# Object Recognition from Large Libraries of Line Patterns

Benoit Huet, M.Sc.

Submitted for the degree of Doctor of Philosophy

Department of Computer Science

THE UNIVERSITY of York

May 19, 1999

**Abstract**

The overall aim of this thesis is to provide a hierarchical framework of methodologies for recognising objects represented as line patterns from large structural libraries.

One of the novel aspects of our work is a new shape representation for rapidly indexing and recognising line-patterns from large databases. The basic idea is to exploit both geometric attributes and structural information to compute a two-dimensional relational pairwise geometric histogram. Shapes are indexed by searching for the line-pattern that maximises the cross-correlation of the normalised histogram bin-contents. A sensitivity study reveals that the structural gating of the histogram not only improves recognition performance, but it also overcomes the problem of saturation when large patterns are being recalled. This technique provides the first level of the hierarchy, which is used to prune the database of many unwanted candidates.

The intermediate level of our hierarchical framework is based on a novel similarity measure for object recognition from large libraries of line-patterns. This operates at a more local image level than the histogram based indexing layer. The measure is derived from a Bayesian consistency criterion and resembles the Hausdorff distance. This consistency criterion has been developed for locating correspondence matches between attributed relational graphs using iterative relaxation operations. Our aim here, is to simplify the consistency measure so that it may be used in a non-iterative manner without the need to compute explicit correspondence matches. This considerably reduces the computational overheads and renders the consistency measure suitable for large-scale object recognition. A sensitivity study reveals that the method is capable of delivering a recognition accuracy of 94%.

A Bayesian graph matching algorithm for data-mining from large structural databases operates as final level of the hierarchy. The matching algorithm uses both edge-consistency and node attribute similarity to determine the *a posteriori* probability of a query graph for each of the candidate matches in the reduced database generated by the lower levels of the hierarchy. The node feature-vectors are constructed by computing normalised histograms of pairwise geometric attributes. Attribute similarity is assessed by computing the Bhattacharyya distance between the histograms. Recognition is realised by selecting the candidate with the largest *a posteriori* probability. Here the recognition technique is shown to significantly outperform a number of algorithm alternatives.

For each of the above methodologies a thorough sensitivity study is undertaken for a library of over 2500 lines-patterns. We investigate the algorithms under line-dropout, line fragmentation, line addition and line end-point position errors. The analysis reveals the robustness of each method on its own as well as within the hierarchical framework. This suggests that there is a degree of complementarity between the approaches.

# Contents

# List of Figures

# List of Tables

## Acknowledgments

Firstly, I would like to thank my supervisor Professor Edwin R. Hancock for his guidance and support during the course of this work. My thanks also go to Dr. Graham Finlayson who assessed my work over the first half of the research. The second part of this work was assessed by Dr. Richard C. Wilson to whom I address my thanks.

Thanks also go to the University of York for the funding of my research scholarship. The department of Computer Science at the University Of York has provided the equipment and financial support, which have made this work possible.

I would like to thank Andrew Cross and Richard Wilson for their valuable discussions and comments as well as their technical help.

I wish to thank my family and more specifically my parents for their help and support throughout all my years as a student.

Finally, I would like to thank my fiancée Fernanda, who helped me and provided moral support both during the good and stressful times, which have led to the writing of this thesis.

# Declaration

I declare that all the work in this thesis is solely my own, except where attributed and cited to another author. Some of the material in the following chapters has been previously published in the cited places; for Chapter 2, (Huet and Hancock, 1996b; Huet and Hancock, 1996a); for Chapter 3, (Huet and Hancock, 1996c; Huet and Hancock, 1997b; Huet and Hancock, 1998c); for Chapter 4, (Huet and Hancock, 1997a; Huet and Hancock, 1998a; Huet and Hancock, 1998b); for Chapter 5, (Huet et al., 1998).

# Chapter 1

# Introduction and Literature Review

## 1.1 Introduction

This thesis is about the retrieval of images from large databases using line-patterns. Line-segments are commonly employed as the basis for representing objects or scenes in computer vision tasks (Huttenlocher and Ullman, 1987; Evans et al., 1993; Princen et al., 1992; Costa and Shapiro, 1995). A large number of object recognition algorithms make direct use of straight-line pixel arrangements detected in digital images although other approaches have also been proposed and offer good recognition performance. Two sources of information are available. Firstly, the recognition process may consist of comparing attributes (or measurements). Secondly, the relational arrangement may be used to perform recognition. Because of the complexity of the task at hand, object recognition algorithms are often computationally expensive and therefore rather slow. Furthermore, current recognition techniques become extremely impractical and inefficient as the number of object model increases.

For content-based image retrieval, shape indexing is the focus of attention (Eakins, 1989; Taubin and Cooper, 1991; Niblack et al., 1993; Sclaroff and Pentland, 1993; Jain and Vailaya, 1996; Picard and Pentland, 1996). However, this problem is approached in a slightly different way than object recognition. Because

of the speed requirement during the retrieval process, new methods have been proposed to achieve shape similarity retrieval. In many cases, these methods are based on boundary descriptions of image regions, which are semi-manually annotated.

Having established the topic of the thesis, we intend to propose a methodology for rapidly indexing and recognising object images into a large library of line-segment representations (or line-patterns). In order to achieve this goal with a high degree of accuracy and robustness as well as an economical amount of processing time, we intend to devise a hierarchical (or multi-level) framework where each level takes as input domain the result obtained by the previous level. The first (lowest) level employs image summaries (or signatures) to represent the shape content of the images. These compact summaries are easily and rapidly compared in order to index into a large library of models according to their shape similarity. The second level uses local image feature sets as a representation. A similarity measure derived from fuzzy set theory, the Hausdorff distance measure and a Bayesian based consistency measure is used to recover matches between local image features and ultimately provides a global measure of similarity between images. The third and final level uses attributed relational graph representation and explicit correspondence matching to verify the recognition.

We are going to review the literature on object recognition. We will be focussing more particularly on techniques based on line-patterns. The field of object recognition can be broadly divided in four areas; (a) evidence combination and voting, (b) correspondence matching, (c) graph matching and (d) alignment. The idea behind evidence combination approaches is to build a summary of image features. Examples include the Hough transform (Ballard, 1981; Illingworth and Kittler, 1987; Illingworth and Kittler, 1988; Princen et al., 1992), geometric hashing (Lamdan et al., 1988a; Lamdan et al., 1988b; Rigoutsos and Hummel, 1995; Procter and Illingworth, 1997) and Histogram (Swain and Ballard, 1990; Jain and Vailaya, 1996; Dorai and Jain, 1995; Evans et al., 1993). Correspondence matching aims at associating in-

dividual image features between a data image and its model. Techniques such as moments (Eakins, 1989; Taubin and Cooper, 1991; Niblack et al., 1993), principal component analysis (Turk and Pentland, 1991; Sclaroff and Pentland, 1993), feature indexing (Grimson and Lozano-Perez, 1987; Knoll and Jain, 1986; Ettinger, 1988) and feature sets (Rucklidge, 1995; Bloch, 1996) fall into this category and are not too computationally demanding. Graph matching approaches offer more detailed analysis of the consistency of matches using structural and relational information. Matching based on this enriched representation is more challenging because of the problem of relational inexactness. A number of algorithms have been proposed to perform graph matching. The methods used include probabilistic relaxation (Kittler and Hancock, 1989; Christmas et al., 1995; Wilson and Hancock, 1995), genetic algorithms (Cross and Hancock, 1996; Cross et al., 1996) and structural indexing (Costa and Shapiro, 1995; Messmer and Bunke, 1994). Alignment techniques may be regarded as a form of template matching, where shape deformations are allowed (Ullman, 1979). There are a number of alignment methods; These include direct alignment methods (Huttenlocher and Ullman, 1987; Viola and Wells, 1997) and optimisation based alignment methods (Ullman, 1979; Scott and Longuet-Higgins, 1991; Shapiro and Brady, 1992; Cootes and Taylor, 1995; Sclaroff and Pentland, 1995). The correspondence matching, graph matching and alignment methods require non-negligible computational effort. They are therefore not particularly appropriate to operate in large libraries of models, unless they can be rendered more efficient.

The aim of the research presented here is to achieve large-scale object recognition using relational and attribute representation of line-pattern arrangements. Current object recognition techniques are too computationally intensive to be used for rapid retrieval and recognition where a large number of possible candidate models have to be investigated. Additionally, the rapid recognition methodologies proposed by the content-based retrieval community do not provide the level of accuracy required by most object recognition tasks. We therefore intend to devise a hierarchical frame-

work by combining the rapidity of image database retrieval techniques with the robustness and accuracy of the most effective object recognition approaches. Each of the three levels of the hierarchy proposed, (i.e. summary based, local feature set based, and attributed graph based) present original contribution to the literature.

One of the novel features of our approach is to make use of single and very compact histogram representation for the purpose of rapid recognition. In this way we can rapidly gauge the similarity between any pairs of objects. Thus, making the problem of object recognition from large model libraries an achievable task. Rather than histograming directly from pixel attributes (e.g. grey-levels, RGB-levels or texture descriptions) (Swain and Ballard, 1990; Jain and Vailaya, 1996), Euclidean invariant attributes computed from line-segment pairs are employed. The pairwise geometric histogram encodes the distribution of geometric co-occurrence between image line-segment pairs.

Another important contribution of our work is to encapsulate relational (or structural) constraints within the histogram representation to improve its robustness to noise, clutter and occlusion without loosing any of the invariant properties of the line-patterns.

Additionally, a novel similarity measure for object recognition from the attributed relational representation is presented. The measure commences from a Bayesian consistency criterion, which has been developed for locating correspondence matches between attributed relational graphs using iterative relaxation operations (Wilson and Hancock, 1994). Here we simplify considerably the consistency measure so that it is used in a non-iterative manner. The design of the simplified measure draws inspiration from both the Hausdorff distance (Huttenlocher et al., 1993; Rucklidge, 1995) and a Bayesian based consistency criterion. This reduces the computational overheads and renders the relational consistency measure suitable for object recognition from large modelbases. We have called it the Fuzzy Hausdorff Distance.

Another novel aspect of the approach presented here is its hierarchical organisation. The hierarchy consists of three distinct levels. The lower level consists of the rapid indexation scheme, briefly introduced in an above paragraph. This aims to discard the images or object models that do not have sufficient similarity with the query image when based on the relational pairwise geometric histograms. The intermediate level refines the rather crude model selection provided by the underlying layer. A novel similarity measure for non-iterative attributed graph based object recognition has been devised. The result is an accurately selected set of image models all bearing geometric and structural similarities with a query object.

Having greatly reduced the number of possible model candidates from the line-pattern library, a more computationally intensive object recognition technique may be used to accurately reject false positives and locate true positives. The technique we are using here is directly related to graph matching. Since the very first level of the hierarchy, image line-patterns have been represented using both relational structure and geometrical attributes. The graph matching algorithm we use allows us to recover line to line correspondences between the query image and selected objects from the library of line patterns.

The thesis is written in the following fashion. We will start with a review of the state of the art in object recognition and content based image retrieval from large databases. This review pays particular attention to shape similarity. We then present in Chapter 2 the particular case of object representation based on line-patterns from other people's work and our own. The following Chapter (Chapter 3) presents the lower level of our hierarchical approach; based on a two-dimensional pairwise geometric structural histogram representation. This Chapter includes comparison of various attribute based histograms. Here we investigate the use of relational information as well as a number of alternative histogram distance measures. Chapter 4 presents the Fuzzy Hausdorff Distance measure. In this chapter we give comparative results with the standard and Rucklidge's Haussdorff (Rucklidge, 1995). We

also present experiments with a number of relational graph structures as well as a variety possible pairwise attributes (vector based and histogram based). The last level of the hierarchical framework is presented in chapter 5. Finally, we give our conclusion about the work reported here and provide detailed information about possible future work and extensions to the proposed methods.

## 1.2 Object Recognition

Object recognition has been one of the primary goals of computer vision, since its origin as a distinct field of research in the early 1960's. Indeed, for many vision systems or applications, object recognition is the ultimate goal. This probably justifies why it is still a very active area of research. Many very advanced, robust and efficient algorithms have been proposed over the last 30 or so years (Suetens et al., 1992). Due to the complexity of the task at hand and computational requirements of most object recognition algorithms, their use in real world problem is limited to small or very specialised tasks. Our aim is to devise a methodology that would enable object recognition to be performed from very large libraries of models. Posed in this manner the problem bears some obvious similarities with the recent and considerable problem of content-based retrieval in multimedia databases, and more specifically image databases (Niblack et al., 1993; Pentland et al., 1994; Gevers and Smeulders, 1992; Swain, 1993; Picard, 1995; Jain and Vailaya, 1996). It is therefore appropriate to survey both areas of research in order to identify the advantages and weaknesses of alternative approaches.

The methodologies for object recognition found in the literature may be broadly categorised into four groups. A key issue for indexing and retrieval of object in large libraries is to achieve speed while retaining accuracy. We will therefore present each method in order of computational complexity and time requirements. The simplest and least demanding approach is evidence combination. Correspondence

matching methods require increased computational effort over the summary based approaches but most benefit from improved accuracy performances. Finally the very computational demanding graph matching and alignment algorithms will be presented. Increased robustness and accuracy usually reward for the loss of rapidity of those methods.

### 1.2.1 Evidence Combination

The basic idea behind evidence combination or accumulation techniques is to produce a very compact object representation. Some approaches make direct use of image pixel values while others compute feature measurements prior to voting. Despite the compactness of the representation a careful selection of the measurements allows rapid, unambiguous, and robust identification and recognition of objects or images. The main voting methods are listed below.

#### Histograms

Swain and Ballard (Swain and Ballard, 1991) have devoted considerable effort into the study of colour histogram. The colours contained in an image are discretised in the RGB (red-green-blue) space and each bin in the histogram is the count of the number of times each discrete colour occurs. They have shown that such a representation may be used to index into an image database and that the technique is invariant to small changes in translation and rotation (Swain, 1993; Stricker and Swain, 1994). However, this technique is not robust to variations in lighting conditions, although more recent research (Finlayson et al., 1996) indicates that alternative methods can solve this problem. Finally, the FINDIT system (Swain, 1993) relies solely on the colour histogram to perform retrieval. In our opinion, this is not sufficient and may lead to ambiguous retrieval due to the insensitivity of the histograms to pixel permutation.

Jain and Vailaya (Jain and Vailaya, 1996) augmented the colour histogram ap-

proach with a histogram computed from edgel direction to capture the general shape information. This very simple and compact representation is shown to perform well but suffers from inherent scale and rotation dependence. This is a major drawback for any object recognition technique. Additionally, since the representation is directly generated from the detected edge pixels, a non-negligible amount of noise and clutter is likely to affect the representation dramatically. This will lead to a poor performance.

The work of Evans *et al.* (Evans et al., 1993) is aimed at recognising two dimensional object shapes using a two dimensional histogram of pairwise geometric attributes for each line segment found in the image. The attributes used are the relative angle and range of perpendicular distances between all possible segment pairs. The second attribute eliminates the problem of line splitting creating multiple entries during histogram binning. However this attribute does not offer scale invariance. The major disadvantage of this approach is that the number of histograms required in order to represent an object or image is equivalent to the number of detected lines segments. This representation increases the storage space required for every image or object in the database and the time required to compute the similarity between two objects views.

This approach has since been improved by Ashbrook *et al.* (Ashbrook et al., 1995) who introduced a region of interest. This restricts the set of line-segment pairs used for histogram creation. This allows the technique to cope with occlusion and improves the robustness to noise and clutter. However, there is no clear or obvious way to define automatically the size of the region of interest. This may lead to instability of the representation. The second improvement offered by this approach concerns improved scale invariance. The method uses a brute force technique, which employs a separate pairwise geometric histogram representation for each anticipated scale. This is neither very elegant, nor does it entirely solve the scale invariance problem. Moreover, it increases the amount of storage and computation required

during recognition.

The most recent addition to the pairwise geometric histogram approach comes from Di Mauro *et al.* (DiMauro et al., 1996). Instead of using the perpendicular distance attribute, they have modified the algorithm of Ashbrook *et al.* (Ashbrook et al., 1995) to use the radial distance attribute. This modification is made in order to model variable object shapes as well as to provide discrimination between ambiguous line-pattern pairs. However, the radial distance is computed from the-mid point of the base segment to the end points of the line-pairs. This is likely to be subject to uncertainty due to noise and clutter, and will have undesirable effects on the representation. Furthermore, the advantage of the perpendicular distance is that the radial distance does not resolve the line-splitting problems.

Recently Dorai and Jain (Dorai and Jain, 1995; Dorai and Jain, 1997) have used histograms in the context of 3D object recognition. Their approach makes use of a shape index to represent three-dimensional objects. Koenderink and Van Doorn originally proposed the shape index as a mean of graphic surface visualisation (Koenderink and vanDoorn, 1992). The object surface orientations are discretised at every image point in the three dimensional space and the histogram is created by counting the number of occurrence of each surface orientation. Their results show that using histograms computed from range images to represent 3D objects is possible and allows for shape spectra to be compared very effectively.

**The Hough Transform**

The Hough transform (Ballard, 1981) is a method which transforms complex pixel patterns usually from an image into some selected parameter space. A set of transformation parameters is computed from each image feature pixel. This parameter set provides votes in the quantised transformation accumulator. The strength of the approach relies on the fact that valid associations will tend to vote for a consistent transform while mismatches will generate evenly distributed votes in the parameter

space. The usual problem with such an approach is the detection of those peaks in the quantised parameter space, which represent valid matches.

Duda and Hart (Duda and Hart, 1972) have applied this technique to the detection of straight lines. The generalised Hough transform was then proposed by Ballard (Ballard, 1981) for the recognition of arbitrary object shapes from their edge feature in cluttered scenes. For a comprehensive review of the Hough transform, refer to (Illingworth and Kittler, 1988).

However, the Hough transform remains inherently unable to cope with scale invariance, nor rotation invariance. This reduces its effectiveness at rapidly performing real world object recognition tasks.

**Geometric Hashing**

The technique of geometric hashing was introduced by Lamdan, Schwartz and Wolfson (Lamdan et al., 1988b; Lamdan and Wolfson, 1988) as a more rapid mean of shape recognition. The various processing steps of this method are as follows. Characteristic points or other features (line-segments, corners, etc.) are extracted from the original image edge map. Pairs of points are selected and are used to establish a reference in the scaling, translation and rotation process. The remaining characteristic points are then used to address a hash table. Retrieval from the hash table involves comparing data and model hash table entries. A shortcoming of this technique resides in the size of the hash table grid needed to store the shape and effects associated with over and under binning.

Sengupta and Boyer (Sengupta and Boyer, 1995; Sengupta and Boyer, 1993) proposed a hybrid method for model-base organisation. Their method combines surface-based geometric hashing with hierarchical database organisation techniques for fast recognition of graph models. In this work a hash table based on local surface properties is used to prune the hierarchical “tree like” modelbase.

The recent work of Rigoutsos and Hummel (Rigoutsos and Hummel, 1995) aims

at reducing the effect of the binning process associated with the hash table. Their approach replaces the quantisation of hash values and avoids the resulting binning of hash entries. Instead, a Bayesian maximum likelihood framework is used to give increasingly less weight to a hash table entry, as a hashed feature becomes more distant from the hash entry position. This does not however solve the problems associated with the size of the hash table, which needs to be as small as possible for computational reasons.

In an attempt to reduce the noise sensitivity typically found in point-based geometric hashing methods, Procter and Illingworth (Procter and Illingworth, 1997) proposed a method based on connected edge-triples. Despite the resulting improvements in efficiency and recognition accuracy, this approach is likely to be sensitive to line-segmentation errors. The main problem is that the line-segments are likely to become disconnected and reduce the number of basis useable for hashing and ultimately recognition. This problem is in some respect similar to the connectivity issue found in Stein and Medioni's (Stein and Medioni, 1990; Stein and Medioni, 1992) supersegment approach (see section 1.2.3).

### 1.2.2 Feature Based Recognition

The methodologies presented in this section have the common property of using vectors or sets of features in order to represent the content of digital images. These representations have various degrees of compactness and suitability for operating on real world data under computational time constraints. The idea behind the statistical techniques presented here, such as moments (Taubin and Cooper, 1991; Prokop and Reeves, 1992) and principal component analysis (Turk and Pentland, 1991; Taubin and Cooper, 1991; Sclaroff and Pentland, 1995; Murase and Nayar, 1995), is to compute a salient image descriptor from image measurements or directly on the pixel data in the form of vectors or N-tuples thanks to statistical analysis.

**Moments**

Moments can be used as scale and orientation invariant shape representations (Taubin and Cooper, 1991; Prokop and Reeves, 1992). A vector of moments (each of a different order) is computed directly from the binary pixel image of a shape. The computed moments can be used for representation and to perform translation, rotation and scale invariant recognition. Moments provide a very compact image representation. However since higher-order moments are computed from the shape centroid (first order moment) this representation is sensitive to noise and occlusion. Such a representation requires a unique closed contour to be extracted. It is therefore likely to be sensitive to thresholding or edge detection. Additionally, it is not possible to represent and therefore recognise multiple part objects. This is likely to be undesirable in many object recognition situations.

