be criticised on the grounds that they collapse too much detail. An alternative is to treat the matching problem as that of comparing sets of feature attributes. This is the approach adopted by Rucklidge [12], who has shown how the Hausdorff distance can be used for relatively robust object recognition and location. The Hausdorff distance is a compelling way of comparing sets of object features since it possesses several important mathematical properties. Most important of these is that it satisfies the metric axioms in its undirected form. From a conceptual standpoint, the metric can be viewed as a saliency measure rather than a similarity measure since it selects the correspondences between members of two sets which maximise distance. Rucklidge et al. use the Hausdorff distance for low-level pixel-based recognition, rather than for relational object recognition. The more recent study of Yi and Camps [13] has shown how line patterns can be aligned using a four dimensional Hausdorff distance to recover translation, scale and rotation. Recently, Santini and Jain [14] have addressed the issue of how similarity should be measured for content-based image retrieval. Their starting point is the observation that the distance axioms are over-restrictive. Instead of using saliency to compare sets, they turn to the psychology literature and Tversky's similarity measure [15] as a means of comparison. Underlying Tversky's similarity measure is the idea of feature contrast. This is posed in terms of crisp set theory. Santini and Jain [14] generalise Tversky's ideas to fuzzy-sets and exploit the resulting similarity measure for face and texture recognition.

The methods described above are effectively ones of template matching. A more complex approach to object recognition is to pose the process in terms of matching attributed relational graphs. When abstracted in this way there are two key computational ingredients that must be supplied. The first of these is an attribute representation that is robust to noise and occlusion, and which is invariant to changes in object viewing geometry. The second ingredient is a means of comparing relational descriptions. Ideally, the distance or similarity measure should have a degree of robustness to the effects of line-clutter, line-dropout and poor attribute measurements. The first of these issues, i.e. of efficient object representation, has recently stimulated considerable interest in the literature [16]. Examples include both geometric [17] and structural hashing [18,7], a variety of invariants [19,20] and pairwise geometric histograms [6]. The second issue of how to compare representations is a long standing problem in the structural pattern recognition literature. For instance, Shapiro and Haralick [21,22] have shown how inexact relational descriptions can be compared and matched by counting consistent arrangements of edge-structure. Sanfeliu and Fu [23] have a finer measure of relational consistency, which uses edit distance to compare relational graphs. Wong and You [24] have developed a more information theoretic approach which uses entropy to measure the similarity of graphs. More recently, Wilson and Hancock [25] have reported a Bayesian framework which can be used for both relational graph-matching and correcting structural errors by graph-editing [26].

Recently, there have been several attempts to use graph-based representations specifically for 2D shape recognition. For instance structural indexing has been demonstrated as an effective tool for line-pattern recognition in the domain of CAD-based vision [7,8]. Specifically, it has been used to successfully recall object-models from large data-bases of engineering drawings. The basic idea is to use the relational structure of object primitives to construct an index that can be used to discriminate object shape. Since efficiency is invariably a key issue when dealing with large image data-bases, conventional structural matching methods are too computationally demanding to be used as search-engines. This rules out many promising computational techniques such as inexact graph matching [25], structural hashing [18] and relaxation labelling [27,28]. Instead, the goal is to construct easily computed structural summaries that can be used to make a rapid comparison on the basis of salient object structure. Concrete examples include the relational indexing scheme of Costa and Shapiro [7]. In this approach, each model view is represented by sub-graphs of fixed size derived from the full relational graph. Each sub-graph is then encoded into a hash-table for subsequent use in indexing. The major drawbacks of this graph encoding scheme are the computational overheads which grow rapidly with

both the number of feature types and the cardinality of the structural relations.

For 2D patterns, the FORMS system of Zhu and Yuille [29] uses skeletal shape graphs to model articulated objects. Liu and Geiger [30] have taken these ideas further by developing a hierarchical model which achieves a degree of unification between the articulated shape graph and the detection of raw image features via the Mumford–Shah functional [31]. Amit and Kong [32] have a MAP framework for modelling 2D deformable shape using a decomposable graph representation. More recently, there have been several attempts to use graph retrieval as a means of recognising 2D shapes from data-bases. Much of this work can be viewed as providing a concrete realisation of the ideas originally introduced by Leyton's [33] process grammar for shape. For instance Siddiqi et al. [34] have used the shock graph derived from the singularities of the reaction–diffusion equation to provide a skeletal representation of 2D binary shapes. Shape recognition is realised using the subtree matching algorithm of Reyner [35]. The matching process has been refined by Pelillo et al. [36] who establish a means of matching association trees using a relaxation algorithm inspired by evolutionary game theory to find the maximal clique of the association graph. There has also been interest in using blobs and regions for the purposes of retrieval [37,38].

1.2. Motivation

The aim in this paper is to address the issue of the relational representation and recognition of line patterns in a more critical manner. The review above suggests that abstracting the patterns in terms of sets of pairwise attributes may offer a useful compromise between full-scale graph-matching and the use of histograms of rather restrictive representational power. However, when abstracted in this way the question of how to compare the sets of attributes remains. Although the Hausdorff distance is an obvious candidate, there are a number of criticisms that can be levelled at its use. In the first instance, the measure is crisply defined over the max–min tests between the elements of the sets of object-primitives being compared. Although this offers a certain degree of robustness to noise and outlier contamination, it fails to adequately capture uncertainties in the image attributes being compared. The second shortcoming, is the failure to impose relational structure on the arrangements of object-primitives. In other words, a considerable wealth of contextual information is overlooked. Moreover, since the object primitives under study are subject to both measurement uncertainty and segmentation error, fuzzy or probabilistic distance measures may be more appropriate to the comparison task than crisply defined set-based methods such as the Hausdorff distance. In particular, recent interest in graph-matching has furnished a prob-

abilistic methodology for exploiting inexact relational constraints in the matching of attributed relational graphs [25,28]. Unfortunately, none of these graph-matching methods are particularly well suited for large-scale object recognition. The reason for this is that they rely upon a consistent arrangement of node and edge correspondences being available. One way of overcoming the need for explicit correspondences is to treat the available attributes as a set of unordered features and to pose the problem of recognition as set comparison. However, this idea has not been exploited in large-scale object recognition tasks.

We aim to fill this gap in the literature by developing a graph-based method for recognising the relational arrangement of line-segments. We pose the problem as one of recognising line patterns by comparing the sets of pairwise geometric attributes on the edges of a neighbourhood graph. The feature-set comparison is effected by maximising a probabilistic relational similarity measure. We must stress that the method requires exemplar patterns for the purposes of retrieval. It is not particularly well suited to the iterative crafting of queries.

The outline of this paper is as follows. We commence Section 2 by providing details of our object representation. This describes the pairwise geometric attributes used in our study together with the relational abstraction of line-patterns. In Section 3 we provide the development of a relational similarity measure which uses robust error kernels to compare the sets of pairwise geometric attributes. In Section 4 we describe some recognition experiments which are used to establish the best choice of neighbourhood graph and robust error kernel. Issues of algorithm sensitivity to noise and segmentation error are the subject of Section 5. Finally, Section 6 summarises our conclusions and offers some directions for future investigation.

2. Object representation

In this section, we review our object representation. There are two aspects to this representation. The first of these is the attribute or measurement content. Here, we describe a set of Euclidean invariant pairwise geometric attributes based on relative angles and relative positions. These attributes have been successfully exploited in histogram-based object retrieval [39]. The second aspect of the representation is structural. Here, we use a nearest neighbour graph to impose relational structure on the lines. The reason for adopting this graph-structure is a recent sensitivity study that reveals that it offers the best compromise between robustness to structural errors and computational overheads [40]. It is the edges of this graph-structure which provide simple relational constraints on the recognition process.

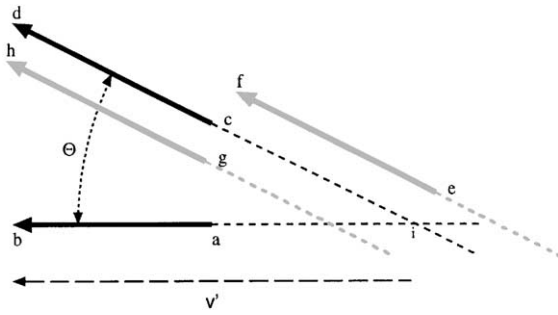


Fig. 1. Geometry for shape representation.

2.1. Pairwise geometric attributes

There are many possible choices of geometric attributes for line-pattern recognition. For instance, Bray [9] has listed five possibilities. They are the relative angle, the length-ratio, the ratio of end-point distances, the intersection length ratio and the line-segment projection cross ratio. Of these it is the relative angle attribute that is most commonly used in the literature [41,28,42] because of its invariance to translation, scaling and rotation. Moreover, it is also robust to line fragmentation and to end-point erosion. However, as we shall demonstrate later, the definition of the relative angle attribute can be improved to reduce recognition ambiguities. It is the choice of an invariant length-based attribute that has posed the greatest difficulty. For example, Thacker et al. [41] use the perpendicular distances between segment pairs. Since the attribute is not scale-invariant, multiple attribute histograms must be computed and stored for the purposes of recognition. In their super-segment based representation, Stein and Medioni [42] use information about the number of sub-segments, the sum of the line-segment length, its location and its overall orientation. Unfortunately, none of these distance attributes meet our requirement of scale-invariance.

In a recent study, we have proposed two Euclidean invariant attributes for line-pattern recognition and have demonstrated their effectiveness for line-pattern recognition using pairwise geometric histograms [11]. Here, we review the computation of these attributes. The raw information available for each line segment includes its orientation (angle with respect to the horizontal axis), the position of its centre and its length (see Fig. 1). To illustrate how the pairwise feature attributes are computed suppose that we denote the line segments indexed (*ab*) and (*cd*) by the vectors \underline{x}_{ab} and \underline{x}_{cd} , respectively. The vectors are directed away from their point of intersection. The relative angle attribute is given by

$$\theta_{\underline{x}_{ab}, \underline{x}_{cd}} = \arccos \left[\frac{\underline{x}_{ab} \cdot \underline{x}_{cd}}{|\underline{x}_{ab}| |\underline{x}_{cd}|} \right].$$

From the relative angle we compute the directed relative angle. We give the relative angle a positive sign if the direction of the angle from the baseline \underline{x}_{ab} to its partner \underline{x}_{cd} is clockwise and a negative sign if the direction is counter-clockwise. This allows us to extend the range of angles describing pairs of line segments from $[0, \pi]$ to $[-\pi, \pi]$ and therefore, reduce retrieval errors associated with angular ambiguities.