**Principal Components**

Principal component analysis aims at reducing the dimensionality of a problem by computing the salient vectors (eigenvectors) for a given data set. In communication theory, it is known as the Karhunen-Loeve transform. The aim is to find a set of $m$ orthogonal vectors in data space that account for as much as possible of the data's variance. Projecting the data from their original $n$-dimensional space onto the $m$-dimensional subspace spanned by these vectors results in a dimensionality reduction that often retains most of the intrinsic information originally present in the data. The first principal component is taken to be along the direction with the maximum variance. The second principal component is constrained to lie in the subspace perpendicular to the first. Within that subspace, it points in the direction of the maximum variance. Then, the third principal component (if any) is taken in the maximum variance direction in the subspace perpendicular to the first two, and so on. In the context of computer vision, an image (data set) is represented as a single point in the multi-dimensional eigen space. Nearby points in the eigen space belong

to images (or image description) that are closely related.

A lot of the computer vision literature, involving principal component analysis (PCA) is related to the problem of face recognition. The work of Turk and Pentland (Turk and Pentland, 1991) is probably the most well known. They used PCA to extract the salient features directly from the grey-level pixel of face images. The technique is often regarded as the best for face recognition. However, its robustness to rotation, illumination and orientation is limited. Furthermore, computing the eigenvectors and eigenvalues directly from image pixels is computationally expensive.

The two-dimensional shape representation problem has also been tackled using eigen analysis. Taubin and Cooper (Taubin and Cooper, 1991) used PCA performed on the scaled pixel maps of the object boundaries as one of the shape representations within the QBIC system (Niblack et al., 1993). A few years later, Sclaroff and Pentland (Sclaroff and Pentland, 1993; Sclaroff and Pentland, 1995) described object shapes in terms of generalised symmetries to which they refer as object eigenmodes in the Photobook project (Pentland et al., 1994). Both approaches are based on binary image representation (shape mask) or measurements extracted from them. This provides invariance to illumination, two-dimensional rotation, and three-dimensional orientation.

Recently Murase and Nayar (Murase and Nayar, 1995) used a combination of two eigen spaces for three dimensional object recognition from two dimensional images. The first eigen space is computed from multiple 2D views of objects (one eigen space per object). The second representation is based on all the views of all the objects within the database. Because of this double representation their approach performs both object recognition and pose estimation. Besides being computational expensive, this method has limited robustness to lighting conditions since the eigen analysis is performed directly on image pixels.

### Other Statistical Approaches

In some cases, the images under consideration are not particularly suited for segmentation. In others, the contour of the object does not contain salient information that is important for recognition or classification. In such a situation the photometric properties of the image can be used.

The work of Nayar and Bolle (Nayar and Bolle, 1993) is concerned with the fact that the illumination and orientation properties of a smoothly curved surface are almost similar for close image intensity measurements. The technique is based on a photometric invariant called the region reflectance ratio. The ratio at each image point is compared with its neighbours and used to segment the image into regions of similar reflectance ratio. The relative position of the region centroid and the reflectance ratio of the segmented region may then be used for object recognition and pose estimation of three-dimensional objects.

The work of Van Gool et al. (Gool et al., 1996) is also concerned, but not solely, with photometric invariants. Their recent work combines geometric shape features and intensity features to generate simple and robust moment invariants. The use of both geometric and photometric moments allows for improved invariance to transformations.

### Feature Sets

Bloch (Bloch, 1996) has applied fuzzy set theory to the problem of object recognition. The idea is to use fuzzy operators such as max and min to measure the amount of overlap between the members of unordered sets of image features. The major benefit of using fuzzy operators is to allow for “soft” decisions to be made about the feature correspondences, which are crucial to any method that is to perform on non-synthetic data.

The Hausdorff distance was first used by Huttenlocher *et al.* (Huttenlocher et al., 1993) to locate objects, which have undergone translation. This method was then

further investigated and enhanced by Rucklidge *et al.* (Huttenlocher et al., 1993) in order to cope with the translation and scale of objects in the scene. The work of Rucklidge (Huttenlocher and Rucklidge, 1993; Rucklidge, 1995) presents a method to recognise and locate objects subjected to affine transformation based on feature point sets. Feature points, such as corners are extracted and it is their locations that are used during the process of finding correspondences. Because of the large number of possible associations between the transformed model and the image, a method for efficiently searching the space of transformations of the model is required. When operated on two point sets, representing image feature locations, the Hausdorff distance is a shape-comparison metric. Rucklidge (Rucklidge, 1995) showed the robustness of this shape metric to point position errors, extra points (due to the presence of other objects in the scene) and missing points (because of possible occlusion). Algorithms based on the Hausdorff distance, unlike correspondence-based matching and alignment techniques (Ayache and Faugeras, 1986; Huttenlocher and Ullman, 1990; Olson, 1994), do not create a one to one correspondence between model and image feature sets. This makes the Hausdorff based recognition metric a very efficient technique even when the number of points in each image becomes large.

**Feature Indexing**

The work of Grimson and Lozano-Perez (Grimson and Lozano-Perez, 1984) is concerned with feature indexing. Features are extracted from three dimensional object data (obtained from a triangulation range sensor). The feature type and any unary constraint associated with it are used as an index. The database is organised to allow efficient access to a model containing particular features. Knoll and Jain (Knoll and Jain, 1986) have proposed a recognition method based on feature indexing. The method takes advantages of the similarities and differences between object types. It is able to handle cases where there are a large number of possible object types in sub-linear computation time.

More recently, Ettinger (Ettinger, 1988) proposed a model-based vision system that exploits hierarchies of both object structure and object scale. The approach taken is to develop an object shape representation that incorporates a component sub-part hierarchy. This allows for efficient and correct indexing as well as for relative parameterisation among sub-parts. It also has a scale hierarchy, to allow for a general to specific recognition procedure (from gross to fine feature).

### 1.2.3 Graph Matching

The term graph matching refers to the process of comparing two (or more) graphs with each other. In the context of computer vision the graph structure represents features and their relation in the visual data. Graphs are high level representation and have been shown to provide very compact representation for object recognition and localisation. The seminal paper of Barrow and Popplestone (Barrow and Popplestone, 1971) established relational graphs as a practical representation for scene matching. The subsequent paper of Barrow and Burstall (Barrow and Burstall, 1976) presented some techniques for structural graph matching based upon the idea of searching for maximal cliques in the association graph. However these early methods are only effective when exact relational descriptions are under consideration. To be effective relational matching must be capable of accommodating both errors from ambiguities in object appearance (initialisation errors) and errors from poor image segmentation or presence of noise and clutter (structural errors).

#### Relational Indexing and Hashing

The underlying idea behind relational indexing (or relational hashing) methods is to use the structural information provided by the features extracted from the image to produce an alternative representation. The resulting representation is intended to provide improvements in term of computational requirement and speed during the matching process when compared with other graph matching algorithms.

The common aspect of structural hashing techniques is the fact that shapes are represented as polygonal approximations of their boundary.

For instance, Stein and Medioni (Stein and Medioni, 1990; Stein and Medioni, 1992) extend the polygon approximation description by computing and dividing outlines into super-segments. A super-segment is a grouping of adjacent segments. Each super-segment contains information about the number of segments of which it consists, the sum of the segment length, a list of the angles between successive segments, its location, its orientation and finally its eccentricity. This description is then encoded using a gray-code (Hamming, 1980). The encoded descriptions are then entered in a hash table for efficient storage and retrieval.

Henikoff and Shapiro (Henikoff and Shapiro, 1990) have proposed a method based on segment triple grouping. Each triple is then encoded according to its type. The encoded triples are then used to address a hash table.

Costa and Shapiro (Costa and Shapiro, 1995) proposed a technique for efficiently indexing relational structures. Each model view is represented as a full relational graph. The graph is then decomposed into $n$ sub-graphs of size $s$. Each subgraph is encoded using a table and then used to index into a hash table in order to map input sub-graphs to model sub-graphs. The technique is limited to a small number of subgraph "codes" in order to keep the size of the hash table as well as the computational requirements low.

The main weakness of relational hashing techniques is the non-uniqueness of the polygonal approximation of a curve, which depends on the line fitting tolerance. Therefore, for the purpose of robustness, several polygonal approximations have to be used in order to represent a single object shape.

**Relational Graph Matching**

A diverse family of algorithms has been proposed over the years to solve the problem of finding error tolerant sub-graph isomorphisms. The best known algorithm

for sub-graph isomorphism is that of Ullman (Ullman, 1976). The technique is based on tree search with backtracking. It has been improved by the use of lookahead for speedup. Shapiro and Haralick (Shapiro and Haralick, 1985) proposed a general relational distance measure, that computes a numeric distance between any two symbolic structural representation in order to gauge relational consistency.

Other algorithms have been proposed with the aim of meeting the dual goal of gauging consistency and overcoming structural errors, based on simulated annealing (Herault et al., 1990), neural networks (Feng et al., 1994), genetic algorithms (Jong and Spears, 1989), continuous optimisation (Kuner and Ueberreiter, 1988) and probabilistic relaxation (Kittler et al., 1992; Wilson and Hancock, 1994). The common feature of these methods is to iteratively minimise an objective function that represents the distance of the current solution to the correct solution.

Messmer and Bunke (Bunke and Messmer, 1995; Messmer and Bunke, 1994) have introduced an exact and error-tolerant graph matching algorithm based on a compact representation of the graphs. Both the input graph and model graph to be compared are pre-processed so that sub-graphs appearing multiple times are represented only once by a "meta-node". The computational effort associated with graph matching algorithms in general is therefore greatly reduced.

### 1.2.4 Alignment

Roughly speaking, the problem of alignment may be seen as a template matching exercise. The basic idea is to use groups of features or templates and to exploit their geometric properties to find correspondences in a target scene (or image). The main difficulty lies in the estimation of the transformation geometry. In other words, the aim is to recover the transformation between image and model co-ordinate systems. A prerequisite for this estimation process is the correct labelling of correspondence matches between extracted features in both images. There is therefore a chicken-and-egg situation to overcome. Before transformation parameters can be estimated

good correspondences need to be available, yet these correspondences need rough transformation parameters. A number of approaches have been proposed to resolve this problem. These consist of placing constraints on the one to one correspondences (Cross and Hancock, 1998a) or the possible transformation, or by employing optimisation methods and the Gaussian weighted proximity matrix (Scott and Longuet-Higgins, 1991; Shapiro and Brady, 1995; Cootes and Taylor, 1995; Sclaroff and Pentland, 1995).

**Direct Alignment Methods**

Huttenlocher and Ullman (Huttenlocher and Ullman, 1987; Huttenlocher and Ullman, 1990) were the first to propose a method where the grouping of nearly connected features forms alignment features. The alignment features include combinations of end-points of segments, points of high curvature and virtual points induced by the intersection of straight segments. For recognition each alignment feature from the image is matched against each possible model alignment feature. Once an alignment between a triple of data features and model features is found a score is kept about how well the remaining features in the model and the data map onto one another. This is repeated for every possible combination of data and model feature triple. The best alignment is the one that obtains the best score. To reduce the number of alignment operation, feature labels are employed. However, it is clear that noise in the image will create mislabelled features. This will lead to increased computational requirements to perform the alignment. Moreover, in extreme cases the alignment operation may fail.

Viola and Well (Viola and Wells, 1997) presented a technique for aligning 3D object models to a real image scene under occlusion and including clutter. Their approach is based on image intensities rather than features and requires no *a priori* model of the relationship between the 3D model and image scene intensities. The idea is to perform alignment by maximising the mutual information between the

model about the scene. The mutual information maxima are found by computing the derivative of the entropy with respect to the transformation between the data and the model to align. The local maximum of the mutual information is obtained by using a stochastic analogue of gradient descent.

**Optimisation Based Alignment**

Ullman (Ullman, 1979) identified the importance of exploiting proximity constraints during the one to one correspondence matching process. Since then, a number of approaches have taken Ullman's idea on point sets matching further. These methods aim to develop general purpose correspondence matching algorithms using the Gaussian weighted proximity matrix.

The work of Scott and Longuet-Higgins (Scott and Longuet-Higgins, 1991) is based on computing the singular value decomposition of the inter-image proximity matrix in order to determine feature point correspondences. Their approach contrasts with the work of Shapiro and Brady (Shapiro and Brady, 1992; Shapiro and Brady, 1995), who effected the matching process by comparing the modal eigenstructure of the intra-image proximity matrix. It is from these techniques that the deformable shape models of Cootes *et al.* (Cootes and Taylor, 1995) and Sclaroff and Pentland (Sclaroff and Pentland, 1995) have been inspired and devised. The major shortcoming of the Point Distribution Model (PDM) approach is the necessity to learn the modes of variation of each shape from landmark (or characteristic) points from a number of examples. Furthermore, an initial estimate of the pose and scale of the object is required. For this reason the PDM approach is more suitable for object location and model fitting rather than object recognition.

The computational complexity of alignment techniques makes them highly unsuitable for working on a large library of models. Finding all the possible correspondences between data image features and each of the models in the model-base is clearly a not viable solution. It makes a lot more sense to devise a method to

automatically select the most appropriate models from the library and perform the alignment process on those selected models only for the purpose of hypothesis verification.

## 1.3 Image Representation

Both the problems of retrieving and recognising object images from a large library based on their similarity to some reference image implies that some features can be automatically (or semi-automatically) extracted from the digital images. Most of the recent image database systems make use of three distinct information sources contained in the image pixel map. These are colour, texture and shape. Object recognition is mainly concerned with the shape aspect of the image content. The methods for representing and comparing colour properties in images are solely statistical. Texture attributes can be dealt with using either statistical (general textures) or structural (man-made patterns) techniques. Characterising shapes is not quite as straightforward. There is no clear manner for mathematically describing general object shapes (Mumford, 1991). Various approaches for solving this problem have been proposed. They are based on voting, feature vectors, structural or relational matching and alignment methods. We have reviewed these techniques in section 1.2.

- Colour information

  Image colour information, often encoded within a multidimensional histogram (Swain, 1993), is used to retrieve image from the database with similar colour distribution. The histogram offers interesting properties such as invariance to rotation and translation, as well as perturbation by small changes in the viewing angle, scale and occlusion. However, because of their direct dependence on the image pixel values, changes in the lighting condition will dramatically affect the histogram properties. In the context of object recognition, the colour histogram representation is not a viable solution. This is because

the spatial information of image pixels is discarded by the histogram.

- Texture information

  The principal characteristic of a texture is a repetition of basic patterns. There are various approaches to texture representation, which are either based on statistical or structural methods. The latter are less suitable for general texture indexing problem because of their inability to cope with natural textures. Statistical methods are numerous and vary in complexity and efficiency. Among the simple techniques are first order gray-level statistics based on the gray-level histograms, second order gray-level statistics and eigen analysis. More complex techniques involve the power spectrum of the texture image, which is based on its Fourier transform. Angular and radial bins in the Fourier domain capture the directionality and the frequency (rapidity of fluctuation) of an image texture. A detailed review on the subject may be found in a recent survey by Reed and DuBuf (Reed and DuBuf, 1993).

- Shape information

  Many methods for extracting shape information from images have been developed. Statistical, Geometrical and Structural recognition approaches have been investigated. Further to the method used is the underlying feature upon which these techniques are applied. The shape feature can be computed from either object boundaries (or their feature of interest) or description of the region occupied by the object in the image plane (or associated features).

  Among the possible feature of interest the most commonly used image attributes for shape representation are edge pixels (Jain and Vailaya, 1996; Rucklidge, 1995), corners (Wang and Brady, 1995), object boundaries (Sclaroff and Pentland, 1995; Farzin Mokhtarian and Kittler, 1996), straight line-segments (Evans et al., 1993; Stein and Medioni, 1990; Huttenlocher and Ullman, 1987; Christmas et al., 1995), geometric invariants (Rothwell et al.,

1992; Lamdan et al., 1988a) or surface curvature (Murase and Nayar, 1995; Dorai and Jain, 1995; Ashbrook et al., 1995).

Section 1.4 reviews many of those shape representation and aims at exposing the weaknesses and advantages of the most commonly used.

The choice of the low level feature representation used by any recognition or retrieval process is very important. If the wrong selection of feature or feature attribute is made, the approach may not be able to reach a correct conclusion. While selecting the optimum shape representation one has to ensure the four following criteria are satisfied (Grimson, 1990).

- **Scope and Sensitivity**: The representation used by the system should be able to cope with all the relevant shapes that the image database will contain. Furthermore, the representation used should be sensitive enough to preserve distinction between shapes.

- **Uniqueness**: There should be a unique mapping between a shape and its representation. If this criterion is met, identical shapes will have identical descriptions. In this way the problem of comparing shapes is simplified.

- **Stability**: The shape representation should not be dramatically affected by small changes in the shape features extracted from the raw image. It should tolerate reasonable amounts of noise and occlusion in the scene.

- **Efficiency**: The computation of the shape descriptors for the input data should be rapid. Similarly the representation should allow effective and straightforward shape comparison properties. The system should not have to preprocess the stored shape description in order to compute the similarity between shapes.

Similarly the retrieval and recognition techniques employed by the system must obey to the following requirements.

- **Multiple Object Scene**: Extra features that may belong to other objects or result from noise in the original image should not affect the techniques used to retrieve images from the database. The recognition process or retrieval should therefore allow grouping of features.

- **Occlusion:** The methodology used should not be dramatically affected by missing feature. The features extraction algorithm may not always provide a perfect output, especially when dealing with real data. The techniques should allow for a reasonable amount of noise, clutter and occluded features.The recognition process should be performed through local features.

- **Invariance under geometrical transformation**: Object recognition application and similarly image retrieval system may only be used on real world problem if they are able to abstract the geometrical positioning of the object under consideration within their representation. A representation that does not allow direct invariance to translation, rotation and scaling dramatically limits the effectiveness of the algorithm and increases greatly the computational complexity. The recognition process should be based on an invariant representation.

- **Efficiency**: The image database retrieval or object recognition engine should allow rapid, but not necessarily real-time, delivery of information. Linear search of the database content may be sufficient for a small library of objects and if the object description allows very efficient comparison performances. However as the number of object models in the collection increases, the need for efficient indexing techniques arises.

The above lists give us the representation and retrieval requirements to consider when developing recognition methods. We now review and discuss the possible representation alternatives.

## 1.4 Shape Representation

The problem of representing shapes is not a trivial one. There is no clear winner as far as the various techniques proposed in the object recognition and image database work. In this section we review some of techniques proposed for representing salient image information. Some representations appear to provide more robustness to noise and clutter in the image than others. Moreover, some allow for invariant shape properties to be encapsulated within the image description. This will reduce greatly the computational requirement during matching.

**Edge Pixels**

Jain and Valaya (Jain and Vailaya, 1996) suggested the direct use of edge pixel orientation information to construct a shape histogram. The technique is used for comparing trademarks and logos. Similarly, the Hausdorff distance based object recognition method proposed by Rucklidge (Rucklidge, 1995) makes direct use of edge pixels during the matching. There are many drawbacks of making direct use of edge pixels within an algorithm. Edge detection algorithms are sensitive to noise in the original image. This may create a large number of unwanted edge pixels and therefore increase the complexity of the algorithms. Additionally, using such raw features does not help in solving the problem of translation, rotation and scale invariance. It would be more effective to use higher level features computed from the edge pixel map. This would reduce the amount of data to be dealt with along with providing better invariant attributes.

**Closed-Contours or Boundaries**

The first representation based on object contours is the chain code (Duda and Hart, 1973). The representation is a circular list that consists of the either four or eight connected paths of edge pixels detected in the image (Gonzalez and Wintz, 1977).

This representation is translation invariant. The major drawback is that a chain code representation does not provide rotation and scale invariance. Additionally, in the presence of noise connected pixels may become disconnected, leading to increased computational requirements during matching because of the increased number of chains to be processed.

Mokhtarian and Kittler (Farzin Mokhtarian and Kittler, 1996) uses the curvature of the object contour between control points carefully positioned in order to create a Curvature Scale Space representation. The curvature scale space is a multi-scale organisation of the curvature zero-crossing points of the contour as it evolves. Multiple scales are introduced by applying a Gaussian filter of increasing standard deviation to smooth the contours from which the curvature zero-crossing points are recovered. Partial occlusion of the object boundary may dramatically change the representation up to a point where it is not matchable against the non-occluded contour.

The recent work of Ogniewicz and Kubler (Ogniewicz and Kubler, 1995) involves creating hierarchical skeletons of planar shapes. The skeletons are created from object outlines and offer an effective transformation of two-dimensional shapes onto a linear structure. The technique copes with noisy contour and rotation. However the skeleton may be quite dramatically affected by occlusion. In addition, this approach does not offer the possibility to work with non-contour based shapes.

Kupeev and Wolfson (Kupeev and Wolfson, 1994) have presented an algorithm for the detection of perceptual similarity among planar shapes. Their approach represents shapes using weighted graphs. The vertices of the graph represent the ”lumps” of the shape in a given orientation. However, this representation is not rotation invariant. The graph structure is affected by changes in orientation. Therefore multiple graphs, one for each shape orientation, have to be constructed. This leads to increased computational effort. Furthermore, small change on the shape boundary may lead to extra and missing ”lumps” and affect the structure of the representation.