The second attribute is a measure of the relative positions of line-segment pairs. This attribute is computed from the lengths of two vectors. The first of these is the vector \underline{x}_{ab} connecting the two ends of the line segment (*ab*). The second vector $\underline{x}_{i,b}$ connects the intersection point *i* of the line-segments (*ab*) and (*cd*) to the furthest end *b* of the line-segment (*ab*). The relative position attribute is computed from the length ratio of these two vectors and is defined to be

$$\vartheta_{\underline{x}_{ab}, \underline{x}_{cd}} = \frac{1}{\frac{1}{2} + |\underline{x}_{ib}|/|\underline{x}_{ab}|}.$$

The physical meaning of this attribute deserves further comment. The minimum value of $|\underline{x}_{ib}|$ occurs when the line *cd* intersects the line *ab* at its midpoint. As a result $|\underline{x}_{ib}| \geq \frac{1}{2} |\underline{x}_{ab}|$ and hence $|\underline{x}_{ib}|/|\underline{x}_{ab}| \geq \frac{1}{2}$. As a result the minimum value of the denominator is 1 and hence the maximum value of the directed relative position is unity. The physical range of the attribute $\vartheta_{\underline{x}_{ab}, \underline{x}_{cd}}$ is therefore $(0, 1]$. A relative position of 0 indicates that the two segments are parallel, while a relative position of 1 indicates that the two segments intersect at the middle point of the baseline. It is worth pointing out that both the directed relative angle and the directed relative position attributes are invariant to changes of scale, rotation and translation. This is an important advantage over the representation proposed by Evans et al. [6] if the technique is to be used for shape retrieval from real world image databases where the size of the target objects is not known in advance.

2.2. Relational constraints

We aim to augment the pairwise geometric attributes with information concerning the relational arrangement of the line-segments. Here, we use the adjacency of the centre-points of the line-segments to establish a neighbourhood structure. The adjacency structure used in our experiments is the *N*-nearest neighbour graph. The recognition strategy described in this paper is based on measuring the similarity of sets of pairwise geometric attributes for line-patterns. As we will demonstrate later, by confining our attention to pairs of line-segments connected by edges of an *N*-nearest neighbourhood graph, we can achieve performance comparable to that of a full-scale graph matching algorithm.

Suppose that the line-segments extracted from the raw image is denoted by the index-set *V*. The set *V*

is the node-set for the nearest neighbour graph for the centre-points of the line segments. The edge set of the nearest neighbour graph $E \subset V \times V$ is constructed as follows. The nodes of our graphs are the centre-points of the line segments extracted from the raw image data. For each node in turn, we create an edge to the N line-segments that have the closest centre-point distances. Associated with the edges of the N -nearest neighbour graph are pairwise geometric attributes. The set of pairwise geometric attributes for the line-segment pairs connected by edges of the N -nearest neighbour graph is denoted by $B = \{(\theta_{i,j}, \vartheta_{i,j}; (i,j) \in E \subseteq V \times V)\}$. With these ingredients in hand, we represent the sets of line-patterns as triples of the form $G = (V, E, B)$. The process of abstracting the line-patterns in this way is illustrated in Fig. 3 for an aerial infra-red image.

We are interested in large data-bases of line-patterns represented in this way. The alternative line-patterns in the data-base are denoted by $G_d = (V_d, E_d, B_d)$, $\forall d \in \mathcal{D}$ where \mathcal{D} is the index-set of the data-base. The task that confronts us is to locate the line-patterns in this data-base which most closely resemble the query pattern $G_m = (V_m, E_m, B_m)$.

2.3. Relational structure

Although it is the N -nearest neighbour graph that is the principal focus of attention in this paper, there are alternative adjacency structures available in the literature. For instance, Tuceryan and Chorzempa [43] have studied the noise sensitivity of three alternatives. These are the Delaunay graph, the Gabriel graph and the relative neighbourhood graph. From an implementation standpoint the simplest structure is the N -nearest neighbour graph. This is computed by connecting the N nearest points by graph edges. The three alternative graphs are all derived from Voronoi polygons seeded from point-sets in a region growing process. The Delaunay graph is constructed by connecting the seed-points if their Voronoi cells share region adjacency. The Gabriel graph is computed by constructing circles with Delaunay edges as diameters. The edge-set of the Gabriel graph is obtained by pruning from the Delaunay graph all edges whose circumscribing circle encloses one or more nodes. The relative neighbourhood graph is obtained by a further pruning of the Delaunay edge-set. In this case a lune is constructed on each Delaunay edge. The circles enclosing the lune have their centres at the end-points of the Delaunay edge; each circle has a radius equal to the length of the edge. If the lune contains another node then its defining edge is pruned from the relative neighbourhood graph. We note that this sequence of pruning operations has the effect of reducing the edge density of the different graph structures. Whereas the Delaunay graph consists entirely of triangulated faces, the relative

neighbourhood graph is more tree-like in its structure. It is worth commenting at this point that for point-sets of size $|V|$ algorithms of complexity $O(|V| \log |V|)$ exist for computing the Delaunay graph. By contrast the complexity of computing the N -nearest neighbour graph is $|V|^2 \log |V|$.

In a recent study of graph-matching Wilson et al. [26] experimented with these different relational structures. Their aim was to determine which structure allowed the best correspondence matches to be recovered. They compared the N -nearest neighbour graph with the Delaunay graph, the Gabriel graph and the relative neighbourhood graph. The best performance is achieved with the Delaunay graph. The N -nearest neighbour graph came a close second. In fact, an N -nearest neighbour graph of order 5 or 6 offered almost comparable performance to the Delaunay graph. It is interesting to note that this is approximately the average connectivity as the Delaunay graph. The Gabriel graph and the relative neighbourhood graphs deliver performance that drops off rapidly with increasing noise. These are observations of critical importance for our study of line-pattern recognition. The computation of the Delaunay graph and its derivatives, i.e. the Gabriel graph and the relative neighbourhood graphs, is involved and therefore not particularly well suited for large image databases. The N -nearest neighbour graph, on the other hand can be computed in a very straightforward manner.

3. Pairwise attribute consistency

The aim in this paper is to develop a similarity measure for rapidly comparing line-patterns in a large data-base of alternatives. The patterns are represented as sets of pairwise geometric attributes on the edges of a neighbourhood graph. Abstractions of this sort can be recognised using one of the large number of graph-matching algorithms described in the literature [25]. However we argue that graph-matching is inappropriate for three reasons. First, graph-matching is frequently posed as an iterative process and is hence time consuming. Second, it is a fragile process which is easily confounded by structural error. Third, it is concerned with detailed correspondence analysis rather than global object recognition. In contrast, the aim here is to provide a simplified relational similarity measure which can be used for recognition without the need to iteratively establish correspondence matches. We abstract the line-patterns as sets of pairwise attributes residing on the edges of an N -nearest neighbour graph for the centre-points of the line-segments. Recognition is achieved by measuring the similarity of the sets of attributes. Rather than using crisply defined set-based distance measures, we gauge the similarity of the attributes using a probabilistic measure.

3.1. Global pattern similarity

Our measure of relational similarity is the average probability of correspondence between the edges of different neighbourhood graphs. Measures such as this have been used to gauge consistency in the relaxation literature [44,28]. To be more formal, suppose that the set of nodes connected to the model-graph node I by an edge of the neighbourhood graph is $C_I^m = \{J \mid (I, J) \in E_M\}$. The corresponding set of data-graph nodes connected to the node i by an edge of the neighbourhood graph is $C_i^d = \{j \mid (i, j) \in E_d\}$. With these ingredients, the relational consistency criterion which combines evidence for the match of the graph G_m onto G_d is

$$Q(G_d, G_m) = \frac{1}{|V_M| \times |V_d|} \sum_{i \in V_d} \sum_{I \in V_m} \frac{1}{|C_i^d|} \sum_{J \in C_I^m} \frac{1}{|C_I^m|} \times \sum_{j \in C_i^d} P((i, j) \rightarrow (I, J) | \underline{v}_{I,J}^m, \underline{v}_{i,j}^d). \quad (1)$$

The probabilistic ingredients of the relational similarity measure need further explanation. The a posteriori probability $P((i, j) \rightarrow (I, J) | \underline{v}_{I,J}^m, \underline{v}_{i,j}^d)$ represents the evidence for the match of the model-graph edge (I, J) onto the data-graph edge (i, j) provided by the corresponding pair of attribute relations $\underline{v}_{I,J}^m$ and $\underline{v}_{i,j}^d$. The meaning of the similarity measure is as follows. The first two summations run over the set of possible correspondences between model-graph and data-graph nodes. The inner two summations are responsible for comparing the attribute structure on the individual edges of the neighbourhoods of the corresponding nodes.

We now consider how to simplify the computation of relational similarity. We commence by considering the inner sum over the nodes in the model-graph neighbourhood C_I^m . Rather than averaging the edge-compatibilities over the entire set of feasible edge-wise associations, we limit the sum to the contribution of maximum probability. Similarly, we limit the sum over the node-wise associations in the model graph by considering only the matched neighbourhood of maximum probability. In other words, we restrict the sums over the potential node and neighbourhood edge correspondences in the model-graph to the most likely items. We average the probabilities of these most likely correspondences over the nodes and their neighbourhoods in the data-graph. With these restrictions, the process of maximising the relational similarity measure is equivalent of maximising the quantity

$$Q(G_d, G_m) = \frac{1}{|V_M| \times |V_d|} \sum_{i \in V_d} \max_{I \in V_m} \sum_{j \in C_i^d} \max_{J \in C_I^m} P((i, j) \rightarrow (I, J) | \underline{v}_{I,J}^m, \underline{v}_{i,j}^d). \quad (2)$$

Later on we will provide some comparison with the use of Hausdorff distance for the purposes of recognition.

It must be stressed that although our relational similarity measure shares with the Hausdorff distance the feature of using extremum tests (max's and min's), it does not satisfy the distance axioms. Moreover, while the Hausdorff distance is saliency-based, i.e. it measures the maximum distance between two sets of observations, our measure returns maximum similarity.

With the similarity measure at-hand, the class identity ω_m of the best matching line pattern in the data-base is the one which satisfies the condition

$$\omega_m = \arg \max_{d' \in \mathcal{D}} Q(G_{d'}, G_m). \quad (3)$$

At this point it is worth commenting on the complexity of the retrieval scheme. When we use an N -nearest neighbour graph then $|C_i^d| = |C_I^m| = N$. Hence, $N^2 |V_m| \times |V_d|$ similarity comparisons must be made. For each of the $|E_d|$ edges of the data-graph we must also calculate $|E_m|$ matching probabilities.

In Section 6 we will experiment with this retrieval scheme. Here, we will focus on two types of query. The first of these are exact queries. Here the query and target line patterns are related to one another through a similarity transform, i.e. through translation, rotation and isotropic scaling. The second type of query is inexact. Here the target pattern has undergone distortion with respect to the query. We investigate various types of line-pattern distortion. The simplest of these involves random perturbation in the length and angle attributes due to measurement error. We also investigate structural corruption effects which may change the size of the line-patterns. These may be the result of noise or poor image segmentation. They include line insertion or deletion and line splitting. Since our similarity measure averages the probabilities over the two pattern sets being matched its computation is not sensitive to differences in pattern size due to structural errors.

3.2. Robust weighting kernels

To develop a concrete recognition algorithm, we require a model of the a posteriori edge-matching probabilities. Here, we use robust weighting functions to compute the desired probabilities. Accordingly, we write

$$P((i, j) \rightarrow (I, J) | \underline{v}_{I,J}^m, \underline{v}_{i,j}^d) = \frac{\Gamma_\sigma(\|\underline{v}_{I,J}^m - \underline{v}_{i,j}^d\|)}{\sum_{(I,J) \in E_m} \Gamma_\sigma(\|\underline{v}_{I,J}^m - \underline{v}_{i,j}^d\|)}, \quad (4)$$

where $\Gamma_\sigma(\cdot)$ is a distance weighting function.

We will consider several alternative robust weighting functions. The most appealing of these is a Gaussian of the form

$$\Gamma_\sigma(\rho) = \exp\left(-\frac{\rho^2}{2\sigma^2}\right).$$

We will also consider several alternatives suggested by the robust statistics literature. These include

- the sigmoidal derivative

$$\Gamma_\sigma(\rho) = \rho^{-1} \tanh\left(\frac{\pi\rho}{\sigma}\right).$$

- Huber’s kernel

$$\Gamma_\sigma(\rho) = \begin{cases} 1 & \text{if } \rho < \sigma, \\ \frac{\sigma}{|\rho|} & \text{otherwise.} \end{cases}$$

- Huber’s narrow-band kernel

$$\Gamma_\sigma(\rho) = \left(1 + \frac{|\rho|}{\sigma}\right)^{-1}.$$

With edge-correspondence probabilities defined according to one of these robust weighting functions, the relational similarity measure can be viewed as an average pairwise attribute consistency measure. Several authors have reported the use of such measures for graph-matching. Christmas et al. [28] have used a similar measure as their starting point in the development of an iterative probabilistic relaxation operator. Wilson and Hancock [25], on the other hand, have a purely symbolic method for enumerating the average consistency of match in discrete relaxation. However, rather than being used for primitive-by-primitive correspondence matching, in the work reported here we use the criterion for recognising primitive ensembles.

3.3. Hausdorff distance

In our experimental evaluation of the new recognition measure, we will provide some comparison with the Hausdorff distance used by Rucklidge [45,12]. It must be stressed that while our method is concerned with finding graphs of maximum similarity, the Hausdorff distance is concerned with finding patterns of maximum saliency. However, in order to make the comparison meaningful, in this section we describe how the Hausdorff distance can be extended to graph-based object representations.

The idea underpinning the conventional Hausdorff distance is to compute the distance between two sets of unordered observations when the correspondences between the individual items are unknown. In object recognition, this problem presents itself when sets of unlabelled image primitives are being compared. In other words, it provides a means of avoiding the computationally demanding problem of attempting to find a consistent arrangement of correspondence matches between individual primitives whilst performing recognition.

The directed Hausdorff distance gauges the distance between the two sets of observations using the maximum value of the minimum pairwise distance. In other

words, it locates the most salient set of correspondences. More formally, with the graph-based notation introduced in Section 2.2, the distance between the sets of pairwise attributes on the data-graph and the model-graph is defined to be

$$H_B(G_d, G_m) = \max_{(i,j) \in V_d \times V_d} \min_{(l,J) \in V_m \times V_m} \|\underline{v}_{(l,J)}^m - \underline{v}_{(i,j)}^d\|.$$

There are $|V_m|^2 \times |V_d|^2$ comparisons involved in the computation of the distance. Moreover, if recognition is being attempted then the large number of possible pairwise correspondences are likely to render the object representation highly ambiguous.

One way of overcoming these problems and of simultaneously improving the quality of recognition is to confine our attention to those pairwise measurement relations that are defined on the edges of the graphs representing the adjacency structure of the object primitives. In Eq. (3), we presented a simplified relational similarity measure which uses the edge structure on the local neighbourhoods of the nodes to assess the similarity of attributed graphs. We use the relational similarity measure to suggest a more computationally efficient Hausdorff distance. We argue by analogy. As noted above the Hausdorff distance is a saliency measure rather than a similarity measure. For this reason we replace the summations over the data-graph nodes appearing in Eq. (3) by max and min operators. The distance measure therefore selects the most salient neighbourhood matches rather than the most similar neighbourhood matches. The resulting directed Hausdorff distance is

$$H_G(G_d, G_m) = \max_{i \in V_d} \max_{j \in C_i^d} \min_{l \in V_m} \min_{J \in C_l^m} \|\underline{v}_{(l,J)}^m - \underline{v}_{(i,j)}^d\|.$$

This can be thought of as a hierarchical-version of the Hausdorff distance. It first selects the most salient neighbourhood matches. Once the neighbourhood matches are at hand, then the most salient node matches are selected.

3.4. Improving the robustness of the Hausdorff distance

The Hausdorff distance is defined over two unordered sets of measurements using crisply defined max and min tests. The max test selects on the basis of saliency while the min test selects on the basis of closeness. As a result the method may become ineffective if there is either lack of saliency due to observational overlap or ambiguity of appearance, or if there is lack of closeness due to excessive noise or outlier contamination. In this section, we consider ways of rendering the Hausdorff distance more robust by using alternatives to the two crisp tests.

3.4.1. fth quantile Hausdorff distance

To overcome the first of these problems, Rucklidge [12] has reported a modification of the standard

Hausdorff distance which produces important performance improvements. His idea is to replace the max comparator with an f th quantile test on the ordering of the measurement distances. It is a simple matter to develop an f th quantile version of the graph-based Hausdorff distance described above. The directed distance is

$$H_G^f(G_d, G_m) = F_{i \in V_d}^f \left\{ \min_{I \in V_m} F_{j \in C_i^d}^f \left\{ \min_{J \in C_i^m} \left\| \underline{v}_{(I,J)}^m - \underline{v}_{(i,j)}^d \right\| \right\} \right\},$$

where the operator $F_{i \in V_d}^f$ selects the f th quantile value from the set of attributes for the node $i \in V_d$.

3.4.2. Controlling outliers

Whereas, Rucklidge's quantile operator represents a robust procedure for ordering pairwise attributes according to their saliency, it does not provide a means of controlling measurement outliers. The second problem outlined above, i.e. that of controlling measurement outliers, can be overcome by making soft associations between the members of the two sets of measurements. Rather than taking the inner min operation, we would like to weight associations according to their closeness. Unfortunately, if this is attempted using a standard distance norm, then when the outer max operation is performed, there will be a tendency to loose saliency. In other words, we need to choose a weighting function which saturates for large attribute distances rather than growing monotonically. The robust statistics literature furnishes several weighting functions which meet this requirement. Here, rather than performing a quantile test we sum the attribute weights. As a final primitive-based distance-measure, we have therefore considered applying the Hausdorff tests to the summed complement of the weighting function. The resulting set-dissimilarity measure is defined to be

$$H^P(G_d, G_m) = \sum_{i \in V_d} \min_{I \in V_m} \sum_{j \in C_i^d} \min_{J \in C_i^m} \times (1 - \Gamma_\sigma(\|\underline{v}_{(I,J)}^m - \underline{v}_{(i,j)}^d\|)). \quad (5)$$

This equation is similar to Eq. (3). This equation may be viewed as an example of fuzzy composite operation that is similar to those described in the review of Bloch [46] on fuzzy operator for computer vision and image processing. This particular kind of fuzzy operator is referred to as a context independent constant behaviour operator.

4. Experiments

In this section, we provide some experimental evaluation of the new matching method. We commence by exploring the algorithm characteristics. Here, we

investigate the best choice of distance measure and graph-structure. We also compare with several variants of the Hausdorff distance. Secondly, we provide some qualitative recognition examples.

4.1. Data

We have conducted our recognition experiments with a data-base of over 2500 line-patterns each containing over a hundred line-segments on average. The data-base has three distinct sections as illustrated in Fig. 2. Firstly, we have simple binary images of characters from the alphabet. These include examples of multiple characters, occlusion and added noisy structure. The second section is more complex and consists of logos. These first two sections of the data-base are used to experiment with exact shape queries. That is to say we are interested in the ability of the algorithm to measure the similarity of shapes when there is no structural corruption present.

The final section of the data-base is used to furnish an example of inexact query when there are structural differences in the patterns. This final section of the data-base is composed of 25 aerial infra-red images collected with a line-scan device. The most prominent feature in these images are road-networks. The reason for this is that our images are taken at night-time. The tarmac roads absorb heat during daylight and radiate strongly in the infra-red at night-time. We have a digital map which corresponds to a small section of road network appearing in two of the images. The digital map and the two infra-red images are shown in Figs. 2(g), (e) and (f), respectively. The images are barrel distorted with respect to the map. Moreover, there are significant structural differences since the map is a cartographers representation of the road network, while the corresponding line-pattern from the aerial image is the output of a segmentation algorithm.

For each image in the data-base in turn we extract line-segments. This is a two-step process.

- *Line detection:* We commence by applying feature detectors to extract pixel-wide edges and lines from the raw imagery [47,48]. Straight line segments have been identified using a variant of the algorithm originally developed by Lowe [49] and later refined by Rosin and West [50]. The basic idea is to find a polygonal approximation to an irregular chain of feature pixels. This is achieved by thresholding the sagittal (or perpendicular) distance from the pixels on the chain to the chord connecting the end-points of the arc. Our refinement of this idea is to make the polygonisation strategy "scale-invariant" by thresholding on the basis of the ratio of the sagittal distance to the chord-length.
- *Construction of the N -nearest neighbour graph:* The centre-points of the straight line segments are used as the nodes of our N -nearest neighbour graphs. The edges are computed by selecting the N nodes that have

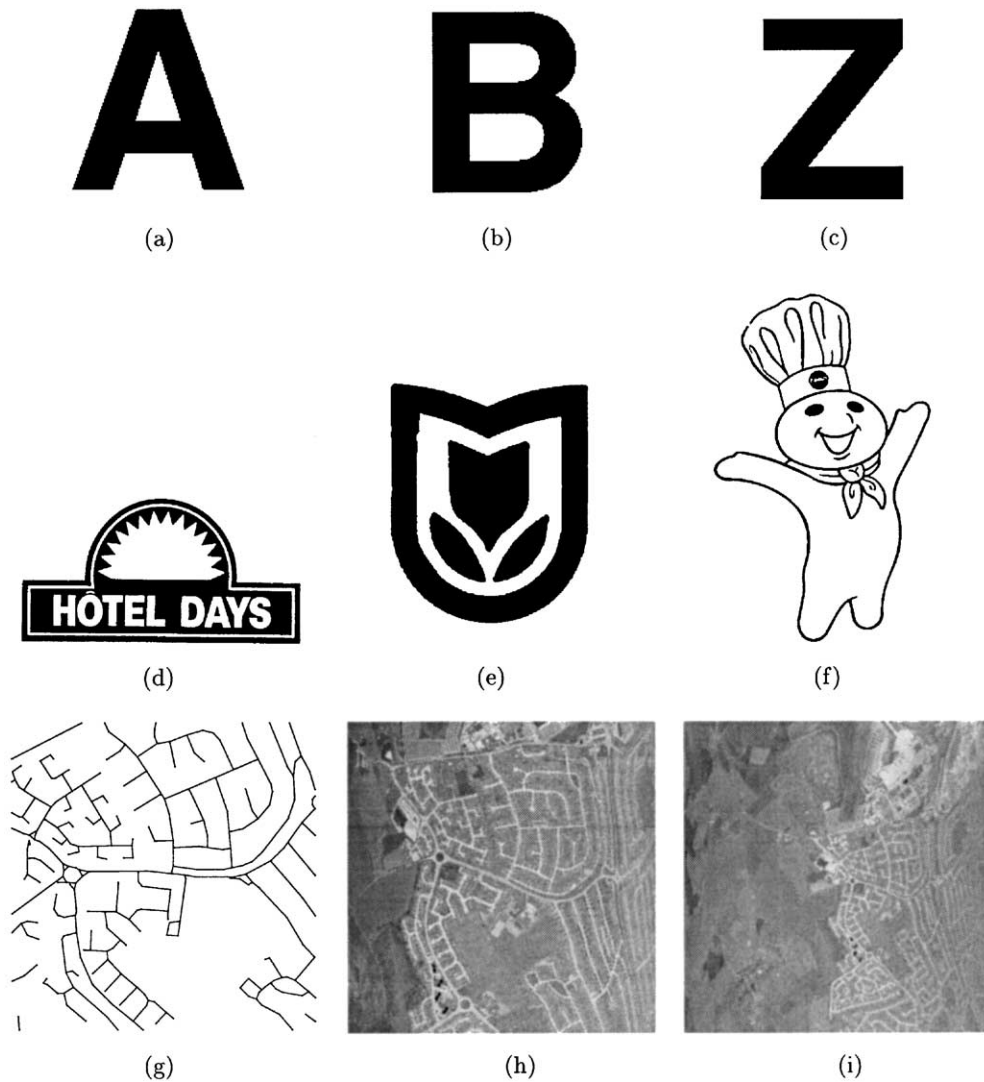


Fig. 2. Images from the data-base. (a) Letter A. (b) Letter B. (c) Letter Z. (d) Logo. (e) Logo. (f) Logo. (g) Digital map. (h) Map 60. (i) Map 170.

the closest Euclidean distance on the image plane. The pairwise attributes for the line-pairs connected by the edges of the N -nearest neighbour graph are used as feature-sets for our probabilistic similarity measure.