The shock graph representation of Zucker *et al.* (Siddiqi et al., 1998) is related

to both skeletons and graphs. The basic idea is to derive a structural description of a shape from the shocks of a curve's evolution process acting on its bounding contour. The most significant shape component is the root of the graph. The graph may therefore be re-written as a tree, which simplifies the matching process. The shocks are labelled into four predefined types according to the local variation of the radius function along the median axis. Again, perturbation on the contour will affect both the structure of the graph and the label of its nodes.

The use of object contours is a popular choice and may be effected via spline fitting, skeletonisation or polygonisation of the edge pixel boundary. However, the contour may not be entirely detected by the low level feature extraction algorithm and therefore the contour may be broken up. A similar situation will arise when parts of an object are occluded. For these reasons we believe that limiting the methodology for object recognition to contours may reduce its ability to perform in real world situations. It is also for these reasons that we favour the use of a straight line-segment based pattern representation for our experiments.

**Straight line-segments**

Using straight line-segments as low level features is probably the most common for the object recognition algorithm. A wide range of algorithms have been proposed based on linear features (Arkin et al., 1991; Christmas et al., 1995; Stein and Medioni, 1990; Thacker et al., 1995). Some approaches opt for direct use of the segment position and orientation (Arkin et al., 1991). Others are based on relative measurements (Kittler et al., 1992; Evans et al., 1993; Stein and Medioni, 1992; Costa and Shapiro, 1995). The use of relative attributes offers a number of advantages. The most interesting advantage is the invariant attributes that relative measurements provide. The most common of these pairwise attributes is the relative angle between a pair of line-segment (Kittler et al., 1992; Evans et al., 1993; Stein and Medioni, 1992). The relative angle is translation, rotation and scale invariant; whereas using single

line-segment orientation does not provide rotation invariance. The other advantage is that most shapes can be simplified to a polygonal approximation, which is nothing more than a collection of possibly connected straight line-segments.

A more detailed description of straight line-segment attributes and their use in fast object recognition algorithms can be found in Chapter 2 Section 2.1.

**Geometric Invariants**

Considerable efforts have been spent on identifying a robust invariant image representation. It is however not possible to compute invariant measurements from three-dimensional objects under perspective projection onto the image plane.

A simple projective invariant called the cross ratio was first introduced in the computer vision literature by Duda and Hart (Duda and Hart, 1973). The cross ratio of four collinear points, although very simple, has the property of being constant regardless of the viewpoint. The work of Maybank (Maybank, 1995; Maybank, 1998) demonstrated the effectiveness of the cross ratio for model-based object recognition. However, the cross ratio is not particularly suitable to non polyhedral object recognition, since accurately locating a set of four collinear points may prove to be difficult.

Other, more general geometric invariants have been proposed for overcoming this problem. Rothwell, Zisserman, Forsyth and Mundy (Rothwell et al., 1992) extended the canonical frame construction for planar object recognition proposed by Lamdan et al. (Lamdan et al., 1988a). The method is based on the extraction of points of interest from the object boundary curve (such as concavities and convexities). Four distinct points are localised for each concavity (or convexity) using two bitangency points to mark the entrance and the two contact points between the contour and its tangent passing by one of points of entrance. These four points are mapped into a reference unit square and the resulting transformation is used to project the curve into the canonical frame. An object shape may be uniquely represented by

its set of canonical frames (describing semi-local invariants). However, computing the tangent and therefore point of interest to object curves is not particularly robust to changes in the edge map generated from the image (noise, occlusion, etc.). This is likely to severely affect the robustness of the representation and significantly decrease the level of performance.

**Surface curvature**

The use of surface curvature indicates that the algorithm will be dealing with three-dimensional objects. Indeed, the work of Dorai and Jain (Dorai and Jain, 1995), Murase and Nayar (Murase and Nayar, 1995) and others (Worthington et al., 1998; Ashbrook et al., 1998) showed how effective the use of surface orientation is for view-based object recognition. A number of approaches have been proposed to compute and represent object's surfaces such as, shape from shading, shape from texture, reflectance ratio, triangulated meshes and range imagery. The major problem with these approaches, is the computation of the surface attributes at every image point under non controlled illumination condition.

## 1.5 Image Database Systems

In this section, we describe the methods currently used by content based retrieval systems and discuss how appropriate they are for object recognition from large library of models.

The content-based representation and retrieval of visual digital data from large collection is considered a very daunting computer vision task. Although it is always possible for humans to painlessly organise this kind of data by brute clerical force, it seems that a more elegant and cost effective self-organising recognition methods would be welcomed (Grosky and Mehrota, 1989; Gevers and Smeulders, 1992; Niblack et al., 1993; Swain, 1993; Pentland et al., 1994; Picard, 1995; Farzin Mokhtar-

ian and Kittler, 1996; Gimelfarb and Jain, 1996; Jain and Vailaya, 1996). Our interest, here, lies in the area of image content analysis for object recognition, which is commonly referred to as shape based.

Recently content-based retrieval techniques have been proposed (Niblack et al., 1993; Swain, 1993; Pentland et al., 1994; Picard, 1995; Eakins et al., 1996; Gimelfarb and Jain, 1996; Jain and Vailaya, 1996). Users are able to specify query by means of examples such as sketches, photos, images or icons. The notion of similarity for two images or two visual items remains a major issue (Scassellati et al., 1994; Santini and Jain, 1995). It is common for us humans to categorise similarly two objects looking rather different in terms of appearances. As far as the work reported here is concerned similarity is defined solely in terms of shape or visual appearance.

In the early days of image database research most approach had little to do with computer vision or image processing and more to do with standard databases. Queries were based on keywords, which were then used to search the textual index of the database for image. While the early types of systems concentrated on one aspect of image retrieval and were mainly based on textual (keyword) description, the new generation incorporates and extends multiple kinds of query. We will now review some of the latest development in intelligent image databases. For a more comprehensive review of the state of the art in content based retrieval from image database, refer to (Picard and Pentland, 1996; Marsicoi et al., 1997).

QBIC (Query By Image Content) (Niblack et al., 1993) was among the first research efforts in content based retrieval. It allows images to be retrieved by a variety of image content descriptors including colour, texture, and shape. The image object outlines are obtained using an interactive outlining method. The pixel values within the area defined by the outline are used to compute the colour features (mean colour, histogram, etc.), the texture features (coarseness, contrast and directionality). The outline itself is used to compute the shape features (circularity, moments, principal component analysis of binary shape images, etc.). Users may use a graphical

user interface to specify their search criterion based on colour, texture, shape (sketch or image example) or combination of the latter. It is clear that the major drawback of the QBIC approach is the interactive outlining, which requires every image in the database to have all its objects manually located and delineated by hand before they can be retrieved. Additionally, QBIC's underlying representation of shapes does not allow for multiple part objects to be dealt with. Similarly, the robustness or the representation to occlusion is only very limited since closed object outlines are the basis for the computation of shape descriptors such as moments, which as described in Section 1.2.2 are very sensitive to changes of the shape centroid.

Photobook (Pentland et al., 1994) is a set of interactive tools for browsing and searching images and image sequences. Direct search on image content is made possible by using of semantic-preserving image compression, reducing images to some perceptually significant coefficients. Users are offered three different types of search; based on grey-level appearance (using eigenfaces (Turk and Pentland, 1991)), using two dimensional shapes (based on Finite Element Method (FEM) performed on shape feature point locations (Sclaroff and Pentland, 1993)), and based on textural properties (using 2D Wold-like decomposition (Picard and Liu, 1994). The Photobook system appears more like a collection of algorithms rather than an integrated content-based retrieval system.

FourEyes (Picard, 1995) aims at combining low-level and high-level vision, as well as interactive input from a human user. The solution is based on learning from the user what is important visually as well as learning associations between text descriptions and visual data. FourEyes appears to try to solve the major drawback of Photobook (disconnected methodologies) by computing various feature representations for each image or object. The selection of the set of features that best represent the data of interest to the user is determined using an online learning process based on user interaction and a self organising map.

SQUID (Shape Queries Using Image Databases) (Farzin Mokhtarian and Kittler,

1996), is a shape based retrieval system. Boundary contours of objects are recovered automatically from a database of 1100 marine creatures pictured on a uniform background. Each contour is represented by a number of global shape parameters and the maxima of the curvature zero-crossing contours in its curvature scale space image, as described earlier (see Section 1.4). The main drawback of this approach, although very effective on clean object contours, is its sensitivity to occlusion. Indeed, global shape parameters, as we discussed in section 1.2.2 do not allow for missing or extra parts on the shape contour. Similarly, as the size of the occluded area increases the Curvature Scale Space image will be dramatically changed.

Jain and Valaya (Jain and Vailaya, 1996) proposed a very simple methodology for retrieval of logos and trademarks from a large database. The technique is based on colour and shape histograms. Three, one dimensional histograms are used to represent the colour distribution, for each of the colour bands (RGB) and another one dimensional histogram of significant edge directions is used to represent the shape information contained in an image. The major drawback of this approach, although it was shown to be very effective, is that in order to cope with rotation invariance a shift of the histogram bin has to be performed during matching. This leads to a dramatic increase of the number of histogram comparison to be computed during the retrieval process.

Unlike object recognition, the result of an image database query is not necessarily error-free, the generated output is an ordered set of images where the first items match more closely the query than the later ones. This suggests that image database approaches would be well suited to the task of rapidly pruning. In other words, those object models baring no similarity with the object presented as query would be removed. Therefore, its seems appropriate to investigate a methodology for retrieval and recognition from large libraries based on a hierarchy of processing levels. In such a framework, each level becomes more selective, by using a more refined but more computationally intensive algorithm thus leading quickly to accurate

solutions.

## 1.6 Critical Discussion

The review of object recognition techniques presented in the previous sections indicates that unlike colour and texture, shape information may be characterised using statistical, geometrical or structural methods. Object recognition algorithms, either model-based or view-based, are all concerned with the shape characteristics of objects.

The major drawback of the statistical techniques lies in selecting the right number of attributes in order to satisfy the "scope and sensitivity" requirement. As an example, one may consider the PCA approach and ask how many eigen vectors and values should be used. Choosing a small number of principal components may not allow discrimination and encoding of all the necessary shapes for a given problem. Voting techniques such as histograming and hashing suffer from a similar problem.

Structural methods suffer from the related drawback of "uniqueness". Indeed, the features (corner, line-segments, etc.) used are directly dependent on the system designer. Unlike for statistical approaches, there is no automatic salient feature extraction. The number of attributes used by a structural technique directly affects the uniqueness of the representation for every non identical shape in the database. If the number of attributes becomes too low the "scope and sensitivity" requirement may also be affected.

As far as geometrical approaches are concerned the difficulties are mainly associated with robustly extracting and grouping the features. In the case of geometric hashing (Lamdan et al., 1988a; Lamdan et al., 1988b; Rigoutsos and Hummel, 1995), the performance of the technique is directly linked to the accurate location and extraction of points of interest. Clearly, if the pair of points used for reference is not found in the original image, or if their location is not correct, the remaining char-

acteristic points will not fall in their original hash table location. This makes the representation “unstable”. Similarly the four points of interest used by Rothwell *et al.* (Rothwell et al., 1992) need to be accurately extracted and located in order to compute a stable representation via canonical frames.

One of the nice aspects of statistical methods for representing shapes is that these techniques are able to directly map the original image to their representation. For example, one can perform principal component analysis directly on the colour, grey level or binary images. This is the approach taken by Turk and Pentland (Turk and Pentland, 1991) for performing face recognition (using grey level images of human faces) and by Taubin and Cooper (Taubin and Cooper, 1991) for shape recognition (using thresholded scaled object shape images). However, computing the principal components directly on the pixel data, does not allow us to easily cope with changes in scale, viewing position, and illumination condition. It is therefore, more appropriate to extract characteristic features from the image first and then perform PCA on the feature data. The work of Cootes *et al.* (Cootes and Taylor, 1995) on Point Deformable Models is a good example of such an approach. However, the drawback is its reliance on the automatic detection of these characteristic features and the construction of the object models.

The main advantage of geometrical approaches for shape representation is based on the existence of invariants. Global invariant, such as moments, can easily be computed. However, they are not robust to occlusion and may be affected by noise or clutter in the image. Fortunately, semi-local invariants allow geometrical techniques to deal with occlusion. In the general sense there are a very limited number of invariant (collinear features, concentric features). When considering planar shapes, parallelism and curvature may be used as invariant features. The work of Rothwell *et al.* (Rothwell et al., 1992) provides an effective use of such planar geometric characteristics by representing local object shape concavities using canonical frames.

Structural methods have the advantage of providing a very compact repre-

sentation. All the techniques reviewed above such as for example, the weighted graph (Kupeev and Wolfson, 1994), structural hashing (Stein and Medioni, 1992; Henikoff and Shapiro, 1990) and skeletons, are able to encode object shapes in a very limited number of attributes when compared with geometric approaches.

In our opinion, the way forward as far as representation is concerned, is to combine statistical, geometrical and structural approaches. The approach we propose is one where geometric and structural attributes related to line-segments are used. These structurally gated geometric measurements are encoded in a histogram data structure called the pairwise geometric attribute histogram. This is used as a low-level shape representation in our work. Similarity between images (in terms of shape) is computed using statistical distance measures between image histograms (see section 3.2.2 for more details).

Given a library or database of models (images or imaged objects) and their corresponding features description or representation, we want to select the models that are likely to match the features found in a visual query. This is a database organisation (or indexing) problem. The goal here is to do much better than simply trying each model in turn. There are various solutions proposed in the literature for organising the database as efficiently as possible to allow rapid selection of the best model match. For example, clustering the models within the database will allow efficient organisation of the search process.

The indexing technique allows ordering of the information within the database according to some criterion (feature type, feature size, etc.). However the index of a database is still a “linear” structure. In the worst case one may have to scan through the whole of this index. Clustering offers a major advantage to indexing in this respect. It allows the database to be represented as a “tree like” structure. In the worst case the search becomes now directly dependent on the depth of the database tree.

Although the actual work will not in its current form make use of indexing or

clustering techniques there is still a need for pruning the vast quantity of irrelevant image models from the database. This is required to enable more refined object recognition techniques to be employed. Indeed, the most powerful object recognition approaches are also the most computationally demanding. It is therefore inappropriate to use traditional object recognition techniques on a large-scale model library of real world image data. We have also established that image retrieval approaches are able to cope with massive numbers of images thanks to content-based compression. However, the level of accuracy of such methods is not usually appropriate for object recognition tasks.

## 1.7 Summary

This review of the literature for object recognition suggests that none of the currently available algorithms is able to achieve rapid retrieval from large model libraries. A close look at the field of content-based retrieval from large image databases indicates that although good progress has been made, the problem of indexing according to shape similarity is still far from achieving level of performance comparable to colour and texture indexing. The speed of shape retrieval algorithms is also an important factor for large-scale object recognition.

The integration of ideas from both fields appears to enable some of the current problems to be solved or effectively tackled. It is clear that the representation needs to be very compact, although still easily and effectively compared with other object representations. It appears that more work needs to be done with higher level structural primitives in combination with lower level geometrical attributes. Furthermore, a hierarchical framework of processing levels needs to be investigated. The low level processing algorithms should be closely related to image retrieval algorithms. Moreover, the system should aim at pruning from the model library all unrelated entries. The higher level algorithms should be closely focused on object

recognition. In addition, it would be a great advantage to use the same representation throughout all the computing steps of the system.

# Chapter 2

# Line Patterns

One of the most important parts of any computer vision process is the representation. This is commonly based on image measurements or image feature measurements. The selection of measurements (or attributes) is crucial to the level of performance of the recognition algorithm. In order to accurately represent a large collection of images (or line-patterns) the attributes have to satisfy a number of properties. It is beneficial to identify and use measurements that are invariant to changes in position (translation), orientation, and scale of the objects depicted in the images. In addition, the robustness of the representation to noise caused either by occlusion or simply because of non optimal image capture or low-level processing, is essential if the method is to be used on real world data. For line-segment pattern representation, multiple geometric attributes can be used. There are advantages in using line-segment pairs over single line-segments measurements. When considering invariance of the attributes to position, orientation and scale it is apparent that unary measurements from single line-segment will not embed such a powerful description. Consider for example, the orientation information extracted from a single line-segment as a possible attribute. Although it would remain unchanged during translation or scaling of the image (or object), small changes in the orientation (rotation) or viewing angle would lead to changes in the attribute. Such problems disappear,

if instead of measuring unary line-segment attributes, one bases the representation on pairs of line-segments. In this chapter, we investigate the properties of various geometric attributes computed from pairs of line-segments and address their ability to accurately and robustly represent line-pattern descriptions of images or objects.

## 2.1 Pairwise Geometric Attributes

Pairwise geometric attributes have been used on several occasions in the literature (Bray and Hlavac, 1991; Evans et al., 1993; Stein and Medioni, 1990; DiMauro et al., 1996; Christmas et al., 1995). The most commonly used attribute is the relative angle attribute that describes the angle between a pair of line-segments. We are going to investigate five different geometric attributes to construct our object representation from the line segments. These are:

- the relative angles,
- the ratio of segments length,
- the ratio of segment end-point distances,
- the ratio of segments length to intersection (referred to as relative position),
- the line-segment projection cross ratio.

The geometric attributes considered here have the major advantage of being invariant to changes in scale, position and rotation. They also present limited invariance to change in viewing position. This presents a major advantage over other possible attributes found in the computer vision literature. For example, Thacker *et al.* (Thacker et al., 1995) proposed to use the perpendicular distances between segment pairs. It is clear that using such attributes for the representation will not allow direct invariance to scale. Instead, a different representation will have to be computed for each

and every scale change that the algorithm needs to be able to cope with. Needless to say that the speed of the algorithm is dramatically decreased and the storage space required for each database entry is increased. In their super-segment based representation, Stein and Medioni (Stein and Medioni, 1990) also made use of the relative angle attribute between successive pairs of connected segments. In addition, each super-segment (set of connected line-segments) contains information about, the number of segments it is composed of, the sum of the line-segment length, its location and its overall orientation. Again, the use of many of these attributes for our representation would compromise its invariance properties.

The raw information available for each line-segments is the orientation (slope or angle with respect to the horizontal axis) and length. To illustrate how the pairwise attributes are computed, suppose that the two raw-attributes for the line indexed $(ab)$ are denoted by the vector $x_{ab} = (l_{ab}, \theta_{ab})^T$ where $l_{ab}$ is the line-segment length and $\theta_{ab}$ is the line-segment orientation (see figure 2.1).

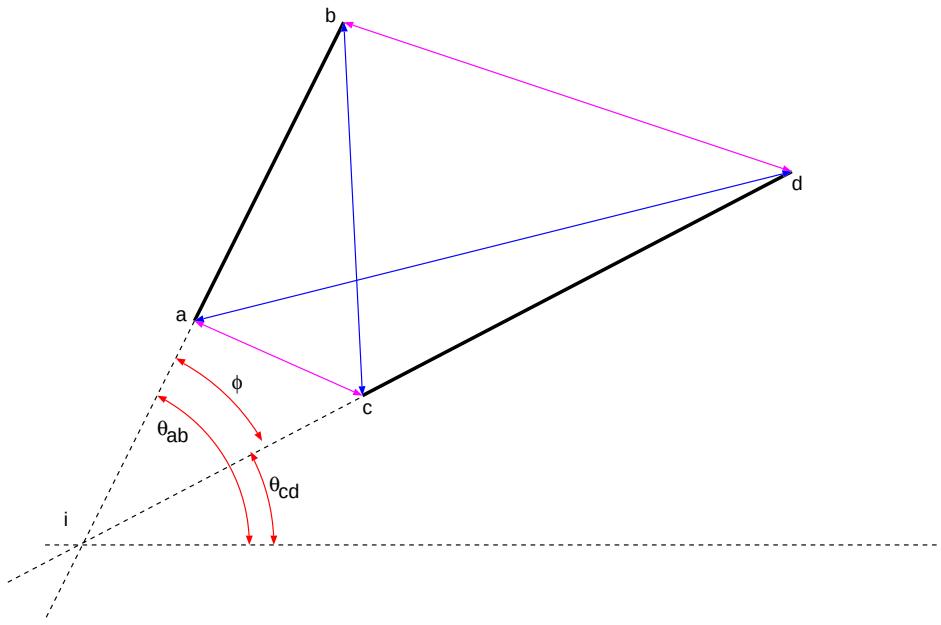



Figure 2.1: Computing the pairwise geometric attributes from line segments $(ab)$ and $(cd)$.

- The relative orientation $\phi$ between the lines indexed ($ab$) and ($cd$) is equal to

$$\phi_{ab,cd} = \min[(\theta_{ab} - \theta_{cd}), (\theta_{cd} - \theta_{ab})]$$

- The ratio of line-segment length $r$ is given by

$$r_{ab,cd} = \frac{\min[l_{ab}, l_{cd}]}{\max[l_{ab}, l_{cd}]}$$

- The ratio of the base line-segment length with its distance to intersection with the line-segment pair $\vartheta$ (relative position) is computed from

$$\vartheta_{ab,cd} = \frac{1}{\frac{1}{2} + \frac{l_{ib}}{l_{ab}}}$$

- The ratio of segment end-point distances $epr$ is generated as follows

$$epr_{ab,cd} = \frac{\min[l_{ac}, l_{bd}]}{\max[l_{ac}, l_{bd}]}$$

- The line-segment projection cross-ratio $xr$ is computed as follows:

$$xr_{ab,cd} = \frac{\min[l_{ad}, l_{bc}]}{\max[l_{ad}, l_{bc}]}$$

The methodology used to compute the histogram attributes allows us to know the range of the attributes. The relative orientation attribute will range between $0$ and $\pi/2$, the line-segment length ratio between $0$ and $1$, the relative position between $0$ and $1$, the ratio of segment end-point distances between $0$ and $1$ and, finally, the cross-ratio between $0$ and $1$.

The retrieval performance associated with each of the pairwise attributes described above may be found in chapter 3 section 3.3.1. The results show that the relative angle attribute provides the best representation for object recognition. The attributes based on line-segment length ratio are too sensitive to the effects of the low level feature extraction process (i.e. line splitting and line end-points displacement). However, the relative angle as it has been described above may be further enhanced to reduce possible description ambiguities.

## 2.2 Directed Pairwise Geometric Attributes

The relative angle pairwise attributes have been used for representing shape from their polygonised approximation (Evans et al., 1993; Huet and Hancock, 1996b; Huet and Hancock, 1996a; Stein and Medioni, 1990). However these simple pairwise attributes can become quite ambiguous. Figure 2.2 depicts three segments. Segment $(ab)$ is set as the baseline and we compute the pairwise relative angle between line-segment $(ab)$ and $(cd)$ (angle $\theta 1$ in figure 2.2) as well as between $(ab)$ and $(ef)$ (angle $\theta 2$ in figure 2.2). From this example the underlying ambiguity of the un-directed relative angle attribute is apparent. One can easily see that $\theta 1$ is equal to $\theta 2$ while they clearly should not.

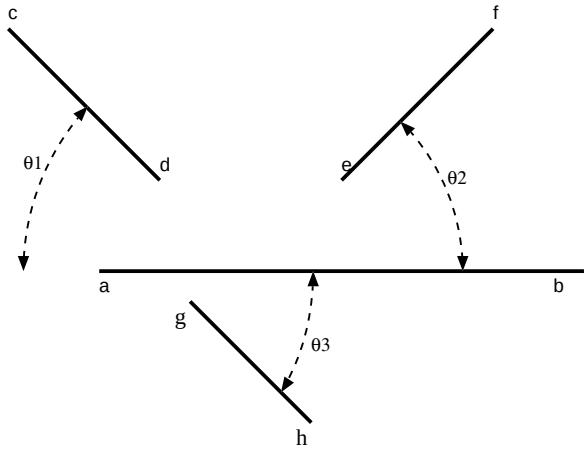



Figure 2.2: Non directed geometric attributes.

The relative angle between segment pairs may be unambiguously determined if both segments are directed according to some pre-defined rule. Figure 2.3 shows how directing both segments away from their point of intersection allows us to un-ambiguously define the relative angle between segments. This is an extension to the relative angle attributes used by Evans et al. (Evans et al., 1993).

A final enhancement is required to cater for the line-segment $(gh)$, which with the current extension still shares the same directed relative angle as line-segment

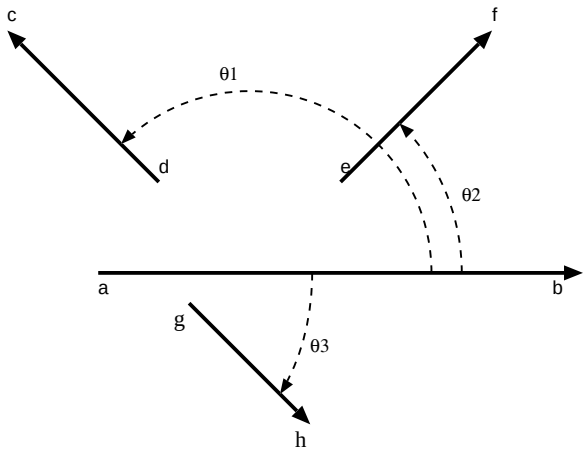



Figure 2.3: Directed geometric attributes.

$(ef)$ with line-segment $(ab)$. The difference between both relative angles is related to the direction of travel from the baseline $(ab)$ to either of its pairs ($(ef)$ and $(gh)$). Therefore, it seems natural to give the relative angle a positive sign if the direction of the angle from the baseline $x_{ab}$ to its pair $x_{cd}$ is clockwise and a negative sign if it is counter-clockwise.

We are using the directed relative angle as the base for the representation. The raw information available for each line segment is its orientation and its length (see figure 2.4). To illustrate how the pairwise feature attributes are computed, suppose that we denote the line segment indexed $(ab)$ and $(cd)$ by the vectors $x_{ab}$ and $x_{cd}$ respectively, directed away from their point of intersection. The relative angle attribute is given by

$$\theta_{x_{ab},x_{cd}} = \arccos\left[\frac{x_{ab} \cdot x_{cd}}{|x_{ab}||x_{cd}|}\right]$$

From the relative angle we compute the directed relative angle. This is an extension to the attribute used by Evans et al. (Evans et al., 1993), that consists of giving the relative angle a positive sign if the direction of the angle from the baseline $x_{ab}$ to its pair $x_{cd}$ is clockwise and a negative sign if it is counter-clockwise. This allows us to extend the range of angles describing pairs of segments from $[0, \pi/2]$ to $[-\pi, \pi]$.

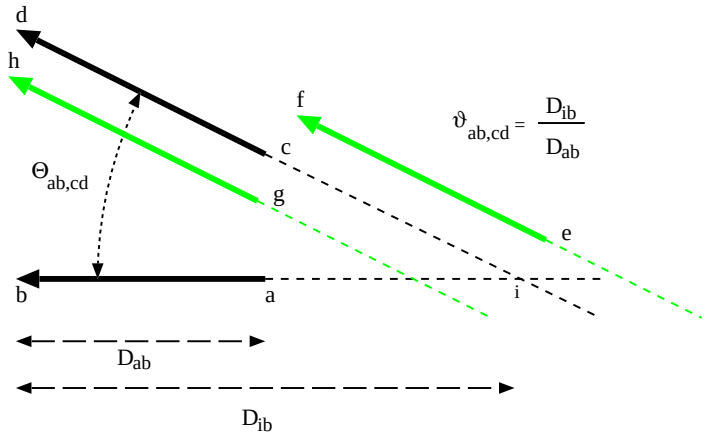



Figure 2.4: Computing the geometric attributes.

In order to describe the relative position between a pair of segments and resolve the local shape ambiguities produced by the relative angle attribute, a second attribute is generated. This second attribute allows the representation to capture the structural arrangement of the shape. Without this structural attribute the line-segments $(cd)$, $(ef)$ and $(gh)$ of Figure 2.4 would be represented in exactly the same way. The relative angle between segment $(cd)$, $(ef)$ and $(gh)$ and the base line-segment $(ab)$ is the same, even using directed geometric attributes. For this reason, an extra attribute is required in order to unambiguously represent object shapes. The directed relative position $\vartheta_{x_{ab},x_{cd}}$ is encoded by the length ratio between the oriented baseline vector $x_{ab}$ and the vector joining the origin of the baseline segment $(ab)$ to the intersection of the segment pair $(cd)$.

$$\vartheta_{\mathbf{x}_{ab},\mathbf{x}_{cd}} = \frac{D_{ib}}{D_{ab}}$$

As a result, the range of this attribute is $[1/2,\infty)$. A relative position of $\infty$ indicates that the two segments are parallel. A relative position of $1/2$ indicates that the two segments intersect at the middle point of the baseline. A particularly interesting value for the relative position attribute is when the length ratio is equal to $1$. This is likely to originate from a pair of connected segments. Indeed the point of

intersection of connected line-segments is the end-point both line-segments have in common. This leads to $D_{ib} = D_{ab}$ and therefore a relative position of $1$.

It is not computationally convenient to handle infinite values and histograming is no exception. It is appropriate to normalise the above described directed relative position $\vartheta_{x_{ab},x_{cd}}$ so that the physical range of this attribute is $(0, 1]$. This is effected by computing $\vartheta_{x_{ab},x_{cd}}$ in the modified manner:

$$\vartheta_{a,b} = \frac{1}{\frac{1}{2} + \frac{D_{ib}}{D_{ab}}}$$

Now we consider the special case of parallel segment pairs. In such a situation the relative angle is $0$ and the relative angle is undefined. We allow for a relative position of $0$ to indicate that the two segments are parallel. 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.* (Evans et al., 1993) if the technique is to be used for shape retrieval from real world image databases where the size of objects or, more generally shapes, is not known in advance.

## 2.3 The Image DataBase

The underlying task common to content-based image retrieval and object recognition from large model-base techniques is to index or recognise a large collection of images according to their similarity with a query image. The problem of retrieving images from a database based on their content similarity to some reference image directly implies that some features can be automatically (or semi-automatically) extracted from the digital images. Most of the recent image database systems make use of three distinct pieces of information contained in the image pixel map. These are colour, texture and shape. Various approaches for solving this problem have been

proposed. A review of those approaches may be found in Chapter 1 Section 1.5. Our current research is concerned solely on retrieval of image or object views that have shape similarities with the query image (or sketch). In order to study and access the efficiency of the current and the proposed algorithms we require large data sets. The following databases will be used for our experiments.

### 2.3.1 Infra-Red Aerial Images

This image database consists of 22 aerial infra-red line scan images (see figure 2.6) and a digital map (see figure 2.5). These images are of both rural and urban areas. The images are formed by a line-scan process in the horizontal direction. The line-scan is controlled by a rotating mirror. Aircraft motion is responsible for the sampling in the vertical direction. The main features are man-made road structures that radiate strongly in the infra-red band. These features present themselves as intensity ridges in the infra-red images. They are extracted using a relaxational line-finder (Hancock, 1993) which encourages contour connectivity using a dictionary of local line structure. Straight-line segments obtained by polygonising the raw line features extracted from the infra-red images are used to compute the feature attributes discussed in Section 2.1.

Some of the aerial views depict the same area but have been captured at different aircraft altitudes. This will obviously have the effect of changing the scale of the objects present in the scene. Because of the geometry of the line-scan capture process, there are known to be severe barrel distortions around the border of each aerial view. The distortions are such that the data cannot be recovered by applying simple Euclidean transform to the data.

A typical query example for this database is the problem of indexing using a digital map (depicted in Figure 2.5) of an area known to be contained in some of the images and see whether the corresponding infra-red images are retrieved accordingly. The aerial views labelled *Map060* and *Map170* in the database (Figure 2.6)

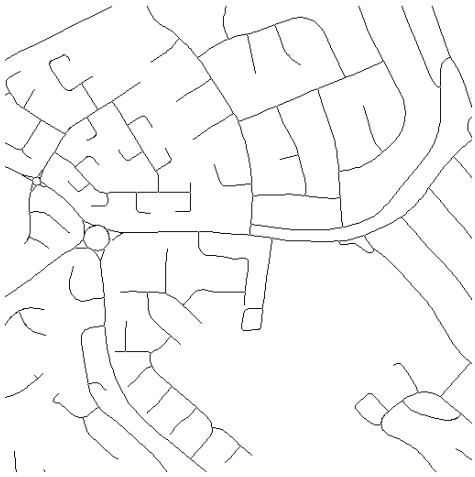


Figure 2.5: The digital map.

depict the same area as the digital map. These two images are taken at different aircraft altitude and cannot be aligned directly with the digital map.

### 2.3.2 TradeMark Database

This database is composed of 5430 trademarks from Canadian and international companies. We would like to thank M. Flickner from IBM Almaden research center for providing us with these images. We will be using a subset of approximately 2000 images out of the 5430 for our retrieval experiments. All the images in this database are binary (black and white) and vary greatly in terms of quality of acquisition since they come from scanned and faxed documents. We have used the Canny edge-finder (Canny, 1986) to extract the edges from the images. The images are then processed further using a simple polygonisation algorithm, bearing similarities with the one used by Evans *et al.* (Evans et al., 1993). Straight-line segments extracted from the images are used to compute the feature attributes discussed in section 2.1.

A typical example for this application is to retrieve logos that have a similar shape (circular, rectangular, triangular, etc...) or are from the same company.

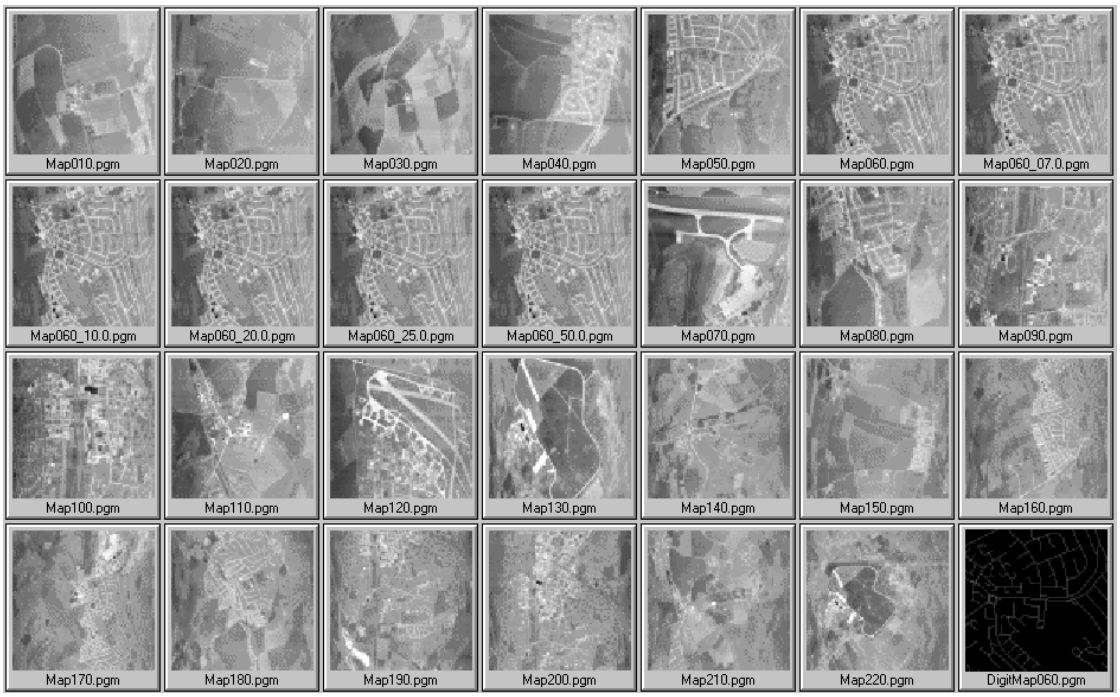



Figure 2.6: The infra-red aerial image database.

### 2.3.3 Character Database

This database is smaller in size and the shapes contained are much simpler than those found in the previous databases. It contains a few hundred alpha-numeric shapes and the number of segments per image rarely exceeds twenty (see figure 2.8). The purpose of this collection is to test the invariance characteristics of the proposed recognition methods. For example several exemplar of the letter “A” are found in this database with various orientations, scales, positions, degrees of partial occlusion and segmental noise level.

## 2.4 Low Level Processing

We are concerned with the task of recognising and indexing into large libraries (or databases) of images according to their content. As described earlier we concentrate our efforts on the shape aspect of the images or objects. Our approach assumes the

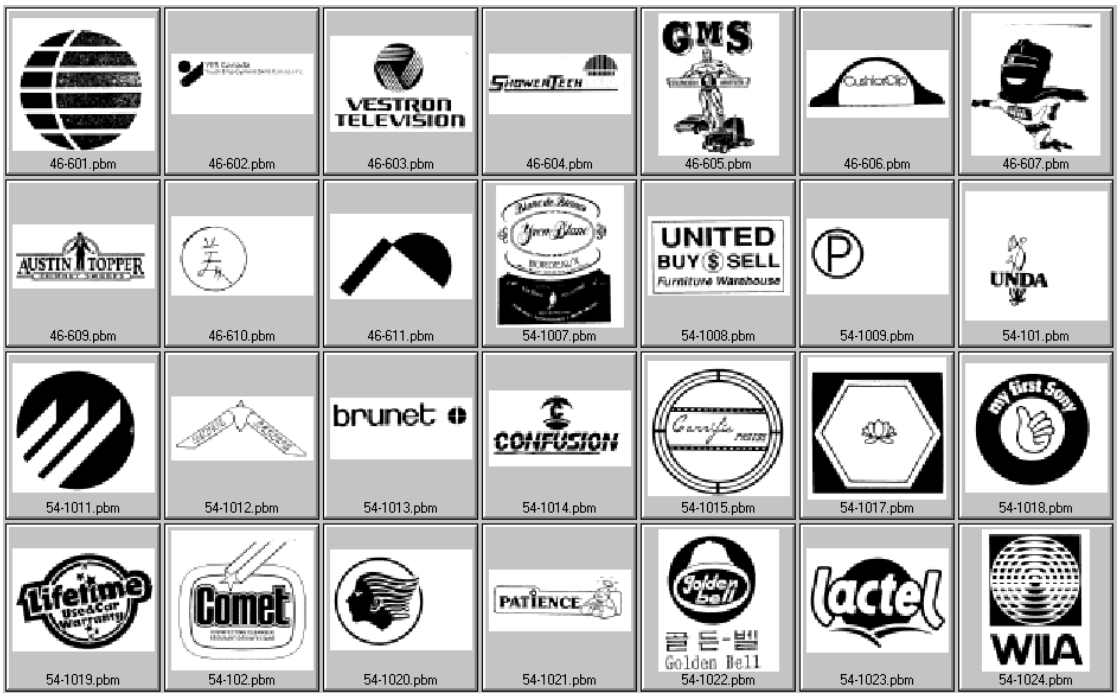



Figure 2.7: Subset of the trademark image database.

line-pattern corresponding to every image in the database has been computed. It is from those line-patterns that the representation for each image or object is going to be generated. Here follows a description of how the creation of the line-pattern is effected. Each image in the database is subjected to the following processing steps prior to histograming and then matching.

- **Feature Detection:** In the case of the aerial infra-red images we apply a relaxational line-finder to detect road structures (Hancock, 1993).For the trademarks and logos we use the Canny edge-detector to locate object outlines (Canny, 1986). Both operators deliver a binary feature map that contains connected edge pixels that are a single pixel wide. It is from this set of irregular chains of edge pixels that we are going to produce the line-pattern.

- **Polygonalisation:** Straight line segments are identified using an algorithm similar variant of the algorithm originally developed by Lowe (Lowe, 1987) and later refined by Rosin and West (Rosin and West, 1989). The basic idea is

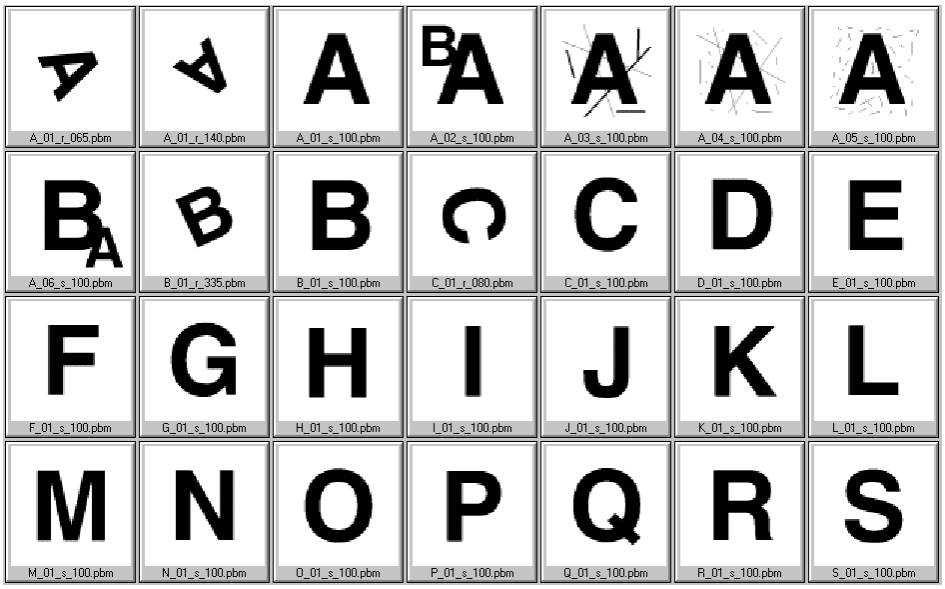



Figure 2.8: Subset of the character image database.

to find a polygonal approximation to an irregular chain of feature pixels using a recursive-split algorithm. 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 under consideration. The end-points of the chain of edge pixels are used during the first recursion. If the sagittal distance is larger than the threshold, a new control point is placed on the contour, splitting the current chord into two new chords. This process is repeated until no further chords (line-segments) may be split. 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. However, since digital images are discrete representation (composed of pixel) there is a point where the definition of the smallest details of the edge contours will not be polygonised in a ”scale invariant” manner. For example, it would be very useful to be able to polygonise circles of various radii with the same number of line-segments. As the resolution (radius in number of pixels) of the circle becomes very small, the sagittal distance precision is very low and the threshold value ineffective.