Fig. 3 illustrates the sequence of processing steps described above when applied to an aerial infra-red image. Fig. 3(a) is the raw image. The detected pixel chains are shown in Fig. 3(b). In Fig. 3(c) we show the straight-line segments that result from the application of our polygonisation algorithm. The 6-nearest neighbour graph generated from the centre point of the straight line segments is shown in Fig. 3(d).

The process of querying the data-base is conducted as follows. We select a query image and extract the attributed relational graph $G_m = \{V_m, E_m, B_m\}$ represented by the index-set of line-segments V_m , the edges of the N -nearest neighbour graph E_m and the set of pairwise geometric attributes B_m . For each of the line-patterns in the data-base we compare the corresponding attributed relational graph $G_d = \{V_d, E_d, B_d\}$ using the similarity measure defined in Eq. (2). This computation can be performed using any of the weighting functions defined in Section 3.2. The best matching pattern in the data-base satisfies the maximum similarity condition of Eq. (3).

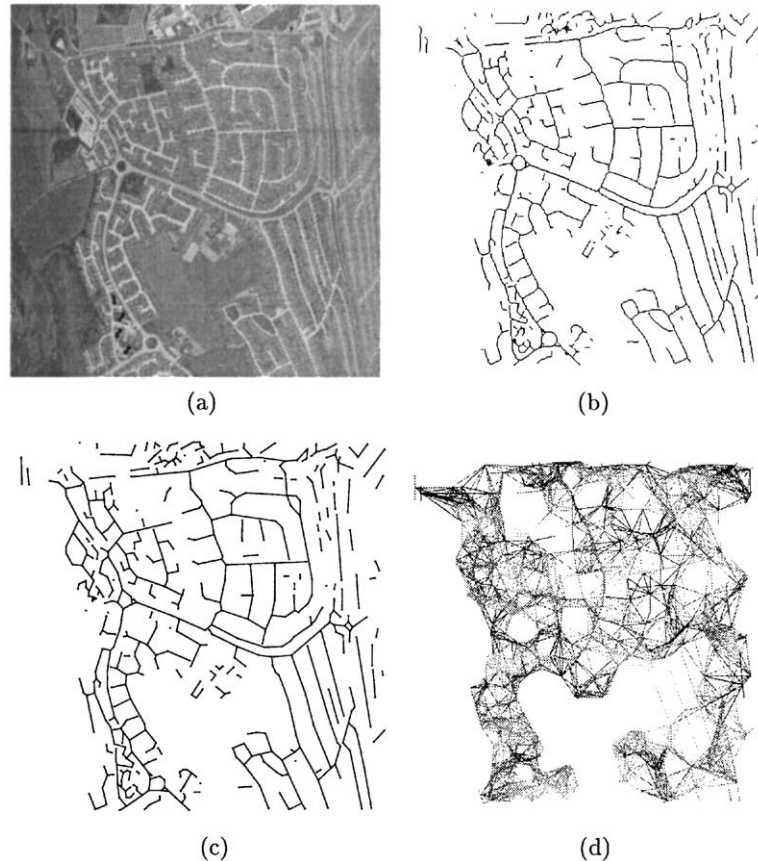


Fig. 3. The series of processing steps used to extract Nearest neighbour graphs. (a) Original image. (b) Raw feature image. (c) Line (polygonised) image. (d) Nearest-Neighbour graph.

In pseudo-code, the retrieval algorithm is as follows:

Begin: Main Program

Read Query Description File; [including
Neighbourhood Graph and Line Segment Coordinate
Information]

For All Query Graph Nodes (line-segments)
Create a Feature Object; [containing the relative
angle and relative position of all connected
line-segments]

For All Model in the Library
Read Model Description File;

For All Model Graph Nodes
Create a Feature Object;
Set SimilarityValue to 0;

For All Query Graph Feature Objects
For All Model Feature Objects

CompareFeatureObjects Query Feature
with Model Feature;

Take the most similar Model Feature Objects;
Add its similarity value to SimilarityValue;

Normalise SimilarityValue;
Sort SimilarityValues;
Show the ordered list;

**Begin: CompareFeatureObjects a Pair Feature
Objects**

Set SimilarityValue to 0;

For All Neighbours in Feature Object 1

For All Neighbours in Feature Object 2

Compute the Distance between each attributes;

Take the most similar feature object attribute;

Add its similarity value to SimilarityValue;

Normalise SimilarityValue;

Return SimilarityValue;

4.2. Algorithm characteristics

Our first set of experiments aim to illustrate the relative recognition performance delivered by the different recognition schemes described in this paper. Here, we investigate the effect of using the different weighting kernels

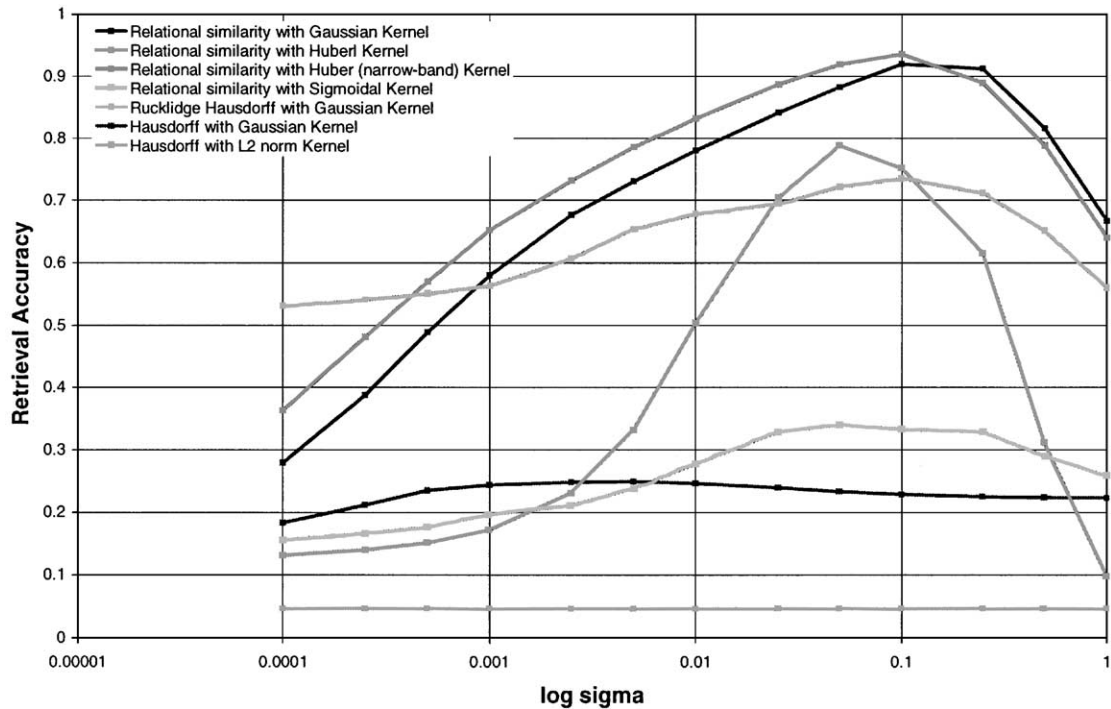


Fig. 4. Relative recognition performance for various distance measures.

described in Section 3.2. We compare with the recognition performance delivered by both the standard Hausdorff distance and the f th quartile variant defined in Eq. (4). We also investigate the effect of applying the Hausdorff tests to the robust weights using the dissimilarity measure defined in Eq. (5).

Our experiments are conducted with line-patterns from the entire data-base. Each of our performance results is obtained as follows. Our queries are selected by drawing 100 patterns at random from the data-base. Each of these randomly selected patterns is used in turn as a query to select the most similar pattern from the data-base. We count the number of times that the query and the retrieved pattern have the same identity. We report fraction of correctly retrieved patterns using the AVRR/IAVRR ratio [51]. Here, IAVRR is the ideal average rank of relevant items and AVRR is the actual average rank of relevant items. Perfect retrieval results in an AVRR/IAVRR ratio of unity.

Fig. 4 shows the recognition performance as a function of the width parameter σ for each of the recognition schemes in turn. When we use the probabilistic similarity measure in conjunction with the each of the weighting kernels described in Section 3.2, we find that the best performance is obtained when the weighting kernel is either Gaussian (black) or a modified narrow-band Huber (red). In each case, however, the performance is considerably

better than the results obtained with the various Hausdorff distances. The poorest performance is obtained with the crisp Hausdorff distance coupled with the $L2$ norm (orange). Rucklidge's modified Hausdorff distance (using a quantile-test rather than a max comparator [12]) does not do well by comparison. However, it does provide an obvious improvement over the standard Hausdorff distance (green). It is important to note that the x -axis of the plot is logarithmic and therefore that recognition performance is not particularly sensitive to the kernel width parameter σ . The maximum achievable retrieval performance is 94%.

Based on its performance and ease of control, we confine our attention to the use of a Gaussian weighting kernel in conjunction with the probabilistic similarity measure as the main test algorithm in the remainder of the paper.