A minimum segment length is required in order to compute an accurate ratio used to decide whether splitting is required or not. The line-segments produced by the polygonisation process are also constrained in terms of minimum length. This potentially allows for a large number of spurious edge pixel chains, which are likely to correspond to noise in the original image to be removed from the representation. However, as the value of this parameter increases the finer details are removed. We have experimented with a number of possible candidate values for the minimum line-segment length and have decided to set it to 5 pixels.

- **Graph Construction**: One of the strengths of our approach is the use of relational information about the image features, in our case line-segments, to constraint and enhance the attribute representation. A number of adjacency graph structures are going to be used as part of the line-pattern representation and their effectiveness evaluated and compared. Here we describe the process by which the graph structures used in this thesis, are created.

  - **N-Nearest Neighbour Graph**: The N-nearest neighbour graph is one of the most conceptually simple closest-point graphs. The method used to generate such a graph structure is as follows. 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 the closest Euclidean distance on the image plane. Choosing N equal to the number of lines in the image but one results in a fully connected graph. This situation will also be considered in our experiments.

  - **Delaunay Graph**: Compared to the nearest neighbour structure the Delaunay graph is more elaborate. The Delaunay graph is more easily described as being the dual of the Voronoi tesselation of image points (which in our case correspond to the centre-points of each line-segment). The

Voronoi tesselation of a set of planar points is a set of polygons. Each polygon corresponds to a point. Each pair of points is used in turn to split the plane in two halves following the perpendicular bisector of the lines-segment joining the two points. A polygon is obtained from the convex intersection of all perpendicular bisectors related to the associated point. From the Voronoi tesselation, the construction of the Delaunay graph simply consists of connecting adjacent areas (polygons) of the tesselation by a graph edge. This graph structure has been shown (Tuceryan and Chorzempa, 1991; Wilson and Hancock, 1997) to provide the most robust relational structure to noise. However, the complexity of efficient algorithms to generate the Delaunay triangulation is such that they are non-trivial to implement. Simple algorithms also exist but their computational requirements make them inappropriate to operate in a rapid object recognition framework. These conditions also rule out the use of the Gabriel graph and relative neighbourhood graph which are both derived from the Delaunay triangulation.

In the context of the field of computer vision, closest-point graph structures are often employed to capture perceptual groupings in an image. The study of the grouping phenomenon in the human vision system by Gestalt psychologists (Koffka, 1935; Wertheimer, 1938) indicated that there are rules of human vision organisation. These include proximity, similarity, continuity, closure and symmetry in analysing images. The work presented here attempts to embed some of those criteria within the representation of images and objects.

## 2.5 Performance Measures

In order to provide comparative results between the various approaches, we need to define a measure representing the level of retrieval or recognition accuracy. Typi-

cally, object recognition algorithm performance is measured in terms of the percentage of times that an object is correctly identified. This is not particularly suitable for the recognition task at hand. Instead of identifying the model from the library that corresponds to the object under consideration, we aim at retrieving all the models from the database that have sufficient similarity with the query image or object. A performance measure that captures the position (ranking) in which target models appear in the retrieval or recognition is required. To evaluate retrieval performance in such situations, a normalised recall metric can be employed (Faloutsos et al., 1994). This measure was devised by the content-based retrieval community as a mean of algorithm comparison. The idea is as follows. Suppose that there are $n$ items (or models) of the same category in the database and these $n$ items appear in the first $n$ positions for a similarity-based retrieval. In this case, the performance measure should indicate maximum accuracy. As the position of the $n$ models from the same category get further apart from the first $n$ positions the performance measure should decrease gradually. It is important to note that the ordering of the models within a category is not crucial. What really matters, however, is that all $n$ models from the correct category appear top in the order of retrieval. Retrieval accuracy is measured in terms of deviation from the ideal average rank of relevant (IAVRR) items (or models). The IAVRR is the ideal average rank which corresponds to the case where all $n$ models from a category have been retrieved in the first $n$ positions. For a database that contains $n$ models in each category the IAVRR is $n/2$. In most cases, the library will be composed of multiple categories each containing varying number of models. The general equation for the ideal average rank is

$$IAVRR = \frac{1}{m} \sum_{i=1}^{c} \frac{n_i^2}{2}$$

where $m$ is the total number of models in the library, $c$ is the number of categories, and $n_i$ is the number of models in the $i^{th}$ category. Normalised recall is measured by comparing the IAVRR with the average rank of all the relevant models (AVRR) for each retrieval. The AVRR is computed based on the actual ordered ranking of $n_i$

models in a particular category for each database retrieval.

$$AVRR = \frac{1}{m}\sum_{i=1}^{c}\sum_{j=1}^{m}\begin{cases} j & \text{if } k_j \in C \\ 0 & \text{otherwise.} \end{cases}$$

where $k$ is the ranking of each model with respect to a query (ie: the best match is 1 the second best is 2 and the least similar possible ranking is $m$) and $C$ is the set of images from the current category. The ratio of the IAVRR to the AVRR can be used to give a measure of average retrieval accuracy over a number of experimental trials. Perfect performance would yield to a ratio $AVRR/IAVRR = 1$. The performance measure posed in this manner presents two very undesirable drawbacks. Both the IAVRR and AVRR measures are currently computed by averaging over all the image models present in the database. In many cases the database is so large that using every model as a query in turn in order to obtain a performance measure is impractical. Furthermore, in order to compute an accurate value for the IAVRR every image or model in the database needs to be assigned to a corresponding ground truth object category by hand. Again such a restriction is impractical in many cases. For example, suppose that in order to increase the size of the database, a large number of synthetically and randomly generated models are included. It is in most cases impossible to assign such models to a particular category. In order to allow for both non-categorised models and a reduced (partial) image query set, the definitions of the IAVRR and the AVRR have to be rewritten. We have done so by devising category dependant IAVRR and AVRR which are defined as follows:

$$IAVRR_c = \frac{1}{m}\sum_{i=1}^{c} i = \frac{n_c + 1}{2}$$

where $n_c$ represents the number of models in the category $c$ under consideration.

$$AVRR_c = \frac{1}{n_c}\sum_{i=1}^{m}\begin{cases} i & \text{if } k_i \in C \\ 0 & \text{otherwise.} \end{cases}$$

The final performance measure, which represents the retrieval accuracy of the algorithm under consideration, is still based on the ratio of AVRR by IAVRR but is now

averaged over the queries.

$$Perf = \frac{1}{n_q} \sum_{c=1}^{n_q} \frac{AVRR_c}{IAVRR_c}$$

where $n_q$ represents the number of queries for the current experiment. All the retrieval performance results provided in this thesis have been computed using this measure.

## 2.6 Summary

In this section we have presented the prerequisites of the work presented in this thesis. The feature attributes, relative angle and relative position, computed from pairs of line segments have been selected among a number of possible alternatives and their computation described in details.

The content of the large library of models was then described and the methodology used to automatically compute the representation from the image was explained. Finally, the performance measure, which will allow us to compare the retrieval performance of the various algorithms and image representations, was described. We will now address the problem of rapidly reducing the number of possible candidates from a large database, which satisfies the shape similarity criterion.

# Chapter 3

# Histogram Based Representation and Retrieval

We have established, in our literature review (see Chapter 1 Section 1.2, that there are a number of shortcomings associated with the current object recognition methods. The major drawback of object recognition techniques is their computational complexity. Since efficiency is a key issue when performing object recognition in large image libraries, conventional methods are too computationally demanding to be used as search engines. This chapter presents a new shape representation for rapidly indexing and recognising line-patterns from large databases. The basic idea is to exploit both geometric attributes and structural information to construct a shape similarity measure. We realise this goal by computing the N-nearest neighbour graph for the line-segments for each pattern. The edges of the neighbourhood graphs are used to gate contributions to a two-dimensional pairwise geometric histogram. Shapes are indexed by searching for the line-pattern that maximises the cross-correlation of the normalised histogram bin-contents. We evaluate the new method on a database containing over 2500 line-patterns each composed of hundreds of lines. Here we demonstrate that the structural gating of the histogram not only improves recognition performance, but that it also overcomes the problem of

saturation when large patterns are being recalled.

## 3.1 Related literature and Motivation

It is important to stress that our aim in this chapter is to perform a rapid indexing of a large database of line-patterns in order to reduce the number of possible model candidates for further object recognition processing. This is in marked contrast to the bulk of the current literature on histogram or accumulator based image retrieval, which has a tendency to concentrate on pixel-based attributes such as colour (Swain, 1993), orientation or texture (Gimelfarb and Jain, 1996).

Our idea of using an accumulator-based method to recognise line-patterns clearly has similarities with a number of object recognition strategies described elsewhere in the literature. For instance, the idea of voting is central to the generalised Hough transform (Ballard, 1981) and its variants (Illingworth and Kittler, 1987). However, Hough-based methods are concerned with shape recovery for the purposes of object localisation. It is therefore important to stress that in our method, the accumulation of evidence is directed at measuring object similarity and presence in the scene rather than localisation. The idea of using attribute histogram as an indexation device was originally popularised by Swain and Ballard (Swain, 1993) for retrieving colour images from databases. The idea has been extended to both texture (Gimelfarb and Jain, 1996) and orientation representations (Dorai and Jain, 1995; Rigoutsos and Hummel, 1995). The work presented is more closely related to the pairwise geometric histogram representation of Evans *et al.* (Evans et al., 1993). Rather than using raw image attributes, this representation uses relative attributes defined over line-pairs. Several alternative attribute sets have been suggested (Bray and Hlavac, 1991), but most algorithms revolve around the use of angle difference. The work of Thacker *et al.* (Thacker et al., 1995) is more in line with what we are aiming to achieve. Histogram based approaches to object recognition can be viewed as accu-

mulator methods that avoid the problems of storage and search that limits the use of the generalised Hough transform.

The use of local geometric features is common to a number of object recognition strategies. For instance, geometric hashing attempts to tabulate local features in such a way as to index object-shape (Hecker and Bolle, 1994; Tsai, 1994). Stein and Medioni (Stein and Medioni, 1992) provide one of the most ambitious hashing scheme, which draws on relatively complex polygonal super-segments as the geometric primitives. Although this representation is considerably more sophisticated than our use of simple line-segment pairs, it is potentially more fragile as a result of sensitivity to the token grouping process. Rigoutsos and Hummel (Rigoutsos and Hummel, 1995) soften the search process using Bayes decision theory. However, these methods require the identification of a set of global token correspondences. In our method, there is no such requirement since we compare objects at the histogram level. It is this feature that also distinguishes our work from that of Grimson and Lozano-Perez (Grimson and Lozano-Perez, 1987) who use pairwise relational constraints to identify objects via the search of the interpretation tree.

Both structural information and attribute histograms have been separately used to recall and recognise image data. However, they have not been used in conjunction. The aim in the work presented in this chapter is to fill this gap in the literature by developing a structurally gated pairwise attribute histogram representation that can be used to recall complex line-patterns. It is this idea of using histogram gating together with our aim of indexing into large databases, which gives us a novel advantage over the work of Thacker *et al.* (Thacker et al., 1995). Although this work has provided us with a starting point, we have considerably refined the choice of feature representation to additionally allow for invariance to scaling. Moreover, we reduce the computational overheads by conglomerating the line-segments in a single histogram rather than having a histogram per line-segment. Viewed from the perspective of structural hashing (Costa and Shapiro, 1995; Sengupta and Boyer,

1995), our method uses a compact histogram representation rather than a large hash-table, which can grow to unmanageable size for complex scenes. It is important to emphasise that we view the histogram-based comparison as the first-step in a hypothesis refinement process that draws on detailed matching for later verification as described in Chapter 4 and 5.

The outline of this chapter is as follows. Having described in the previous chapter (see Chapter 2 Section 2.1) the pairwise geometric attributes that may be used to represent the shape content of each image (or line-pattern) in the database, we may now further our explanation of the rapid indexing method. In Section 3.2 we describe our histogram based recognition method. This focuses on details of histogram accumulation and comparison. In Section 3.2.1 we present various relational structures and address their suitability both in term of simplicity of computation and robustness to noise for the task at hand. Experimental evaluation of the technique is presented in Section 3.3. This takes the form of a comparative study aimed at establishing the effectiveness of the proposed algorithm alternatives on a database of over 2500 line-patterns. This experimentation has several goals. The first sensitivity analysis is concerned with the choice of the most appropriate pairwise attribute to represent line-patterns. In the second experiment we investigate the suitability of a number of distance measures used to gauge histogram similarity. The third experiment aims to establish the most effective graph structure for gating the histograms. Finally, we present some retrieval examples using both the gated and un-gated pairwise geometric histogram representation in Section 3.3.4. Section 3.4 addresses the sensitivity of the rapid recognition methods to errors of the line-segmentation process. Finally, Section 3.5 presents some conclusions and suggests directions for future investigation as far as this approach is concerned.

## 3.2 Histogram Representation

The two dimensional histogram representation is based on the relative pairwise attributes described in Chapter 2 Section 2.2. In this section, we provide details about the construction of pairwise histograms from two invariant attributes together with the structural constraints provided a N-nearest neighbour graph. We also describe the distance measures used to compare the histogram bin contents.

The method used to compute the pairwise attributes (see chapter 2 Section 2.2) allows us to know their range. The directed relative angle attribute will range between $-\pi/2$ and $\pi/2$, and the directed relative position between $0$ and $1$. Therefore, we are able to choose the right number of bin and their corresponding bin size for each histogram. For the experiments, described hereafter, the directed relative angle dimension of the histogram is composed of 36 bins (each bin spanning an angle of $\pi/18$ radian). The dimension of the histogram based on the ratio of the base line-segment length and its distance to the intersection point with the line-segment pair (directed relative position) is composed of 12 bins. The experiment from which the number of bins in each histogram is selected was described in (Huet, 1996). The final two-dimensional histogram representing each line-pattern is composed of 36 bins (directed relative angle) by 12 bins (directed relative position). All our histograms are normalised so that the sum of all the bins is equal to unity.

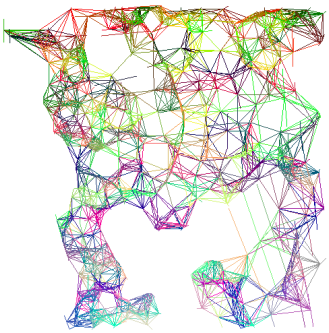
### 3.2.1 Relational Constraints

In the previous chapter, we have shown that it is important to carefully choose the attributes (or measurements). Similarly, we intend to reveal how critical the choice of whether or not using structural information within a representation may be for the performance of the algorithm. Recent research in the field of graph matching (Shapiro and Haralick, 1985; Wilson and Hancock, 1997; Cross et al., 1996; Bunke and Messmer, 1995) indicates that using the structural (or relational) information of

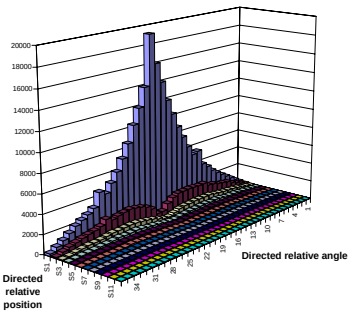
(a) Original image.

(b) Raw feature image.

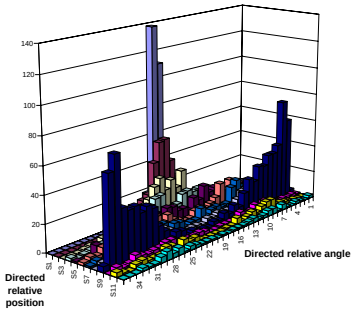
(c) Line image.



(d) Nearest-Neighbour graph.




(e) 2D Histogram.




(f) 2D Relational Histogram.

Figure 3.1: A typical aerial infra-red image going through the processing step leading to the histogram representation.

image features may provide recognition improvements when image noise and measurement errors are present. The basic idea is to use image features as graph nodes and related features as graph edges.

There are many alternative graph representations. Four different neighbourhood graph structures have been particularly studied (Tuceryan and Chorzempa, 1991; Wilson and Hancock, 1997). These are the Delaunay graph, the Gabriel graph, the relative neighbourhood graph and the N-nearest neighbour graph. The difference among the various type of graph depends on the algorithm used to create the edge-set. In other words, the rule that defines how graph nodes are connected (are related) to each other. In Chapter 2 Section 2.4 we described a number of alternative graph structures.

It is interesting to 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.

The work of Tuceryan and Chorzempa (Tuceryan and Chorzempa, 1991) and Wilson and Hancock (Wilson and Hancock, 1997) indicates that the most robust graph representation in presence of noise is the Delaunay graph. The N-nearest neighbour graph comes close second. It was also shown that various orders of N-nearest neighbour graphs lead to various level of structural matching accuracy. The results of Wilson and Hancock (Wilson and Hancock, 1997) illustrate that optimal performance is obtained when the number of neighbours is in the range 5-6. It is interesting to note that these graphs also have approximately the same average connectivity as the Delaunay graph. The Gabriel graph and the relative neighbourhood graphs deliver performance that drops off rapidly with increasing noise. This is an observation of critical importance for our study of line-pattern indexation. The computation of the Delaunay graph is intensive and therefore not particularly well suited for large image databases. The N-nearest neighbour graph on the other hand can be computed

inexpensively using very simple algorithms. Moreover, as demonstrated by our experimental study, the increased computational efficiency is only at the expense of marginal performance deterioration.

One of the novel ideas introduced in this chapter is to embed relational and structural information about the line-pattern within the pairwise geometric histogram using the edge-set of the N-nearest neighbour graph. Although our histograms integrate information over the complete line-pattern, contributions are gated by the local structure conveyed by a neighbourhood graph. The motivation here is that local object representations are more robust to occlusion, missing or extra features and noise. Moreover, by integrating both geometric attributes and structural information into a single histogram we provide a compact relational representation of line-structure. This gating process greatly reduces the number of entries in each histogram, hence diminishing the risk of histogram saturation. Structural gating ensures that this representation does not become saturated. As pointed out by Stricker and Swain (Stricker and Swain, 1994), this feature can severely restrict the recall capacity of the recognition process.

Our idea contrasts with a number of related contributions in the literature. For instance, Evans *et al* (Evans et al., 1993) effectively have a local representation of line-pattern structure, which is less compact than ours since it employs one histogram per line-segment. Moreover, their attributes are not scale invariant. Di Mauro *et al's.* (DiMauro et al., 1996) idea of using a region-of-interest is close to that of using a neighbourhood graph. However, their regions are controlled by a scale parameter. By virtue of their structural character, neighbourhood graphs are scale-invariant.

Suppose that line-segments extracted from an image are indexed by the set $V$. More formally, the set $V$ represents the nodes of our nearest neighbourhood graph. The edge-set of this graph $E \subset V \times V$ is constructed as follows. For each node in turn, we create an edge to the $N$ line-segments that have the closest distances. With the edge-set of the nearest neighbour graph to hand, we can construct the structurally

gated geometric histogram. The bin-incrementing process can be formally described as follows. Let $i$ and $j$ be two segments extracted from the raw image. The angle and position attributes $\theta_{ij}$ and $\vartheta_{ij}$ are binned provided the two segments are connected by an edge, i.e. $(i,j) \in E$. If this condition is met then the bin $H(B_\theta, B_\vartheta)$ spanning the two attributes is incremented as follows

$$H(B_\theta, B_\vartheta) = \begin{cases} H(B_\theta, B_\vartheta) + 1 & \text{if } (i,j) \in E \text{ and } \theta_{ij} \in R(B_\theta) \text{ and } \vartheta_{ij} \in R(B_\vartheta) \\ H(B_\theta, B_\vartheta) & \text{otherwise.} \end{cases}$$

where $R(B_\theta)$ is the range of the directed relative angle attributes spanned by the $B_\theta$th horizontal histogram bin and $R(B_\vartheta)$ is the range of the directed relative angle attributes spanned by the $B_\vartheta$th horizontal histogram bin. Each histogram contains $n_\theta$ relative angle bins and $n_\vartheta$ length ratio bins. The normalised gated geometric histogram bin-entries are computed as follows

$$h(B_\theta, B_\vartheta) = \frac{H(B_\theta, B_\vartheta)}{\sum_{B'_\theta=1}^{n_\theta} \sum_{B'_\vartheta=1}^{n_\vartheta} H(B'_\theta, B'_\vartheta)}$$

The gating process greatly reduces the number of entries in each histogram, hence diminishing the risk of histogram saturation. As an illustrative example some of the images used in our experiments contain over 1000 segments. Such a line-pattern would result in over $(1000 * (1000 - 1))/2) = 498500$ histogram entries. By contrast, in our structurally gated histograms based on a six nearest neighbour graph there are on average only $1000 * 6 = 6000$ entries. Both the un-gated and the gated histogram are depicted in Figure 3.1(e) and 3.1(f) for a typical image. In the case of the standard histogram representation many histogram bins contains large number of votes with the highest peak reaching 19000 entries. The highest peak for the structurally gated histogram is below 140 entries.