In the next set of experiments we investigate the effect of relational structure on the recognition process. In all cases we confine our attention to the relational similarity measure of Eq. (2) with a Gaussian weighting kernel. Here, we compare the performance obtained with N -nearest neighbour graphs of various orders. We also provide results for the recognition performance obtained when the relational constraints are weakened. We do this in two ways. In the first example, we relax the requirement for neighbourhood structure, and evaluate the

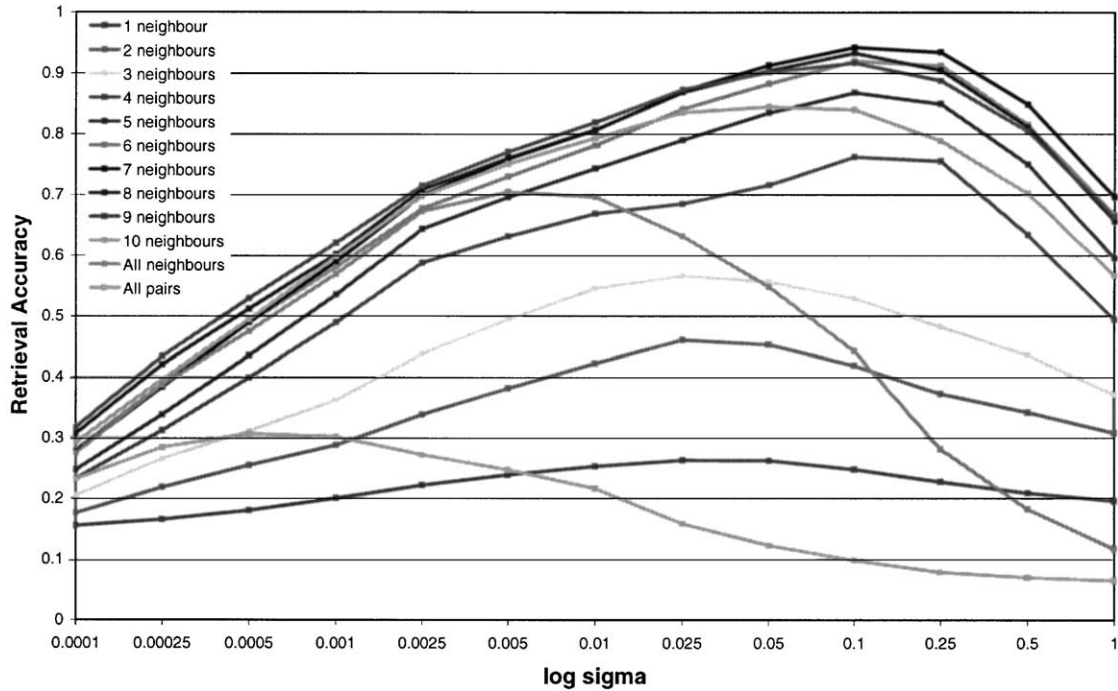


Fig. 5. Relative recognition performance for various relational structure.

similarity measure over all possible edge associations. In the second example we remove the edge-structure and compute the similarity measure over the complete space of pairwise associations.

Fig. 5 shows the recognition accuracy as a function of the width parameter σ for when different relational constraints are used in the recognition process. There are a number of conclusions that can be drawn from this set of experiments. The first observation that can be drawn is that the best recognition performance is obtained when the order of the nearest neighbour graph is seven. Here a performance rate of 95% is achievable. The poorest performance is obtained when the order of the graph is 1. Although performance falls when the order is increased beyond seven, it does not degrade significantly. For instance the maximum retrieval accuracy achievable with an order of 10 is 92%.

The best performance rate that can be obtained when the neighbourhood structure is relaxed is 70%. When edge structure is removed altogether, then the best performance is 30%. This is better than that obtained with a 1-NN graph but poorer than that obtained with a 2-NN graph.

However, even when the order of the graph is small (i.e. one) or large (i.e. ten), the recognition performance exceeds that obtained when either the neighbourhood structure or the edge-structure is ignored.

4.3. Recognition examples

We now provide some examples of the recognition results obtained from the data-base. We present the results as thumbnails ordered according to similarity with the query.

The first experiment involves querying the data-base with a letter "A". There are 12 images of the letter "A" subjected to various rotations and scalings. In some of these images the letter "A" appears alongside another letter or has random clutter lines superimposed. In Fig. 6, we compare the recognition results obtained with the relational histogram and the new similarity measure. In the case of the new similarity measure, the 12 images appear in the top ranked positions.

Fig. 7 shows the result of querying with one of the logos. There are four versions of this logo in the data-base. Each has the same overall shape, but has a different caption. The result gives the exact query in first position and the three similar logos in the next three positions. Moreover, the result exhibits good clustering of logos with a circular component in the top-ranked positions. In particular the "Branigans" and "Crush" logos, which are also composed of semi-circular and rectangular blocks, appear near the top.

Finally, Fig. 8 shows the result of querying the data-base with the digital map. The digital map and its

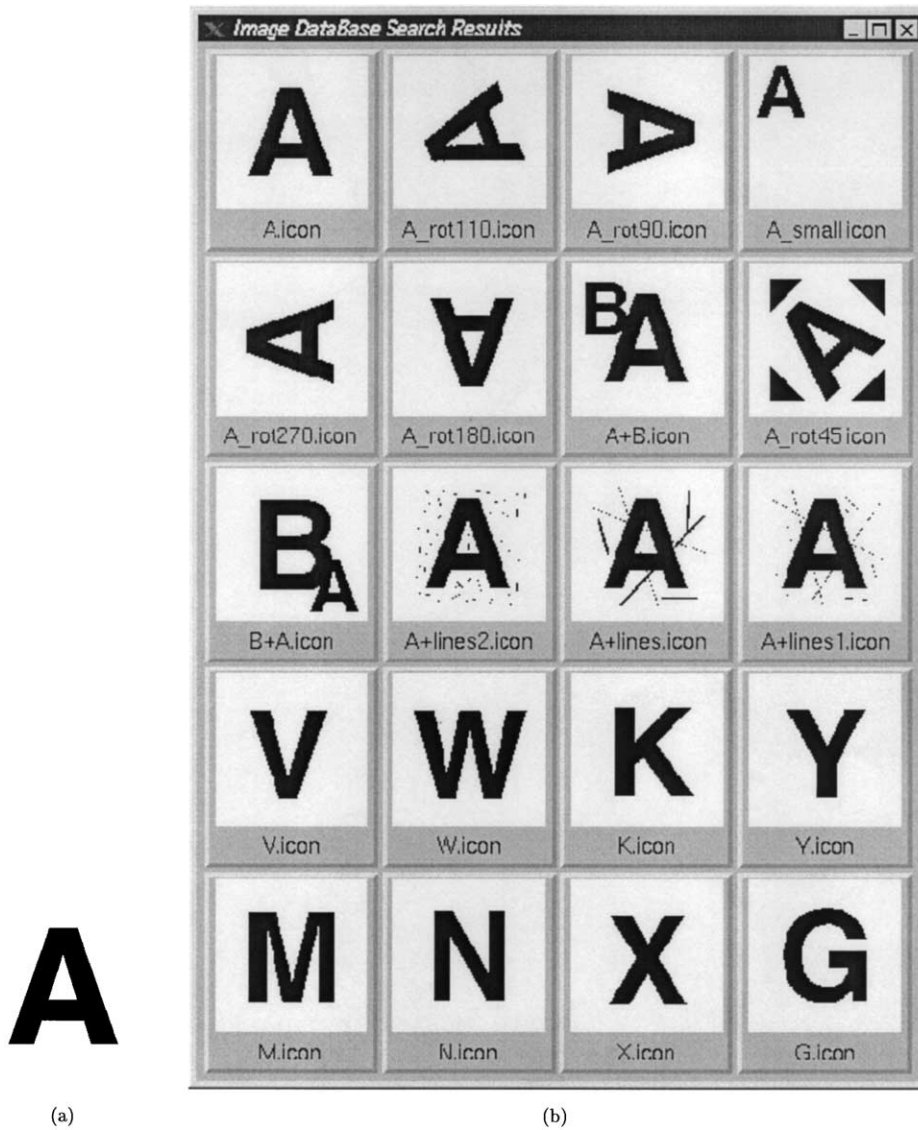


Fig. 6. The result of querying the data-base with the letter “A”: The images are ordered from left-to-right and top-to-bottom in decreasing similarity with the query image, which is shown at the top of the figure.

corresponding target infra-red images are shown in Fig. 2. This is an inexact query since the query image is not part of the data-base. In order to explore the sensitivity of our recognition method to segmentation systematics, we have introduced multiple segmentations of the target images into the data-base. These different segmentations have been obtained by maliciously adjusting the control parameters of the feature extraction algorithm. In total there are eight different segmentations for each of the two target images. The eight segmentations of the two images containing the road-pattern are recalled in the top-ranked positions.

Finally, in Table 1 we list the values of the similarity measure $Q(G_d, G_m)$ for the different images shown in Figs. 6–8.

5. Sensitivity analysis

Having established that the relational similarity measure in conjunction with the Huber kernel offers the most effective recognition performance, we turn to measuring its noise sensitivity. We pose this as a comparative study. Here, we compare the performance with

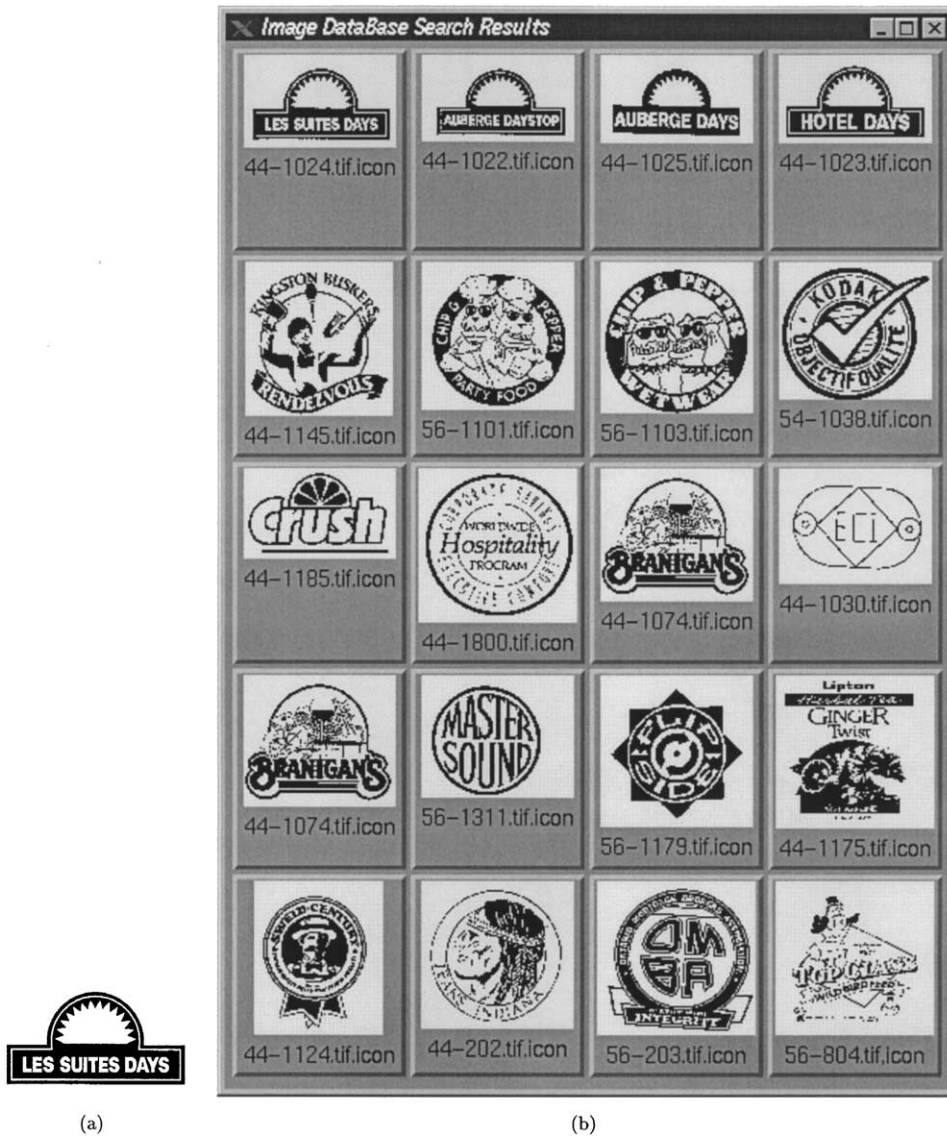


Fig. 7. The result of querying the data-base with the “Le Suites Days” logo: The images are ordered from left-to-right and top-to-bottom in decreasing similarity with the query image, which is shown at the top of the figure.

a histogram-based method [11] and a graph-matching method [52], both reported elsewhere in the literature and reproduced in Appendix A for completeness. These algorithms and some of their sensitivity data have already been reported elsewhere in the literature. We reproduce the data here in order to make the paper self-contained.

The algorithms are compared under (a) the addition of extra lines, (b) line-deletion, (c) line-splitting, (d) line end-point position errors and (e) a combination of the errors. The procedure used to simulate these errors is outlined in Appendix B.

The first set of data reports the results of exact query of the data-base where the single pattern to be retrieved from the data-base is identical to the query apart from changes in translation, scale, rotation or any of the simulated segmentation errors described above. Figs. 9(b)–(c) show the results obtained with a relational histogram (see [11] and Appendix A for more details), the relational similarity measure described in this paper, and a graph-matching method (see [52] and Appendix A for more details). When the relational similarity measure is used, then recognition accuracy does not degrade until



Fig. 8. The images are ordered from left-to-right and top-to-bottom in decreasing similarity with the query image.

the level of error reaches 70%. It is the missing or deleted lines that pose the main difficulty. The method is relatively insensitive to the addition or splitting of lines. It is important to stress that worst performance curve for the relational similarity measure is better than the best curve obtained with the relational histogram. The method does not perform significantly worse than the graph-matching method.

We have repeated these experiments with an inexact query. Here the query pattern is a distorted version of the target in the data-base. An example is furnished by the digital map (see Fig. 2(a)) which is a barrel-distorted version of its corresponding target aerial images. Figs.

10(a) and (b) again show the retrieval accuracy as a function of the fraction of segmentation errors. A more complex sensitivity pattern emerges in this case. In the case of the relational similarity measure (Fig. 10(a)) the increased importance of line-splitting errors emerges. It is again large number of missing lines that limit the effectiveness of the technique. However, the onset of errors occurs when as few as 40% of the lines are deleted. The line-patterns are least sensitive to segment end-point errors. In the case of both line-addition and line-splitting there is an onset of errors when the fraction of segment errors is about 20%. However, at larger fractions of segmentation errors the overall effect is significantly less

Table 1
Similarity values for experiments presented in Figs. 6–8

Rank	Query “A”		Query “Les Suites Days”		Query “Digital Map”	
	Image name	Similarity	Image name	Similarity	Image name	Similarity
1	A	1.00	44-1024	1	InfraredImage060	0.67
2	A_rot110	0.75	44-1022	0.78	InfraredImage060_10	0.67
3	A_rot90	0.66	44-1025	0.73	InfraredImage060_20	0.67
4	A_small	0.64	44-1023	0.71	InfraredImage060_07	0.67
5	A_rot270	0.63	44-1145	0.59	InfraredImage060_25	0.66
6	A_rot180	0.61	56-1101	0.53	InfraredImage170	0.65
7	A + B	0.59	56-1103	0.52	InfraredImage170_2	0.63
8	A_rot45	0.50	54-1038	0.49	InfraredImage050	0.63
9	B + A	0.43	44-1185	0.44	InfraredImage120	0.62
10	A + lines2	0.42	44-1800	0.43	InfraredImage100	0.58
11	A + lines	0.39	44-1074 (A)	0.43	InfraredImage090	0.54
12	A + lines1	0.39	44-1030	0.43	InfraredImage160	0.53
13	V	0.37	44-1074 (B)	0.42	InfraredImage180	0.51
14	W	0.35	56-1311	0.42	InfraredImage080	0.48
15	K	0.29	56-1179	0.41	InfraredImage200	0.48
16	Y	0.28	44-1175	0.4	InfraredImage070	0.47
17	M	0.27	44-1124	0.4	InfraredImage150	0.44
18	N	0.26	44-202	0.38	54-1193	0.42
19	X	0.24	56-203	0.37	InfraredImage110	0.42
20	G	0.21	56-804	0.37	54-1187	0.41

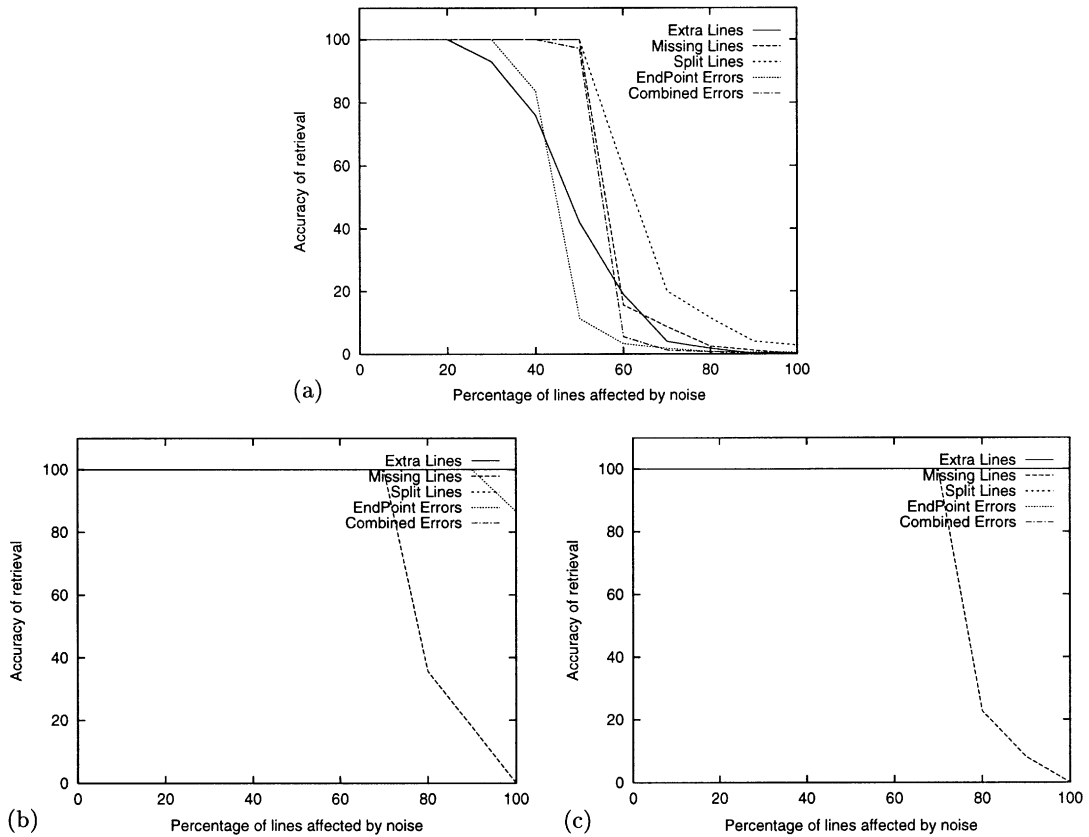


Fig. 9. Effect of various kinds of noise on the retrieval performance when the target and the query are the same apart from added noise. (a) Relational attribute histograms. (b) Feature set matching. (c) Graph-matching.

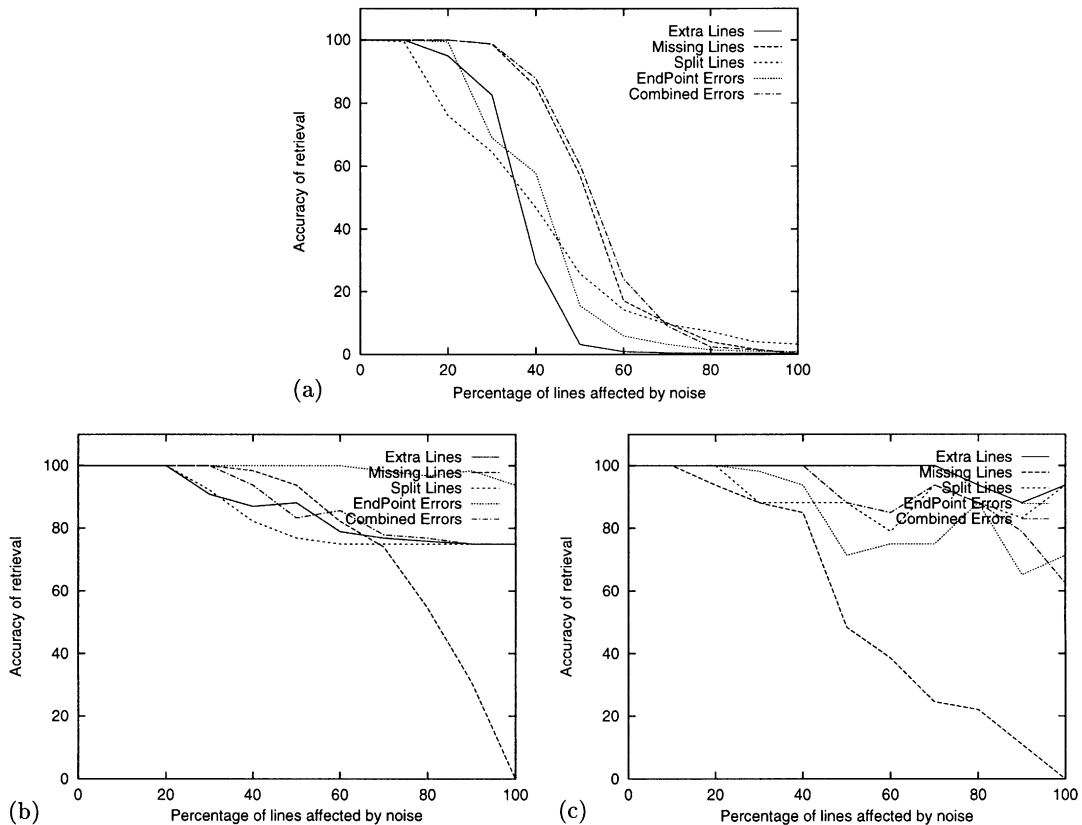


Fig. 10. Effect of various kinds of noise on the retrieval performance when the targets and the query are the similar. (a) Relational histograms. (b) Relational similarity. (c) Graph-matching.

marked than in the case of line-deletions. Again, the performance curves obtained using the relational similarity measure are consistently better than those obtained using histograms. This is reflected not only in terms of larger values of the onset error, but also in terms of smaller rates of degradation. Comparing the results obtained with the relational similarity measure and those obtained with the graph matching method, there is little difference. It seems that the effort expended in trying to improve the pattern of correspondences returns marginal performance improvements.