To provide some illustrative examples of our methodology, Figure 3.1 shows the sequence of processing steps from one of the infra-red aerial image leading to the extraction of the two-dimensional histogram representation. Figure 3.1(a) is the raw (original) image. The detected pixel chains are shown in Figure 3.1(b). In Fig-

ure 3.1(c) we show the straight-line segments that result from the application of our polygonisation algorithm. The 6-nearest neighbour graph generated from the center point of the straight line segments is shown in Figure 3.1(d). Figure 3.1(e) shows the standard pairwise geometric histogram, while figure 3.1(f) shows the gated histogram computed using a 6-nearest neighbour graph. In both histograms, the horizontal axis corresponds to directed relative positions and the vertical axis encodes directed relative angles. The main feature to note from the two histograms is the much sharper structure of the gated version.

### 3.2.2 Histogram Distance Measures

Having established the means of representation for our rapid indexing algorithms, a measure needs to be defined in order to indicate the similarity between line-patterns. Since we have opted for a histogram based representation we will investigate the effectiveness of a number of histogram distance measures. The normalised histograms $H_M(i)$ and $H_D(i)$ are composed of $n = n_\theta \times n_\vartheta$ distinct bins $i$, each representing the frequency $f$ of occurrence of a pairwise attribute between the line-segments within each line-set. The indices $M$ and $D$ correspond respectively to a model line-pattern and a query. The distance measures under consideration are the following:

- $L_1$ **Norm**

$$L_1(H_D, H_M) = \sum_i |H_D(i) - H_M(i)| \tag{3.1}$$

- $L_2$ **Norm**

$$L_2(H_D, H_M) = \sqrt{\sum_i (H_D(i) - H_M(i))^2} \tag{3.2}$$

- $Bhattacharyya$ **Distance**

$$B(H_D, H_M) = -\ln \sum_i \sqrt{H_D(i) \times H_M(i)} \tag{3.3}$$

- $Matusita$ **Distance**

$$M(H_D, H_M) = \sqrt{\sum_i (\sqrt{H_D(i)} - \sqrt{H_M(i)})^2} \tag{3.4}$$

- $Divergence$

$$D(H_D, H_M) = \sum_i [(H_D(i) - H_M(i)) \ln \frac{H_D(i)}{H_M(i)}] \tag{3.5}$$

Before we proceed to experiment with these different distance measures, it is important to understand the way in which they gauge differences in histogram structure. Both the $L1$ and $L2$ norms use the difference in histogram bin contents to determine the similarity in the line patterns. One of the problems associated with these two measures is an undue bias towards bins of zero contents. The Bhattacharyya distance is effectively a correlation measure. If the case of Gaussian probability density functions, the Bhattacharyya distance is proportional to the Mahalanobis distance between the class means. The Bhattacharyya distance uses the correlation between the normalised bin contents as a measure of pattern similarity. This means that bins of zero contents do not contribute to the distance. In other words, the measure favours association between model and data (query) histogram bins, which have dominant large content. For highly structured histograms, i.e. those which are not uniformly populated, this can lead to the selection of matches in which there is a strong compatibility between the salient structure of the model and the data. The Matusita distance is proportional to the negative exponential of the Bhattacharyya distance. The divergence has a more subtle structure. In the case of Gaussian mixtures, it does not only depend upon the between class Mahalanobis distance, it also gauges the difference in class covariance.

## 3.3 Experiments

The examples included here aim to illustrate the effectiveness of the two dimensional relational histogram representations for performing shape based recognition from a large database of images.

The database used in our experiments has been described in Chapter 2 Section 2.3. For some of our experiments, the recognition task is posed as that of finding

the infra-red images that contain a pattern of road structure represented in a digital map (see figure 2.5). Since we also aim to investigate the sensitivity of recall to the line segmentation process, we have augmented the database with four additional sets of over and under segmented line-segments for the target aerial infra-red image. This database may seem rather artificial in its construction. However, the trademarks and logos provide structured background data, which is more realistic than random line patterns. Each image in the database is subjected to the processing steps described in Chapter 2 Section 2.4 (Feature Detection, Polygonalisation and Construction of the N-nearest Neighbour graph) prior to histograming.

The overall aims of our experiments are to demonstrate the retrieval power and recognition accuracy of the structurally gated pairwise geometric histogram (or relational pairwise geometric histogram). Following recent work on image retrieval from a large database of images (Niblack et al., 1993; Pentland et al., 1994; Swain, 1993; Picard, 1995), the aim is to recover the set of images that resemble most closely a query image. As mentioned above, in the bulk of our experiments the query image (see figure 2.5) is a *Digital Map* that represents a pattern of a road structure known to occur in some of the infra-red aerial images present in the database.

We commence our experimental study by providing some empirical results, which point to the best choice of pairwise geometric attribute(s), histogram distance measure, representation and order of relational structure.

### 3.3.1 Pairwise Geometric Attributes

Figure 3.2 shows the performance results of using each of the pairwise geometric attributes described in Chapter 2 Section 2.1 to index into a small database of 22 aerial images (see Figure 2.6) using a digital map (see Figure 2.5). From the results shown in figure 3.2 it is obvious that using the relative angle pairwise attribute offers the best level of retrieval accuracy. Even on such an experiment, the attributes based on length ratios show their fragility. Indeed, the end-point position of the line seg-

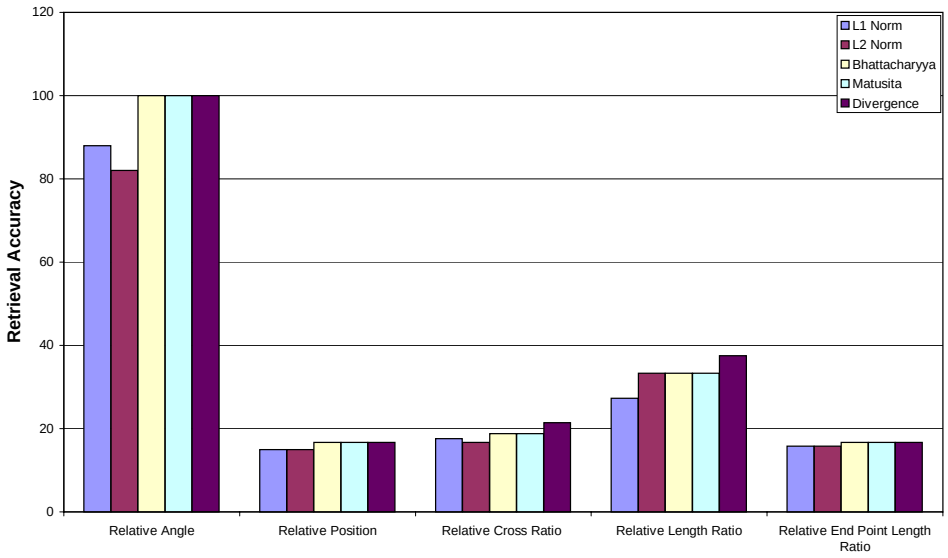



Figure 3.2: Performance results using various attributes for shape similarity retrieval.

ment is likely to be affected by the extraction process. Based on this experiment, we have established that a robust representation should favour orientation over length of the line-segment measurements. However, for ambiguity reasons, as described earlier in Chapter 2 Section 2.2, the representation needs to be augmented with an extra pairwise geometric attribute. From the results shown in figure 3.2, it appears that the relative length ratio is the most appropriate choice for the second attribute. Unfortunately, this attribute does not solve the ambiguity issues raised in Chapter 2 Section 2.2. Therefore, augmenting the relative angle representation with the relative length ratio would not improve recognition performance. However, the relative position attribute is able to solve the ambiguity that arise from using the relative angle alone. For this reason, the directed relative angle attribute will be combined with the directed relative position in a two-dimensional histogram representation.

### 3.3.2 Histogram Distance Measures

In order to determine the most appropriate distance measure for geometric pairwise histogram comparison we use the augmented aerial infra-red image database. The

multiple segmentations of the infra-red image related to the digital map allow us to measure the effectiveness of the various histogram distance measure in the presence of segmental clutter. The results in table 3.1 show the ranked distances between the ten first (most similar) aerial images and the digital map. The corresponding N-nearest neighbours graph may be seen in figure 3.2.

| Match | L1 Norm | | L2 Norm | | Bhattacharyya | | Matusita | | Divergence | |
|---|---|---|---|---|---|---|---|---|---|---|
| Best | Map60 | 0.31 | Map60 | 0.09 | Map60 | 0.01 | Map60 | 0.19 | Map60 | 0.14 |
| 2nd | Map60_20 | 0.34 | Map60_20 | 0.10 | Map60_20 | 0.02 | Map60_20 | 0.21 | Map60_20 | 0.19 |
| 3rd | Map60_07 | 0.37 | Map60_07 | 0.11 | Map60_07 | 0.03 | Map60_07 | 0.24 | Map60_07 | 0.24 |
| 4th | Map60_25 | 0.41 | Map60_10 | 0.13 | Map60_10 | 0.03 | Map60_10 | 0.27 | Map60_10 | 0.30 |
| 5th | Map130 | 0.44 | Map60_25 | 0.13 | Map60_25 | 0.04 | Map60_25 | 0.28 | Map60_25 | 0.32 |
| 6th | Map170 | 0.46 | Map150 | 0.14 | Map170 | 0.04 | Map170 | 0.28 | Map170 | 0.33 |
| 7th | Map60_10 | 0.46 | Map130 | 0.15 | Map180 | 0.04 | Map180 | 0.29 | Map180 | 0.34 |
| 8th | Map180 | 0.47 | Map180 | 0.15 | Map130 | 0.04 | Map130 | 0.29 | Map130 | 0.37 |
| 9th | Map90 | 0.48 | Map90 | 0.15 | Map90 | 0.04 | Map90 | 0.30 | Map90 | 0.37 |
| 10th | Map50 | 0.49 | Map50 | 0.15 | Map150 | 0.04 | Map150 | 0.30 | Map150 | 0.37 |

Table 3.1: Effect of segmental clutter on relative angle histograms using various distance measures.

The query image (the digital map depicted in Figure 2.5) corresponds to infra-red images Map60, Map60_07.0, Map60_10.0, ..., Map60_25.0 (various segmentation parameters) and Map170 (viewpoint variations). It is therefore, those database entries that should be retrieved as first best matches. At first sight, since all the techniques identify Map60 as having the closest histogram similarity to the model data, we may conclude that they deliver comparable performance. However, it should also be noted from these results that the $L_1$ and $L_2$ norms do not perform as well as the other metrics. Particularly, the $L_1$ and $L_2$ norms are the only measure pro-

viding an incorrect classification of Map170. It is clear from these results that the Bhattacharyya distance measure, the Matusita distance and the divergence are able to perform noticeably better that the standard $L_1$ and $L_2$. The sensitivity study accessing the retrieval performance of the different distances under varying quality of segmentation, revealed that the Bhattacharyya and Matusita distance measures are robust to significant differences in the segmentation process (extra and missing segments). Indeed, both distance measures outperform the divergence measure in many situations. As expected, the Matusita distance produces identical indexing results than the Bhattacharyya. However, we favour the use of the Bhattacharyya distance measure for computational reasons.

### 3.3.3 Histogram Representation and Distance Measure

To continue our study, we aim to establish which combination of distance measure and histogram representation (i.e. structurally gated or ungated), gives the best recognition performance. In figure 3.3 we show the effect of the size of the database on the accuracy of recall. The plot shows the fraction of times the correct image is top ranked when 20 trials are repeated. The results are plotted as a function of the size of the database. The plot presents a comparison of recognition performance between structurally gated and ungated histograms when some of the different distance measures outlined in section 3.2.2 are used for the of purposes pattern recall.

The main feature to note from the plot is that the results are clustered according to the distance measure used. The Bhattacharyya distance delivers the best performance, while the poorest performance is delivered by the L2 norm. In each case the structurally gated histogram outperforms the ungated histogram. However, in the case of the Bhattacharyya distance the margin of improvement is greatest. Over the range of database size, the performance of the gated histogram never falls below 85%. For the ungated histogram, the performance is consistently 30-40% poorer. Moreover, without gating, the performance of the Bhattacharyya distance is only

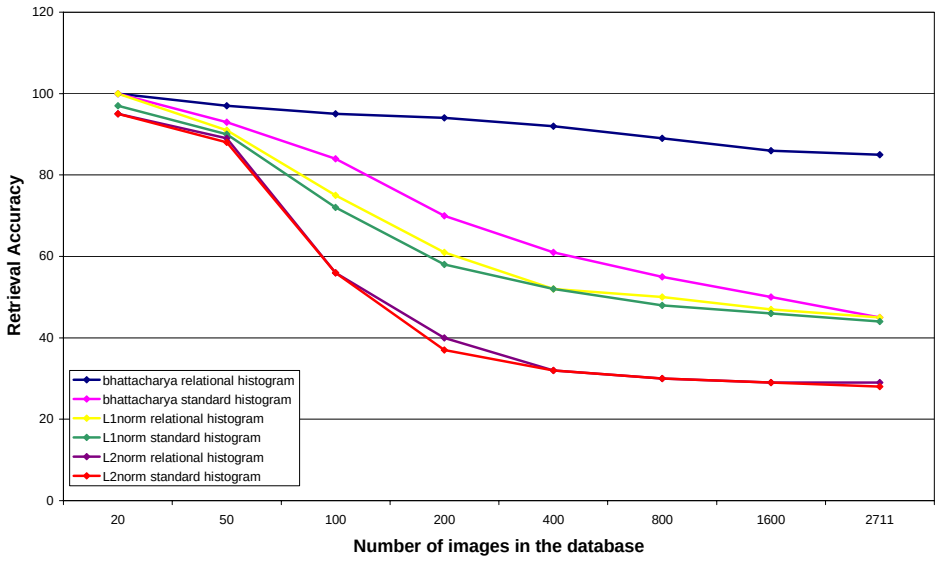



Figure 3.3: Effect of the database size on retrieval accuracy.

marginally better than the L1 norm.

### Graph Order

Having established that the structurally gated histogram outperforms its ungated counterpart when the Bhattacharyya distance is used for the purposes of histogram comparison, our second sensitivity study focuses on the role of relational structure. Here we are concerned with the identification of the most suitable relational structure for efficient and robust shape recognition regardless of the size of the database. The study presented in (Wilson et al., 1996) suggested that nearest neighbour graphs of order six provided the best performance when full relational matching is performed. In this subsection, we aim to examine whether this assertion holds for shape recognition using structurally gated pairwise attribute histograms. Our concern here is to determine the minimum order of the neighbourhood graph that will give the most robust recall when noise and clutter are limiting factors. Since we are dealing with a large number of images and each of these images may contain several hundred extracted segments, it is imperative that the order should be kept as small as

possible.

| 4 neighbours | | 6 neighbours | | 8 neighbours | | 10 neighbours | | N-1 neighbours | |
|---|---|---|---|---|---|---|---|---|---|
| Map60 | 0.053 | Map60 | 0.038 | Map60 | 0.029 | Map60_20.0 | 0.036 | 56-1079.tif | 0.005 |
| Map60_10.0 | 0.059 | Map60_25.0 | 0.038 | Map60_20.0 | 0.029 | Map60 | 0.038 | 56-1165.tif | 0.005 |
| Map60_20.0 | 0.060 | Map60_20.0 | 0.039 | Map60_25.0 | 0.029 | Map60_25.0 | 0.039 | 44-1123.tif | 0.008 |
| Map60_07.0 | 0.062 | Map60_10.0 | 0.041 | Map60_10.0 | 0.034 | Map60_10.0 | 0.040 | Map060_25.0 | 0.008 |
| Map60_25.0 | 0.064 | Map60_07.0 | 0.047 | Map60_07.0 | 0.037 | Map50 | 0.043 | 54-404.tif | 0.008 |
| Map100 | 0.072 | Map170 | 0.050 | Map170 | 0.038 | Map60_07.0 | 0.044 | Map060_20.0 | 0.009 |
| Map170 | 0.075 | 54-1187.tif | 0.052 | Map50 | 0.039 | Map170 | 0.047 | 54-1185.tif | 0.009 |
| 54-1187.tif | 0.075 | Map50 | 0.053 | 54-1187.tif | 0.039 | 54-1187.tif | 0.049 | 56-1301.tif | 0.009 |
| Map200 | 0.078 | Map100 | 0.054 | Map120 | 0.043 | Map70 | 0.052 | 56-1078.tif | 0.009 |
| 56-1121.tif | 0.080 | Map120 | 0.057 | 44-1154.tif | 0.043 | Map100 | 0.054 | 56-1109.tif | 0.009 |
| Map120 | 0.081 | 44-1154.tif | 0.058 | Map100 | 0.044 | Map120 | 0.055 | 46-308.tif | 0.009 |
| Map50 | 0.081 | 56-1209.tif | 0.058 | 56-906.tif | 0.044 | 56-1209.tif | 0.055 | 44-405.tif | 0.009 |
| 56-1209.tif | 0.082 | 44-1155.tif | 0.058 | 54-113.tif | 0.045 | Map200 | 0.056 | 44-1063.tif | 0.009 |

Table 3.2: Effect of the number of N-nearest neighbours on the retrieval performance.

We now aim to establish the optimal order of the N-nearest graph used to construct the histogram. Table 3.2 presents the result of querying the image database using the *Digital Map* for a number of N-nearest neighbour graphs whose order ranges from $N = 4$ to $N = M - 1$ where $M$ is the number of linear segments extracted from the raw image. Each entry shows the code-word assigned to the image together with its Bhattacharyya distance to the query image. The results in this table are ordered from the top-down in order of increasing distance. The top line indicates the image that resembles most closely the query image. The smallest recall distance is obtained when the structure is of order six. In the case when the orders of the graph are six and eight, then the alternative segmentations of *Map60* rank in the top five places of the table and are closely followed by *Map170* in rank six. The right-most column shows the retrieval obtained when all the segment pairs in the images are used to

compute the histograms. This corresponds to removing the structural gating process from the histogram. In this case the highest ranks for the target image are five and seven. In other words, discarding the gating process results in poor recall.

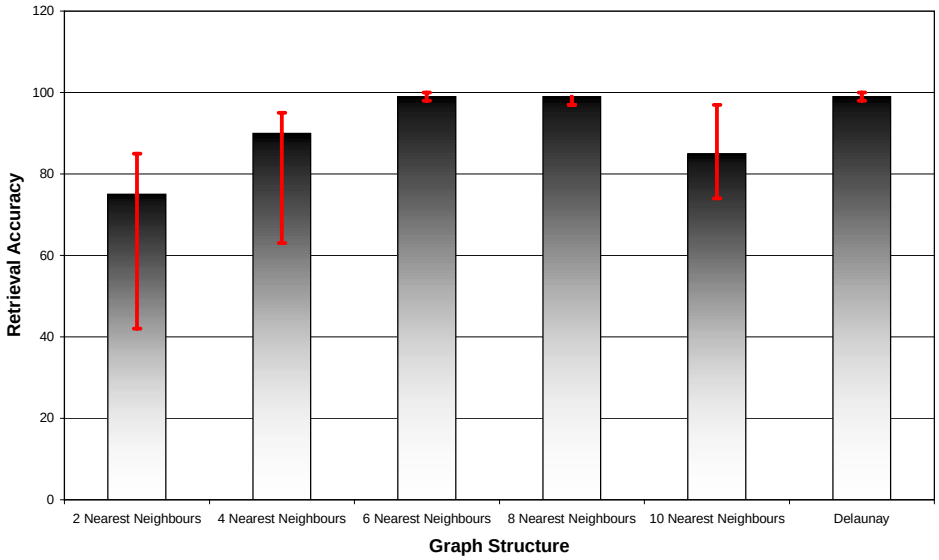



Figure 3.4: Effect of the choice of relational structure on retrieval accuracy.

Figure 3.4 compares the performance of a number of N-nearest neighbour graph with the Delaunay graph. The best performance is achieved with the Delaunay graph and the N-nearest neighbour graph of order 6 or 7. This is an observation of critical importance for our study of line-pattern indexation. The computation of the Delaunay graph is complex and computationally demanding. It is, therefore, not particularly well suited for large image databases. The N-nearest neighbour graph on the other hand can be easily computed inexpensively.

From these results, we can conclude that using a nearest neighbour graph of order six provides the most time efficient and accurate means of line-pattern recall.

### 3.3.4 Recognition Experiments

Having established the most effective recognition strategy, in this subsection we provide some qualitative examples of the results obtained in our experiments. The

results are presented in the form of panels of “thumbnail” images from the database used in our recognition experiments. The “thumbnails” are ordered (from left-to-right and top-to-bottom) according to their distance from the query image. The first experiment involves querying the database with a letter “A”. In this example the database is augmented with 11 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 Figure 3.5, we compare the recognition results obtained with the gated and ungated histograms. In the case of the relationally gated histogram, the first error occurs at rank position 11. Of the 12 “A” in the database the worst recall appears at rank position 14. By contrast, in the case of the ungated histogram, the first error is at rank position 7 and the worst recall is at rank position 47. In the case of the structurally gated representation, it is worth mentioning that letter “V” at rank position 11 and letter “K” at rank position 13 are not severe recognition errors. Indeed, the symbol corresponding to the letter “V” is similar to an “A” rotated by 180 degrees and missing its horizontal bar. Similarly, the letter “K” is closely related to an “A” rotated by 90 degrees with its bar displaced. To some extent a similar explanation can be provided for letter “W” and “Y” ranked at position 15 and 16.