6. Conclusion

In this paper, we have presented a new similarity measure for comparing relational object descriptions. The idea underpinning the measure is to gauge the similarity of the pairwise attributes residing on the edges of a graph structure that represents the proximity structure of a set of object-primitives. The measure exploits the neighbourhood structure to limit the set of comparisons required.

For a data-base of 2500 objects (or line-patterns) we have shown that a recall accuracy of over 94% is achievable when the weighting function is Gaussian. We have presented a number of experiments demonstrating the performance of the proposed methodology. Moreover, the results obtained indicate that the method is relatively insensitive to the under and over segmentation of the line-patterns. Moreover, the method consistently outperforms the Hausdorff distance in terms of its recognition performance.

Our future plans are two-fold. Firstly, we aim to use the new relational similarity measure to perform clustering on the patterns in the data-base. The aim here is to group the patterns according to similarity and to use the resulting equivalence classes to perform indexation. The second aim is to treat the three recognition methods reported here as classifiers and to combine their decisions. In this way we can exploit their complementarity under the different kinds of error investigated in Section 6.

This graph-based recognition process can be viewed as an intermediate step in a coarse-to-fine object retrieval system. As we envisage the process, a set of candidate images is retrieved using a coarse-grained representa-

tion. In fact, in our previously reported work we have provided a tangible example of how such a set of candidates can be located by comparing pairwise geometric histograms [11]. Once overall object recognition has been achieved then detailed correspondences may be recovered. Here, techniques such as graph-matching [25] and template alignment [53] can be used to verify recognition hypotheses and initiate new searches if necessary.

Appendix A. Alternative algorithms

In this Appendix, we present brief details of the algorithms used in the comparative evaluation of the new retrieval algorithm presented in this paper. These methods have been reported separately in the literature [11,52] and we reproduce details here for the sake of completeness. The first algorithm uses a global histogram of pairwise geometric attributes on the edges of a nearest neighbour graph. Secondly, we compare with a graph-matching algorithm. Here, the similarity of the local attribute histograms is used to provide explicit node correspondences. For each line-pattern in the data-base, the algorithm iteratively recovers the set of maximum a posteriori probability correspondences with the query pattern. Once the detailed correspondences are at hand, recognition is effected by finding the library pattern which has the largest probability of match against the query pattern.

A.1. Attribute histograms

Each node in the shape graph, i.e. each line in the pattern, is represented by the histogram of its pairwise geometric attributes to the remaining lines in the pattern. This histogram can be thought of as a local estimate of the probability distribution for the pairwise attributes. Accordingly, the angle and position attributes $\theta_{a,b}$ and $\vartheta_{a,b}$ are binned in a histogram. Rather than binning the geometric attributes over all node-pairs, we restrict ourselves to pairs of nodes that are connected by edges in the shape graph. Suppose that $S_a(\mu, \nu) = \{(a, b) | \theta_{a,b} \in A_\mu \wedge \vartheta_{a,b} \in R_\nu \wedge (a, b) \in E_d\}$ is the set of nodes whose pairwise geometric attributes with the node a are spanned by the interval of directed relative angles A_μ and the relative position attribute interval R_ν . The contents of the histogram bin spanning the two attribute intervals are given by $H_a(\mu, \nu) = |S_a(\mu, \nu)|$. Each histogram contains n_A relative angle bins and n_R length ratio bins. The normalised geometric histogram bin-entries are computed as follows:

$$h_a(\mu, \nu) = \frac{H_a(\mu, \nu)}{\sum_{\mu'=1}^{n_A} \sum_{\nu'=1}^{n_R} H_a(\mu', \nu')}.$$

We use the notation h^d to denote the normalised histogram for the graph G_d from the database and h^m to denote the query histogram.

To perform recognition using the pairwise geometric histograms we conglomerate the node histograms into a global histogram. This histogram provides a statistical summary for the pairwise attributes residing on the edges of a nearest neighbour graph. The normalised histogram bin-contents is given by

$$\hat{h}_T^d(\mu, \nu) = \frac{\sum_{a \in V_d} H_a(\mu, \nu)}{\sum_{a \in V_d} \sum_{\mu'=1}^{n_A} \sum_{\nu'=1}^{n_R} H_a(\mu', \nu')}.$$

The best-matching pattern is retrieved from the data-base on the basis of similarity with the query pattern histogram. Our similarity measure is the histogram correlation. The measure of pattern correlation is the Bhattacharyya distance. The class identity of the retrieved pattern is

$$\omega_m = \arg \max_{d \in \mathcal{D}} \ln \sum_{\mu=1}^{n_A} \sum_{\nu=1}^{n_R} \sqrt{h_T^m(\mu, \nu) \times h_T^d(\mu, \nu)}.$$

This is the least computationally demanding of our retrieval algorithms. Once the histograms have been pre-compiled and normalised, then the computational overheads are purely related to the number of bin comparisons that must be performed. Each query requires $n_A \times n_R$ bin comparisons to be made. Typically, the dimensions of the histograms are 36×12 bins.

A.2. Graph matching

The relational similarity measure does not utilise any information concerning the consistency of the arrangement of correspondences between the individual graphs. This clearly represents an important additional information source which could be used to improve recognition performance. In order to exploit the consistency of correspondence, we use a simplification of the graph-matching scheme developed by Finch et al. [54] which was recently reported by Huet and Hancock [52]. This poses the retrieval process as the one associating with the query the graph from the data-base that has the largest a posteriori probability of match. In other words, the class identity of the graph which most closely corresponds to the query is

$$\omega_m = \arg \max_{d \in \mathcal{D}} P(G_d | G_m).$$

However, since we wish to make a detailed structural comparison of the graphs, rather than comparing their overall statistical properties, we must first establish a set of best-match correspondences between each ARG in the data-base and the query G_m . At iteration n the set of correspondences between the query G_m and the ARG G_d is a relation $f_d^n : V_d \mapsto V_m$ over the vertex sets of the two graphs. The mapping function consists of a set of Cartesian pairings between the nodes of the two graphs, i.e. $f_d^n = \{(a, \alpha) | a \in V_d, \alpha \in V_m\} \subseteq V_d \times V_m$. The retrieved

pattern is the one which has the most consistent pattern of correspondences and satisfies the condition

$$\omega_m = \arg \max_{d \in \mathcal{D}} \max_{f_d} P(f_d | G_d, G_m).$$

The pattern of correspondences is assigned to satisfy the following *maximum* a posteriori probability condition

$$f_d^n(a) = \arg \max_{x \in V_m} p(x_a, x_x | f_d^n(a) = \alpha) P(f_d^n | E_d, E_m).$$

Suppose that we use the notation

$$s_{a,x}^n = \begin{cases} 1 & \text{if } f_d^n(a) = \alpha, \\ 0 & \text{otherwise} \end{cases}$$

to represent the correspondence assignment. The consistency of global match against the query pattern can be improved by iterating the assignment condition

$$f_d^n(a) = \arg \max_{x \in V_m} \left[\ln p(x_a, x_x | f_d^n(a) = \alpha) + \sum_{(a,b) \in E_d} \sum_{(\alpha,\beta) \in E_m} \{ \ln(1 - P_e) s_{a,x}^{n-1} s_{b,\beta}^{n-1} + \ln P_e (1 - s_{a,x}^{n-1} s_{b,\beta}^{n-1}) \} \right]. \quad (6)$$

The probability of match between the pattern-vectors is again computed using the Bhattacharyya coefficient between the normalised histograms

$$P(f_d^n(a) = \alpha | x_a, x_x) = \frac{\sum_{\mu=1}^{n_A} \sum_{\nu=1}^{n_R} \sqrt{h_a(\mu, \nu) h_x(\mu, \nu)}}{\sum_{x \in V_m} \sum_{\mu=1}^{n_A} \sum_{\nu=1}^{n_R} \sqrt{h_a(\mu, \nu) h_x(\mu, \nu)}} = \exp[-B_{a,x}]. \quad (7)$$

With this modelling ingredient, and using the correspondence matches delivered by the graph-matching scheme outlined in Eq. (8), the condition for recognition is

$$\omega_m = \arg \max_{d \in \mathcal{D}} \sum_{(a,b) \in E_d} \sum_{(\alpha,\beta) \in E_m} \{ -B_{a,x} - B_{b,\beta} + \ln(1 - P_e) s_{a,x}^n s_{b,\beta}^n + \ln P_e (1 - s_{a,x}^n s_{b,\beta}^n) \}. \quad (8)$$

This is the most computationally demanding of the methods. For each iteration of the correspondence matching process, there are $|V_d| \times |V_m| \times |E_d| \times |E_m|$ consistency computations to be done. This should be compared with the $|E_d| \times |E_m|$ edge probability computations that must be performed for the method described in Section 3.1. In other words the computation of the edge matching probability is of order $|V_d| \times |V_m|$ more complex per iteration (typically four iterations are required). The retrieval process, on the other hand, requires $|E_d| \times |E_m|$ computations to be performed in order to measure the similarity between graphs. For an N -nearest neighbour

graph, the number of computations involved in this step is $N^2 |V_d| \times |V_m|$; this is identical to that required for the probabilistic similarity measure described in Section 3.1. In other words, although there are no significant differences in the complexity of the retrieval computations, the iterative computation of consistency does introduce additional expense.

Appendix B. Simulated pattern corruption errors

The sensitivity analysis presented in Section 5 focuses on the behaviour of the new retrieval method under various types of pattern corruption error. The experimental procedure used in this analysis has already been detailed elsewhere in the literature [11,52]. In this appendix, we reproduce a description of the types of pattern error studied in the analysis.

The analysis focuses on the sensitivity of the recognition schemes to the systematics of the line-segmentation process. To this end we have simulated the segmentation errors that can occur when line-segments are extracted from realistic image data. Specifically, the different processes that we have investigated are listed below:

- *Extra lines*: Here we have added additional lines at random locations. The lengths and angles of the added lines have generated by randomly sampling distribution for the existing image-segments.
- *Missing lines*: Here we have deleted a known fraction of line-segments at random locations.
- *Split lines*: Here a predefined fraction of lines have been split into two segments. The splitting process is effected by deleting an internal fraction of each line-segment. The deleted segment is randomly positioned along the line. The fraction of the line deleted is uniformly sampled from the range (0, 1).
- *Segment errors*: Here we have introduced random displacements in the end-point positions for a predefined fraction of lines. The distribution of end-point errors is Gaussian. The degree of error is controlled by the variance of the Gaussian distribution.
- *Combined errors*: Here we have introduced the four different segment errors described above in equal proportion.

The performance measure used in our studies is computed as follows. We query the data-base with a sample of line patterns. For each pattern in turn we determine whether or not the correct retrieval occurs in the top-ranked position. By computing the fraction of queries that return a correctly recognised recall, we determine the average retrieval accuracy. We compare the recall systematics of our set-based recognition strategy with that obtained using the relational histogram-based method.