In the second example, we experiment with the “Le Suites Days” logo. The results are shown in figure 3.6. There are four logos in the database that have the same shape, but they carry a different legend (Hotel Days, Auberge Daystop and Auberge Days). In the case of both the relationally gated histogram and its ungated counterpart, the correct logo is correctly recognised in the top ranked position. However, in the case of the relational histogram, the three logos with the same shape but different legends are ranked from second to fourth. In the ungated case, these three logos are more dispersed and appear at rank positions 2, 4 and 6.

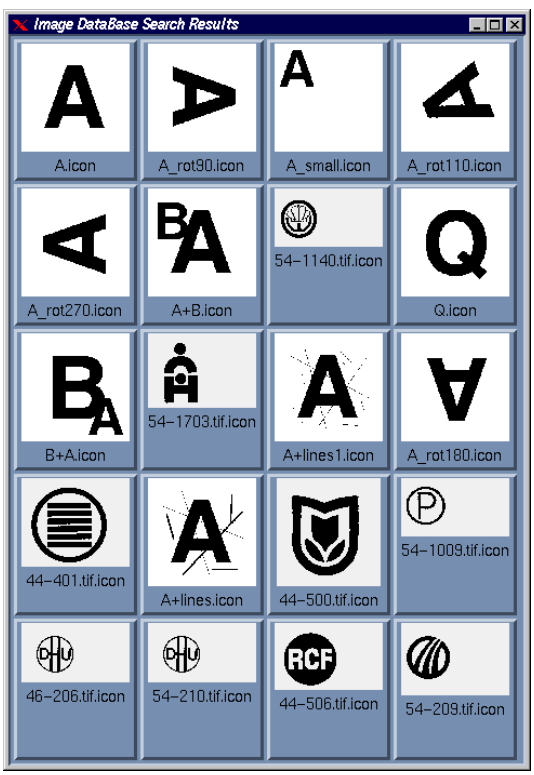
The third example draws on the cartographic section of the database. The line-patterns contained in the cartographic section of the database are segmented from

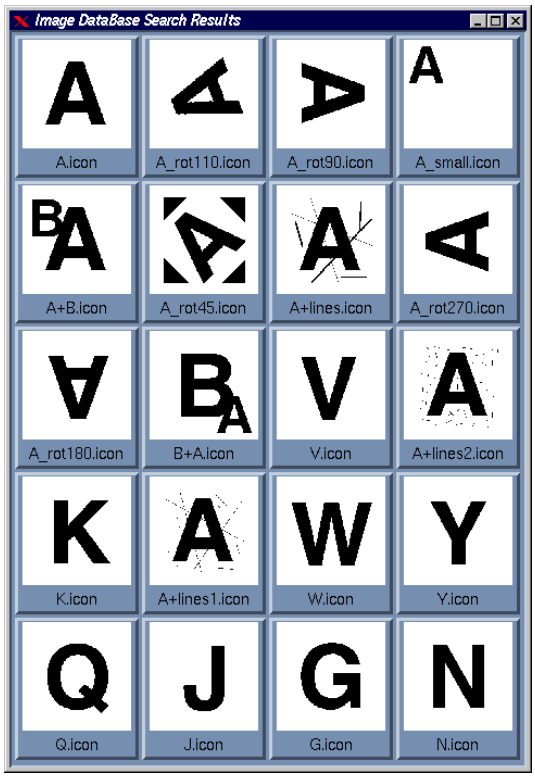
infra-red aerial images of urban and semi-urban areas. The *Digital Map* (or cartographic model) used to query the database represents a road network known to appear in a subset of the images contained within the database. The images labelled *Map060* and *Map170* both contain the road network area depicted in the cartographic model, but are acquired when the imaging apparatus is making different altitude passes over the scene. However, there is known to be a significant barrel distortion in the horizontal direction of the infra-red images caused by the rotating mirror optics underlying the line-scan process used during sensing. Additionally, the viewing angle, altitude and direction is not similar to the *Digital Map*. Ideally the result of a database query using the *Digital Map* depicted in figure 3.7(a) is a retrieval where both the images *Map060* and *Map170* are recalled as the first and second best matches.

The ranked "thumbnail" images are compared for the relationally gated and ungated histograms in Figure 3.7. In the case of the relational histograms, the five different segmentations of the target image (*Map060*) are top-ranked (i.e.*Map060_07* etc). The outcome of controlled over and under segmentation of the image results in significant variation in the number of segments used to construct the histogram representation. The retrieval is not specifically affected by these segmentation systematics as it can be seen in figure 3.7. The high altitude image of the same area (i.e. *Map060*) appears at rank 6.

In the case of the ungated histogram the different segmentations of the target image are badly dispersed. Four of the five different segmentations appear at rank positions 4, 6, 12 and 19. In other words, the recognition performance is very poor.

(a)




(b) (c)

Figure 3.5: The result of querying the database with the letter “A” (a): The left-hand panel (b) is the result obtained with the ungated histogram. The right-hand panel (c) is the result obtained when a relational histogram is used. The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image, which is shown at the top of the figure.

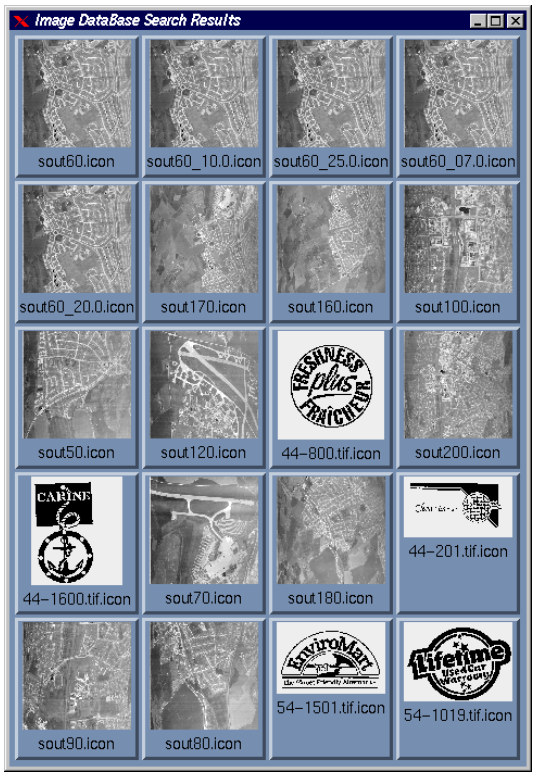

(a)



(b) (c)

Figure 3.6: The result of querying the database with the “Le Suites Days” logo (a): The left-hand panel (b) is the result obtained with the ungated histogram. The right-hand panel (c) is the result obtained when a relational histogram is used. The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image, which is shown at the top of the figure.

(a)


(b) (c)

Figure 3.7: The result of querying the database with the digital map (a): The left-hand panel (b) is the result obtained with the ungated histogram. The right-hand panel (c) is the result obtained when a relational histogram is used. The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image, which is shown at the top of the figure.

## 3.4 Sensitivity Study

The aim in this section is to investigate the sensitivity of the two recognition strategies 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 been generated by randomly sampling the distribution for the existing image-segments.
- **Missing lines:** Here we have deleted a known fraction of line-segments at random locations. This has a similar effect on the resulting line-pattern than occlusion of object parts.
- **Split lines:** Here a predefined fraction of lines have been split into two segments. The splitting process is realised 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 end-point 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 sensitivity analysis is the retrieval accuracy. This is computed as described in Chapter 2 Section 2.5.

In our first experiment, we query the database with a line pattern that is known to have a unique exact counterpart. In other words the line pattern used for indexing

is present in the database. However a number of lines from this object have suffered segmentation errors. Figures 3.8 and 3.9 compare the retrieval accuracy as a function of the fraction of lines that are subjected to segmentation errors. In the case of the standard pairwise geometric histogram (Figure 3.8) the performance starts to degrade as soon as a small amount of segmentation errors is introduced. The most destructive types of error are line segment splitting and the addition of extra lines. Splitting line-segments has the effect of increasing the level of binning of the existing peaks. It is also likely to create a large number of histogram entries in the bin corresponding to collinear line-segments. The addition of extra-lines has the effect of saturating the histogram background. This is particularly important in the case of the standard (ungated) histogram representation since each extra line-segment will generate as many histogram entries as line-segment pairs present in the line pattern. The method is significantly less sensitive to missing lines and segment end point errors. Both of these effects simply reduce the histogram contents rather than distorting its shape. It is, however, worth pointing out that the retrieval performance in the presence of combined errors, situation most likely to simulate real world scenarios, is rather poor with only 5% accuracy for 20% of the lines affected.

In the case of the structurally gated pairwise geometric histogram representation the retrieval performance (see figure 3.9) shows some great improvement over the ungated version. Extra line-segments are again the most destructive errors. However, it is only after over 20% of the line-segments have been corrupted that the accuracy becomes non-optimal. The least destructive type of error is line segment splitting. This is followed by line deletion. When all types of error are combined in equal amount the retrieval performance is very good until the fraction of line-segments subjected to errors becomes greater than 50%. This is a major improvement over the results obtained using the ungated version of the pairwise geometric histogram.

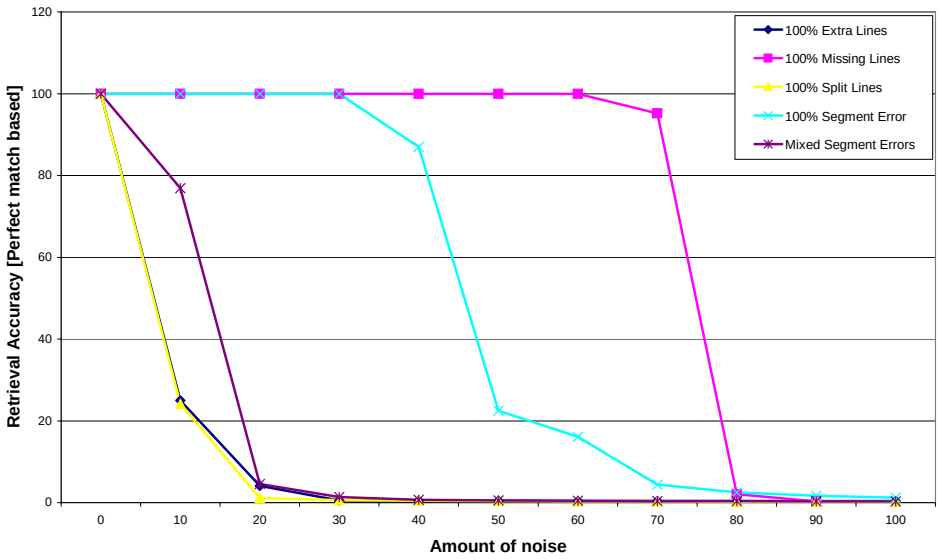



Figure 3.8: Effect of various kinds of noise on the retrieval performance using the standard pairwise geometric histogram representation. The query is a noisy version of a unique target in the database.

We have repeated these experiments with an inexact query. Here the query pattern is a distorted version of the target in the database. An example is furnished by the digital map (see figure 3.7(a)), which is a barrel-distorted version of the target. Figures 3.10 and 3.11 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 standard (ungated) histogram (Figure 3.10), the retrieval performances are very poor indeed, dropping down to just a few percent accuracy whatever the type of error. This is attributable to the fact that the large number of contribution saturates the histogram. In the case of the relational histogram (Figure 3.11) the performances are more acceptable. Split lines and extra lines are again the first to affect the representation. It is interesting to note that under combined errors, a retrieval accuracy of 87% is still achievable when 40% of the line-segments have suffered perturbation.

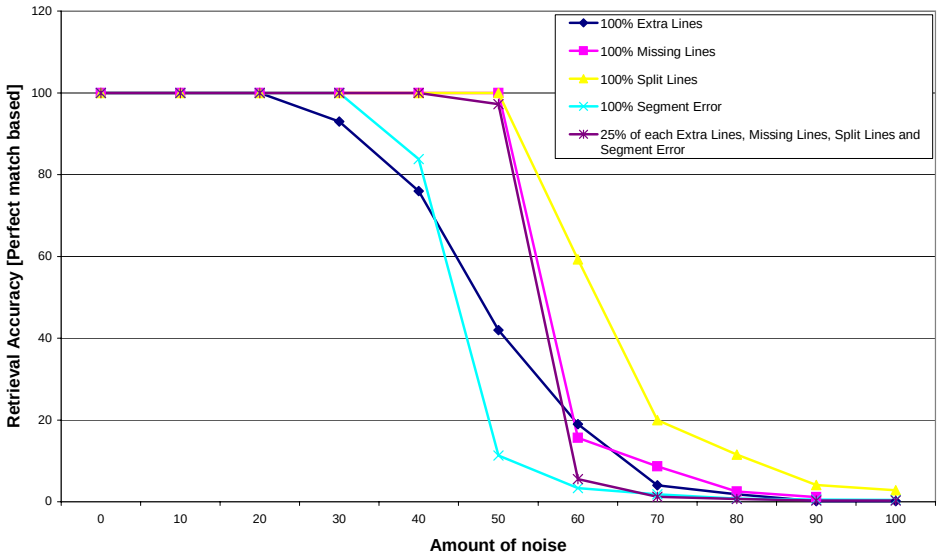



Figure 3.9: Effect of various kinds of noise on the retrieval performance using the relational pairwise geometric histogram representation. The query is a noisy version of a unique target in the database.

To continue the sensitivity study, we focus more closely on the role of segment end-point errors. The reason for this is that such errors will effect the accuracy of the relational measurements. Figure 3.12 shows the average absolute error in the relative angle attribute. This quantity is plotted as a joint function of the fraction of lines affected by errors and the standard-deviation of the Gaussian position error. The main feature to note from this plot is that the angle error increases with both the fraction of effected lines and the variance of the positional errors.

Figure 3.13 shows the effect of line end-point position errors for exact queries using the ungated version of the pairwise geometric histogram. Figure 3.13 shows its structurally gated counterpart. The different curves in the plots correspond to different values of the standard deviation of the end-point position errors. They show the accuracy of retrieval as a function of the fraction of lines affected by end-point errors. As the standard deviation of the position error increases, then so does the fraction of corrupt lines. It appears that perfect recall becomes gradually more

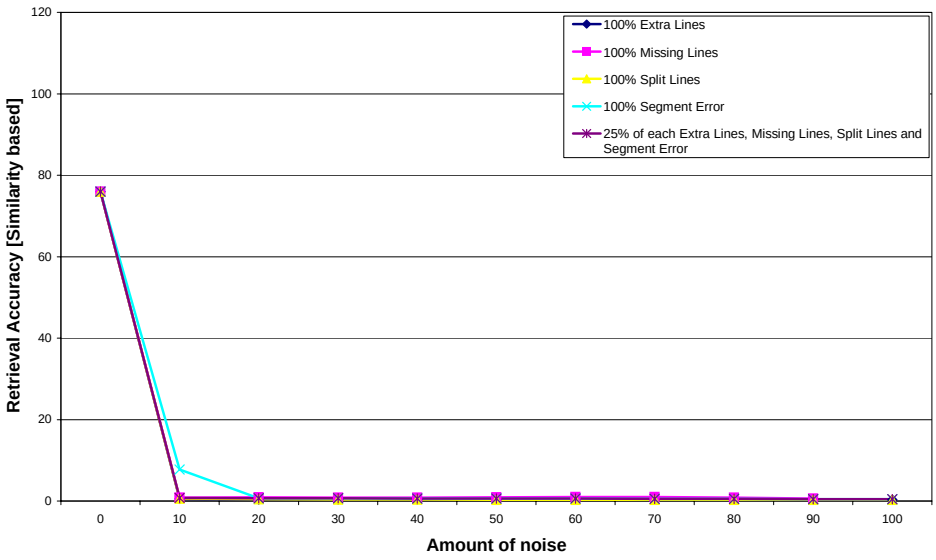



Figure 3.10: Effect of various kinds of noise on the retrieval performance using the standard pairwise geometric histogram representation. The target and the query are similar but not necessarily identical.

difficult as the error in end-point position increases and the fraction of ”correct” lines decreases. The main point to note from these plots is that the performance of the relational histogram method degrades less rapidly under line end-point errors than the standard histogram.

Finally, we turn our attention to the level of pruning provided by the gated pairwise geometric histogram. The idea is to use the relational histogram as a filter that can be applied to the database to limit the search. Once this is done more computationally demanding techniques can be used for refinement. The important issue is therefore the rank threshold that can be applied to the histogram similarity measure. The threshold should be such that the probability of false rejection is low while the number of images that remain to be verified is small.

To address this question we have conducted the following experiment using the database described in Chapter 2. The database contains several groups of images, which are variations of the same object. Some groups contain just a few variations

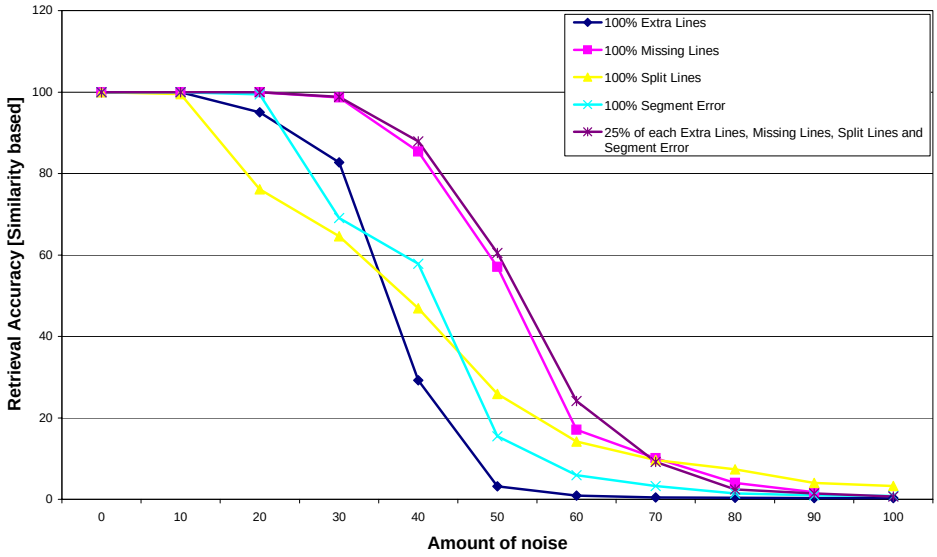



Figure 3.11: Effect of various kinds of noise on the retrieval performance using the relational pairwise geometric histogram representation. The target and the query are similar but not necessarily identical.

while others have up to 50 variations (including rotation and scale). In figure 3.15 we show the result of querying the database with an object selected from each group. The plot shows the worst ranked member of the group as a function of the amount of added image noise. The plot shows a different curve for each of the five different noise types listed above. The main conclusion to be drawn from this plot is that additional lines and end-point segment errors have the most disruptive effect on the ordering of the rankings. However, provided that less than 20% of the line-segments are subject to error, then the database can be pruned to 1% of its original size using the relational histogram comparison. If a target pruning rate of 25% of the database population is desired then the noise-level can be as high as 75%. This result indicates that it is possible to prune the database to a more manageable size. The lower the threshold, the more selective the method becomes. Only limited corruption is allowed to ensure that all the related line patterns are selected for further processing.

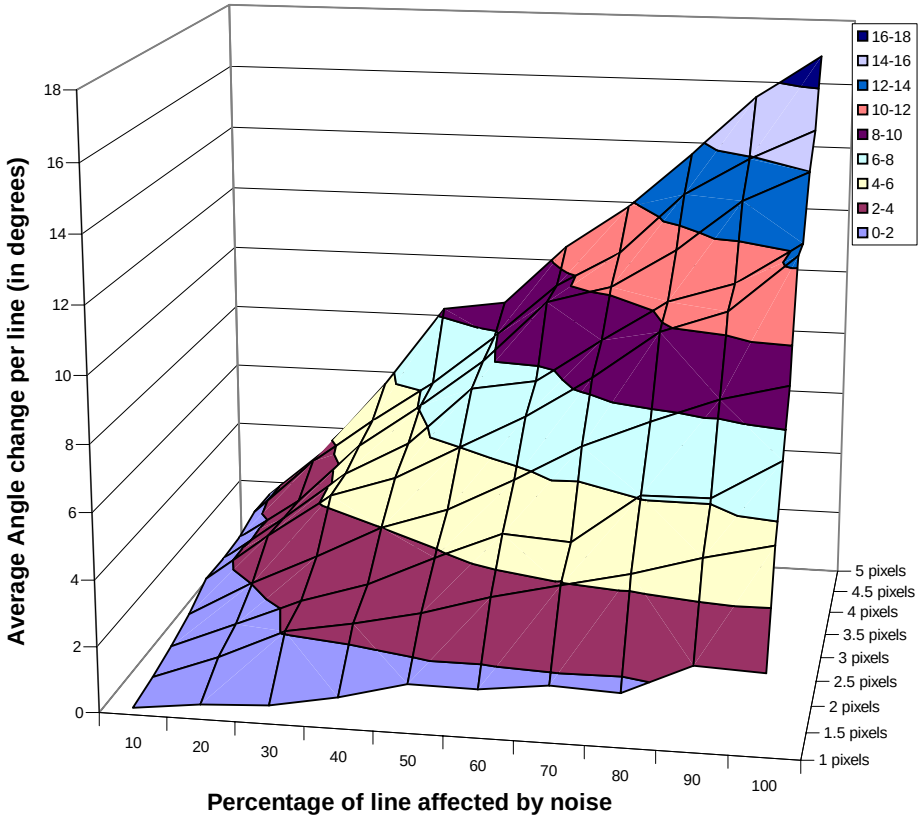



Figure 3.12: Effect of introducing end-point errors on the line-segment orientation.