References

- [1] M.J. Swain, Interactive indexing into image databases, *Image Vision Storage Retrieval* (1993) 95–103.
- [2] G.L. Gimelfarb, A.K. Jain, On retrieving textured images from an image database, *Pattern Recognition* 29 (9) (1996) 1461–1483.
- [3] C. Dorai, A.K. Jain, View organisation and matching of free-form objects, *IEEE Computer Society International Symposium on Computer Vision*, 1995, pp. 25–30.
- [4] I. Rigoutsos, R. Hummel, A bayesian approach to model matching with geometric hashing, *Comput. Vision Image Understand.* 62 (26) (1995) 11.
- [5] C. Dorai, A.K. Jain, COSMOS: a representation scheme for 3D free-form objects, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (1997) 1115–1130.
- [6] A.C. Evans, N.A. Thacker, J.W.E. Mayhew, The use of geometric histograms for model-based object recognition, *Proceedings of the 4th British Machine Vision Conference*, Sept. 1993, pp. 429–438.
- [7] M.S. Costa, L. Shapiro, Scene analysis using appearance-based models and relational indexing, *IEEE Computer Society International Symposium on Computer Vision*, 1995, pp. 103–108.
- [8] K. Sengupta, K.L. Boyer, Organising large structural databases, *IEEE Trans. Pattern Anal. Mach. Intell.* 17 (4) (1995) 321–332.
- [9] A.J. Bray, V. Hlavac, Properties of local geometric constraints, *Proceedings of the 2nd British Machine Vision Conference*, Sept. 1991, pp. 95–103.
- [10] D.H. Ballard, Generalizing the Hough transform to detect arbitrary shapes, *Pattern Recognition* 13 (2) (1981) 111–122.
- [11] B. Huet, E. Hancock, Line pattern retrieval using relational histograms, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (1999) 1363–1370.
- [12] W.J. Rucklidge, Locating objects using the Hausdorff distance, *IEEE International Conference on Computer Vision*, 1995, pp. 457–464.
- [13] X. Yi, O.I. Camps, Line-based recognition using a multidimensional Hausdorff distance, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (September 1999) 901–916.
- [14] S. Santini, R. Jain, Similarity measures, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (1999) 871–883.
- [15] A. Tversky, Features of similarity, *Psychol. Rev.* 84 (1977) 327–352.
- [16] W.E.L. Grimson, *Object Recognition by Computer: the Role of Geometric Constraints*, MIT Press, Cambridge, MA, 1990.
- [17] Y. Lamdan, H.J. Wolfson, Geometric hashing: a general and efficient model-based recognition scheme, *Proceedings of the IEEE International Conference on Computer Vision*, 1988, pp. 238–249.
- [18] F. Stein, G. Medioni, Structural indexing: efficient 2D object recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (1992) 1192–1204.
- [19] C.A. Rothwell, A. Zisserman, D. Forsyth, J. Mundy, Canonical frames for planar object recognition, *European Conference on Computer Vision*, 1992, pp. 757–772.
- [20] Y. Lamdan, J.T. Schwartz, H.J. Wolfson, Object recognition by affine invariant matching, in: *Proceedings of the conference on Computer Vision and Pattern Recognition* 1998, pp. 335–344.
- [21] L.G. Shapiro, R.M. Haralick, Structural description and inexact matching, *IEEE Trans. Pattern Anal. Mach. Intell.* 3 (1981) 504–519.
- [22] L.G. Shapiro, R.M. Haralick, A metric for comparing relational descriptions, *IEEE Trans. Pattern Anal. Mach. Intell.* 7 (1) (1985) 90–94.
- [23] A. Sanfeliu, K.S. Fu, A distance measure between attributed relational graph, *IEEE SMC* 13 (1983) 353–362.
- [24] A.K.C. Wong, M. You, Entropy and distance of random graphs with application to structural pattern recognition, *IEEE Trans. Pattern Anal. Mach. Intell.* 7 (1985) 599–609.
- [25] R. Wilson, E.R. Hancock, Structural matching by discrete relaxation, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (1997) 634–648.
- [26] R.C. Wilson, A.D.J. Cross, E.R. Hancock, Structural matching with active triangulation, *Computer Vision Image Understand.* 72 (1998) 21–38.
- [27] A.M. Finch, R.C. Wilson, E.R. Hancock, Matching Delaunay graphs, *Pattern Recognition* 30 (1997) 123–140.
- [28] W. Christmas, J. Kittler, M. Petrou, Structural matching in computer vision using probabilistic relaxation, *IEEE Trans. Pattern Anal. Mach. Intell.* 17 (1995) 749–764.
- [29] S.C. Zhu, A.L. Yuille, Forms: a flexible object recognition and modelling system, *IJCV* 20 (3) (1996) 187–212.
- [30] T.L. Liu, D. Geiger, Visual deconstruction: recognizing articulated object, in: *International Workshop EMMCVPR'97 Lecture Notes in Computer Science*, Vol. 1223, 1997, pp. 295–310.
- [31] D. Mumford, J. Shah, Boundary detection by minimizing functionals, in: *CVPR85*, 1985, pp. 22–26.
- [32] Y. Amit, A. Kong, Graphical templates for model registration, *IEEE Trans. Pattern Anal. Mach. Intell.* 18 (1996) 225–236.
- [33] M. Leyton, A process grammar for shape, *Artif. Intell.* 34 (1988) 213–247.
- [34] K. Siddiqi, A. Shokoufandeh, S.J. Dickinson, S.W. Zucker, Shock graphs and shape matching, in: *Proceedings of the IEEE International Conference on Computer Vision*, 1998, pp. 222–229.
- [35] S.W. Reyner, An analysis of a good algorithm for the subtree problem, *SIAM J. Comput.* 6 (1977) 730–732.
- [36] M. Pelillo, K. Siddiqi, S. Zucker, Matching hierarchical structures using association graphs, in: *ECCV98*, 1998, pp. 3–16.
- [37] L. Liu, S. Sclaroff, Index trees for efficient deformable shape-based retrieval, in: *IEEE Workshop on Content-Based Access of Image and Video Libraries*, 2000, pp. 83–87.
- [38] S. Belongie, C. Carson, H. Greenspan, J. Malik, Color- and texture-based image segmentation using em and its application to content-based image retrieval, in: *ICCV98*, 1998, pp. 675–682.
- [39] B. Huet, E.R. Hancock, Relational histograms for shape indexing, *Proceedings of the IEEE International Conference on Computer Vision*, 1998, pp. 563–569.
- [40] R.C. Wilson, A.D.J. Cross, E.R. Hancock, Sensitivity analysis for structural matching, in: *Proceedings of the 13th International Conference on Pattern Recognition*, Vol. 1, August 1996, pp. 62–66.

- [41] N.A. Thacker, P.A. Riocreux, R.B. Yates, Assessing the completeness properties of pairwise geometric histograms, *Image Vision Comput.* 13 (1995) 423–429.
- [42] F. Stein, G. Medioni, Efficient two dimensional object recognition, in: *Proceedings of the International Conference on Pattern Recognition 1990*, pp. 13–17.
- [43] M. Tuceryan, T. Chorzempa, Relative sensitivity of a family of closest-point graphs in computer vision applications, *Pattern Recognition* 24 (5) (1991) 361–373.
- [44] J. Kittler, E.R. Hancock, Combining evidence in probabilistic relaxation, *Int. J. Pattern Recognition and Artif. Intell.* 3 (1) (1989) 29–51.
- [45] D.P. Huttenlocher, G.A. Klanderma, W.J. Rucklidge, Comparing images using the Hausdorff distance, *IEEE Trans. Pattern Anal. Mach. Intell.* 15 (1993) 850–863.
- [46] I. Bloch, Information combination operators for data fusion: A comparative review with classification, *IEEE Trans. Systems, Man Cybernet. Part A* 26 (1996) 52–67.
- [47] E.R. Hancock, Resolving edge-line ambiguities using probabilistic relaxation, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1993, pp. 300–306.
- [48] J. Canny, A Computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (6) (1986) 679–698.
- [49] D.G. Lowe, Three-dimensional object recognition from two-dimensional images, *Artif. Intell.* 31 (3) (1987) 355–395.
- [50] P.L. Rosin, G.A.W. West, Segmentation of edges into lines and arcs, *Image Vision Comput.* 7 (2) (1989) 109–114.
- [51] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, W. Equitz, Efficient and effective querying by image content, *J. Intell. Inform. Systems* 3 (1994) 715–729.
- [52] B. Huet, E. Hancock, Shape recognition from large image libraries by inexact graph matching, *Pattern Recogn. Lett.* 20 (1999) 1259–1269.
- [53] A.D.J. Cross, E.R. Hancock, Recovering perspective pose with dual step EM algorithm, *Adv. Neural Inform. Process. Systems* 10 (1998) 780–786.
- [54] A.M. Finch, R.C. Wilson, E.R. Hancock, An energy function and continuous edit process for graph matching, *Neural Computat.* 10 (1998) 1873–1894.

About the Author—BENOIT HUET received his B.Sc. degree in Computer Science and Engineering from the Ecole Supérieure de Technologie Electrique (Groupe ESIEE, France) in 1992. In 1993, he was awarded the M.Sc. degree in Artificial Intelligence from the University of Westminster (UK) with distinction, where he then spent two years working as a research and teaching assistant. He received his D.Phil degree in Computer Science from the University of York (UK) in 1999, for his research on the topic of object recognition from large databases. He is currently working as a research and teaching associate in the multimedia information processing group of the Institut Eurecom (France). He has published some 30 papers in journals, edited books and refereed conferences. His research interests include computer vision, content-based retrieval, multimedia data indexing (still and/or moving images) and pattern recognition.

About the Author—EDWIN HANCOCK studied Physics as an undergraduate at the University of Durham and graduated with honours in 1977. He remained at Durham to complete the Ph.D. in the area of high energy physics in 1981. Following this he worked for 10 years as a researcher in the fields of high-energy nuclear physics and pattern recognition at the Rutherford-Appleton Laboratory (now the Central Research Laboratory of the Research Councils). During this period he also held adjunct teaching posts at the University of Surrey and the Open University. In 1991 he moved to the University of York as a lecturer in the Department of Computer Science. He was promoted to Senior Lecturer in 1997 and to Reader in 1998. In 1998 he was appointed to a Chair in Computer Vision. Professor Hancock now leads a group of some 15 faculty, research staff and Ph.D. students working in the areas of computer vision and pattern recognition. His main research interests are in the use of optimisation and probabilistic methods for high and intermediate level vision. He is also interested in the methodology of structural and statistical pattern recognition. He is currently working on graph-matching, shape-from-X, image data-bases and statistical learning theory. His work has found applications in areas such as radar terrain analysis, seismic section analysis, remote sensing and medical imaging. Professor Hancock has published some 60 journal papers and 200 refereed conference publications. He was awarded the Pattern Recognition Society medal in 1991 for the best paper to be published in the journal *Pattern Recognition*. The journal also awarded him an outstanding paper award in 1997. Professor Hancock has been a member of the Editorial Boards of the journals *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and, *Pattern Recognition*. He has also been a guest editor for special editions of the journals *Image and Vision Computing and Pattern Recognition*, and he is currently a guest editor of a special edition of *IEEE Transactions on Pattern Analysis and Machine Intelligence* devoted to energy minimisation methods in computer vision. He has been on the programme committees for numerous national and international meetings. In 1997 he established a new series of international meetings on energy minimisation methods in computer vision and pattern recognition.