## 3.5 Summary

The main contribution of this chapter has been to describe a simple and compact histogram representation, which combines both geometrical and structural information about line-patterns. This representation can be used effectively to index into a large database according to shape and line-pattern similarity. The best retrieval performance is obtained when the Bhattacharyya distance measure is used to compare structurally gated pairwise geometric histograms. For a database of over 2500 line-patterns, recall accuracy of over 85% is achievable. Moreover, we show that optimal performance is obtained when the nearest neighbour graph used to gate the histogram is of order six. There are also indications that the method is relatively robust to the under and over segmentation of the line-patterns.

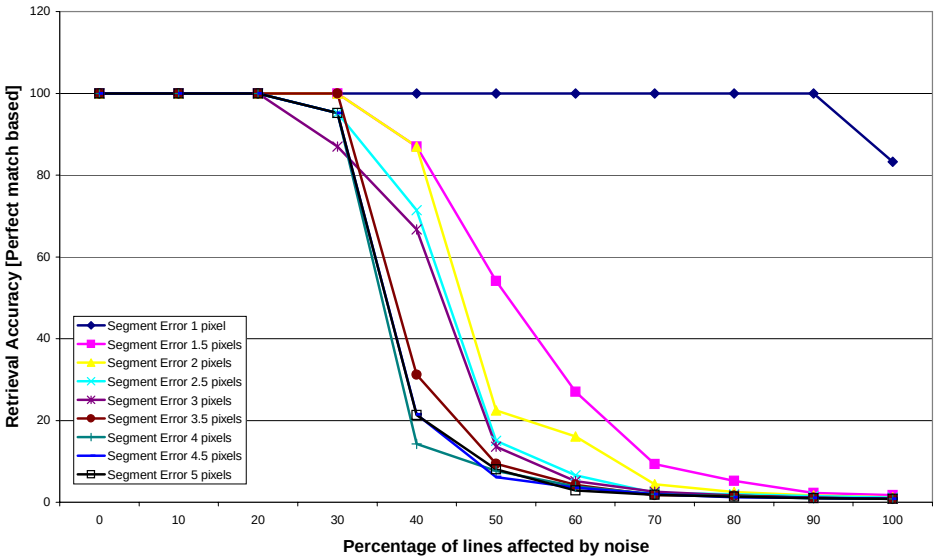



Figure 3.13: Effect of introducing segment errors on the retrieval performance using the standard pairwise geometric histogram representation. The query is a noisy version of a unique target in the database.

Based on a thorough sensitivity study the effect of a number of segmental error types on the histogram representation has been established. Similarly, our experiments show that 75% reduction of the line-pattern candidates may be performed while retaining all those models, which are relevant for the next level of recognition.

The work presented in this chapter may be regarded as an important step in a more ambitious programme aimed at developing a hierarchical recognition system for large image database. Our next goal is to refine the rather crude index of structural similarity. In particular, we aim to derive fine-grain similarity measures from the recently reported Bayesian framework for relational graph matching of Wilson and Hancock (Wilson and Hancock, 1997) and the Hausdorff distance measure proposed by Huttenlocker *et al.* (Huttenlocher et al., 1993).

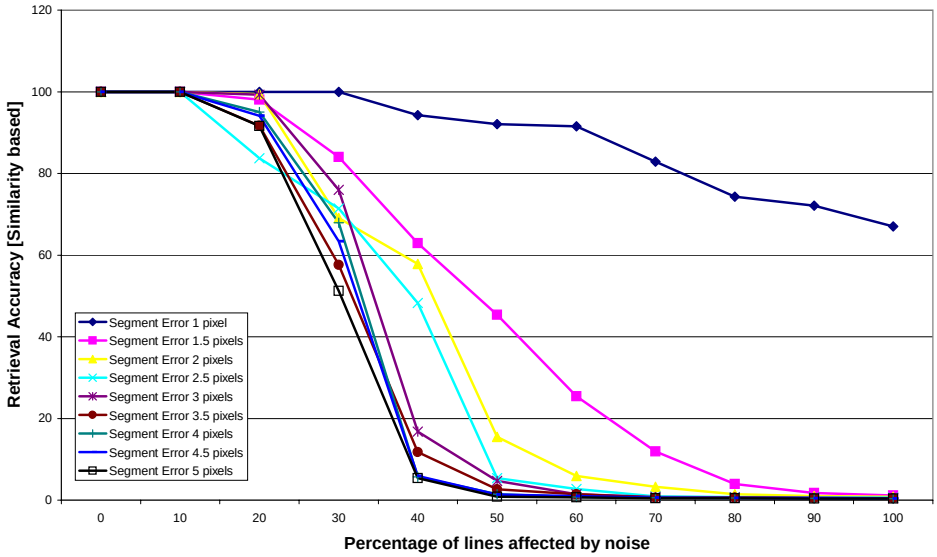



Figure 3.14: Effect of introducing segment errors on the retrieval performance using relational pairwise geometric geometric representation. The query is a noisy version of a unique target in the database.

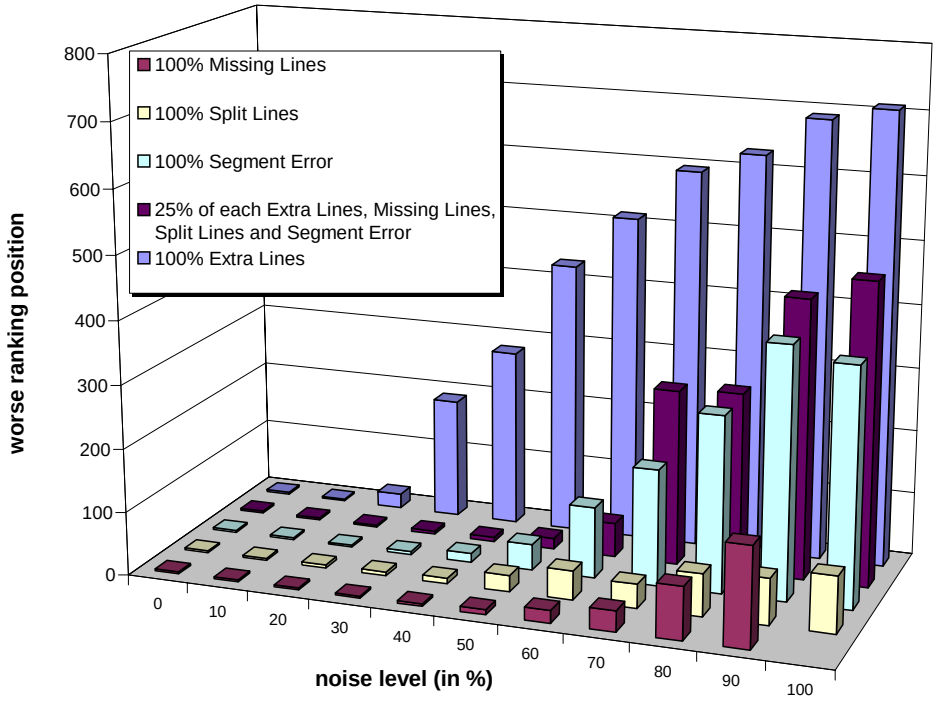



Figure 3.15: Retrieval position of the last member of an image class when subjected to various types and degree of noise.

# Chapter 4

# Feature Based Representation and Retrieval

In the previous chapter we have shown that representing image shapes using relational histograms of pairwise attributes can be very effective for image database indexing when coupled with the correct distance measures. However, because of the very nature of histograms it is possible for two completely different line-patterns to have identical representation. Such a situation would lead to incorrect classification. In order to solve for this problem and refine the quality of the recognition a new similarity measure for object recognition from large libraries of line-patterns based on local image feature has been devised and is presented in this chapter. The measure commences from a Bayesian consistency criterion, which has been developed for locating correspondence matches between attributed relational graphs using iterative relaxation operations. The aim in this paper is to simplify the consistency measure so that it may be used in a non-iterative manner without the need to compute explicit correspondence matches. This considerably reduces the computational overheads and renders the consistency measure suitable for large-scale object recognition. The measure uses robust error-kernels to gauge the similarity of pairwise attribute relations defined on the edges of nearest neighbour graphs. We use the similarity

measure along with some alternatives in a number of recognition experiments. 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 a Gaussian kernel or Huber's robust kernel is used to weight the attribute relations. Moreover, the method consistently outperforms Rucklidge's median Hausdorff distance.

## 4.1 Related literature and Motivation

Our aim in this chapter is to provide a method to refine the comparison of the pairwise geometric attributes described in Chapter 2. Ideally, the distance measure should have a degree of robustness to outliers. The issue of how to compare representations has received less attention than the problem of efficient object representation. One exception is the recent work of Rucklidge (Rucklidge, 1995) which has shown how the Hausdorff distance can be used for relatively robust object recognition and location.

Despite offering an interesting and effective strategy for comparing image representations, there are a number of criticisms that can be raised concerning the use of the Hausdorff distance. 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 outliers, 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.

In this chapter, we address the two issues of object similarity in a more critical manner. Our first observation is that since the object primitives under study are subject to both measurement uncertainty and segmentation error, fuzzy or probabilistic

distance measure may be more appropriate to the comparison task. In particular, recent interest in the matching of relational graphs has furnished methodology for exploiting contextual constraints. However, this methodology has not been exploited in large-scale object recognition tasks.

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 representation. In Chapter 3, we have provided a tangible example of how such a set of candidates can be located by comparing structurally gated pairwise geometric histograms. Once overall object recognition has been achieved then detailed correspondences may be recovered. Here, techniques such as graph-matching (Wilson and Hancock, 1997) and pose estimation (Cross and Hancock, 1998b) can be used to verify recognition hypotheses and initiate new searches if necessary. This final step of the retrieval process will be addressed in Chapter 5.

## 4.2 Object Representation

Our representation has already been successfully exploited in histogram-based object retrieval as described in Chapter 3. Rather than using a global histogram, we represent each line-segment pair by a vector of pairwise geometric attributes. Accordingly, each segment pair (i.e.: $(x_i, x_j)$) relationship will be represented by a two dimensional vector $P_{i,j}$ with

$$P_{i,j} = (\theta_{x_i,x_j}, \vartheta_{x_i,x_j})^T$$

where $\theta_{x_i,x_j}$ is the directed relative angle and $\vartheta_{x_i,x_j}$ is the directed relative position.

We aim to augment the pairwise attributes with constraints provided by the edge-set of the N-nearest neighbour graph. The conventional Hausdorff distance explores the complete set of associations between the set of tokens constituting the model and the data. Here our aim is to limit the set of associations to those that are

consistent with the local structure of the neighbourhood graph. The motivation here is that local object representations are more robust to occlusion, missing and extra features and noise.

Our idea contrasts with a number of related contributions in the literature. For instance, Evans *et al* (Evans et al., 1993) effectively have a local representation of line-pattern structure which is less compact than ours since it employs one histogram per line-segment. Moreover, their attributes are not scale invariant. The idea of using a region-of-interest is close to that of using a neighbourhood graph. However, their regions are controlled by a scale parameter. By virtue of their structural character, neighbourhood graphs are scale-invariant. Additionally, using segment neighbour as opposed to the connected segment approach of Stein and Medioni (Stein and Medioni, 1990) offers greater robustness to segmental clutter. It also allows the representation to be much more compact since a single description of each line-pattern is required.

We represent the sets of line-patterns as 4-tuples of the form $G = (V, E, U, B)$. Here the line-segments extracted from an image are indexed by the set $V$. More formally, the set $V$ represents the nodes of our nearest neighbourhood graph. The edge-set of this graph $E \subset V \times V$ is constructed as follows. For each node in turn, we create an edge to the $N$ line-segments that have the closest distances. Associated with the nodes and edges of the N-nearest neighbour graph are unary and binary attributes. The unary attributes are defined on the nodes of the graph and are represented by the set $U = \{(\phi_i, l_i); i \in V\}$. Specifically, the attributes are the line-orientation $\phi_i$ and the line-length and $l_i$. By contrast, the binary attributes are defined over the edge-set of the graph. The attribute set $B = \{(\theta_{i,j}, \vartheta_{i,j}; (i, j) \in E \subseteq V \times V\}$ consists of the set of pairwise geometric attributes for line-pairs connected by an edge in the N-nearest neighbour graph.

We are concerned with attempting to recognise a single line-pattern $G_m = (V_m, E_m, U_m, B_m)$, or model, in a database of possible alternatives. The alternative

data-patterns are denoted by $G_d = (V_d, E_d, U_d, B_d), \forall d \in \mathcal{D}$ where $\mathcal{D}$ is the index-set of the database.

## 4.3 Pairwise Attribute Consistency

The aim in this work is to draw on recent work on relational graph matching to develop a similarity measure for rapidly comparing relational descriptions of line-patterns, which are represented in the manner outlined in the previous section. Although the framework furnishes a principled Bayesian measure of relational consistency, it has hitherto been used exclusively for graph-matching using iterative relaxation operations. It is, hence, unsuitable for rapid recognition of objects from large object libraries on two counts.

In the first instance, graph matching is concerned with detailed correspondence matching rather than global object recognition. Secondly, since relaxation algorithms are iterative in nature, they are too computationally demanding to be used when large object-libraries are being considered. The aim here is to provide a simplified relational consistency measure, which can be used, for recognition without the need to iteratively establish correspondence matches.

### 4.3.1 Global Pattern Similarity

We take as our starting point the weak-context version of the average consistency measure developed for evidence combination by Kittler and Hancock (Kittler and Hancock, 1989). Following Christmas, Kittler and Petrou (Christmas et al., 1995) we measure the compatibility of the graphs being compared using pairwise attribute relations defined on the edges of the nearest-neighbour graph. To be more formal, suppose that the set of nodes connected to the model-graph node $I$ is $C_I^m = \{J|(I,J) \in E_M\}$. The corresponding set of data-graph nodes connected to the node $i$ is $C_i^d = \{j|(i,j) \in E_d\}$. With these ingredients, the consistency criterion,

which combines evidence for the match of the graph $G_m$ onto $G_d$ is

$$Q(G_d, G_m) = \frac{1}{|V_M| \times |V_d|} \sum_{i \in V_d} \sum_{I \in V_m} \frac{1}{|C_i^d|} \sum_{j \in C_i^d} \frac{1}{|C_I^m|} \sum_{J \in C_I^m} P\left((i,j) \rightarrow (I,J) | \mathbf{v}_{I,J}^m, \mathbf{v}_{i,j}^d\right) \tag{4.1}$$

The probabilistic ingredients of the evidence combining formula need further explanation. The *a posteriori* probability $P\left((i,j) \rightarrow (I,J) | \mathbf{v}_{I,J}^m, \mathbf{v}_{i,j}^d\right)$ 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 $v_{I,J}^m$ and $v_{i,j}^d$.

We would like to use the Bayesian consistency criterion as the basis of a similarity measure for graph-based object recognition. To commence this development, we consider a very simple form for the structural error process. We assume that the conditional prior can be modelled as follows

$$P\left((i,j) \rightarrow (I,J) | \mathbf{v}_{I,J}^m, \mathbf{v}_{i,j}^d\right) = \Gamma_\sigma(||\mathbf{v}_{I,J}^m - \mathbf{v}_{i,j}^d||) \tag{4.2}$$

where $\Gamma_\sigma(||\mathbf{u}_I^m - \mathbf{u}_i^d||)$ is a distance weighting function.

We now consider how to simplify the computation of relational consistency. 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 compatibility. With these restrictions, the process of maximising the Bayesian consistency measure is equivalent of maximising the following relational-similarity measure

$$Q(G_d, G_m) = \sum_{i \in V_d} \max_{I \in V_m} \sum_{j \in C_i^d} \max_{J \in C_I^m} \left(\Gamma_\sigma(||\mathbf{v}_{I,J}^m - \mathbf{v}_{i,j}^d||)\right) \tag{4.3}$$

With the similarity measure to-hand, the best matched line pattern is the one which satisfies the condition

$$Q(G_d, G_m) = \arg\max_{d' \in \mathcal{D}} Q(G'_d, G_m) \tag{4.4}$$

### 4.3.2 Robust Weighting Kernels

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

$$\Gamma_\sigma(\rho) = \exp\left(-\frac{1}{2}\frac{\rho^2}{\sigma^2}\right)$$

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

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

Stated in this way, the recognition metric has much in common with the graph-matching criterion recently reported by Wilson and Hancock (Wilson and Hancock, 1997). However, rather than being used for primitive-by-primitive correspondence matching, in the work reported here we use the criterion for recognising primitive ensembles.

## 4.4 Feature Set Comparison

In this section, we consider how the Hausdorff distance can be used for comparing relational representations.

### 4.4.1 Hausdorff distance

The idea underpinning the 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 mean of avoiding the computationally demanding problem of attempting to find correspondence matches between individual primitives whilst performing recognition.

The distance is computed by exploring the entire space of possible model-data associations between two sets of unstructured measurement vectors. The metric gauges the distance between the two sets of observations using the maximum value of the minimum pairwise data associations. More formally, with the graph-based notation introduced in Section 4.2, the distance is defined to be

$$H_U(G_d, G_m) = \max_{i \in V_d} \min_{I \in V_m} ||\mathbf{u}_I^m - \mathbf{u}_i^d||$$

where $||\mathbf{u}_I^m - \mathbf{u}_i^d||$ is the distance-norm between the unary measurement vectors. Here we will consider several alternative distance norms. The first of these is the familiar L2 norm, the second is the Gaussian weighting function while the others are the robust statistic weighting kernels listed in Section 4.3.2. The measure can be extended to pairwise attributes in a straightforward manner by exploring the set of possible pairwise associations. In the case of pairwise attributes, the Hausdorff distance is given by

$$H_B(G_d, G_m) = \max_{(i,j) \in V_d \times V_d} \min_{(I,J) \in V_m \times V_m} ||\mathbf{y}_{(I,J)}^m - \mathbf{y}_{(i,j)}^d||$$

From the computational standpoint, this represents an increase in the number of associations that have to be compared. Whereas in the unary case there are $|V_m| \times |V_d|$ comparisons, in the pairwise case there are $|V_m|^2 \times |V_d|^2$ comparisons. Moreover, if

recognition is being attempted then the large number of possible pairwise associations is likely to render the object representation highly ambiguous.

One way of overcoming these problems and 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. With this goal in mind we can redefine the Hausdorff distance over the edge-sets of the model and data graphs. The Hausdorff distance needs to be augmented by new terms in order to capture both the pairwise geometric attribute level and the relational level of the comparison. One could compare our novel distance measure as two levels (or dual step) Hausdorff distance. The basic idea behind the Hausdorff distance is to compare two sets of points each extracted from an image in order to find the best match for each of those points. In our case, the underlying representation is not computed from image points measurements but from line-segments pairwise attributes. Additionally, and this is the reason for modifying the standard Hausdorff distance, the local relational information between line-segment pairs is retained within the representation by the mean of the N-nearest neighbour graph. The lower level of the modified Hausdorff distance (rightmost hand $\max\min$ operation) performs the node to node comparison by looking at pairwise attributes between graph nodes. The higher level (leftmost $\max\min$), on the other hand, computes the global correspondences between the data and the model graph by looking at the nodes previously compared. The modified object-distance is

$$H_G(G_d, G_m) = \max_{i \in V_d} \min_{I \in V_m} \max_{j \in C_i^d} \min_{J \in C_I^m} ||\mathbf{v}_{(I,J)}^m - \mathbf{v}_{(i,j)}^d||$$

Rucklidge (Rucklidge, 1995) has reported a further modification of the standard Hausdorff distance which produces tangible performance improvements. His idea is to replace the $\max$ operator by an operator that selects from the set of attribute distances using a median test or $f$-th quantile test. This version of the Hausdorff distance can be written as

$$H_G^f(G_d, G_m) = F^f_{i \in V_d} \min_{I \in V_m} F^f_{j \in C_i^d} \left\{ \min_{(J \in C_I^m} ||\mathbf{v}^m_{(I,J)} - \mathbf{v}^d_{(i,j)}|| \right\}$$

where the operator $F^f_{i \in V_d}$ selects the $f$-th quantile value from the set of edge-wise attribute distances.

### 4.4.2 The Fuzzy Hausdorff Distance Measure

Rucklidge's median operator represents a robust procedure for selecting pairwise attribute differences. 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 new distance measure is defined to be

$$H^p(G_d, G_m) = \sum_{i \in V_d} \min_{I \in V_m} \sum_{j \in C_i^d} \min_{J \in C_I^m} \left(1 - \Gamma_\sigma(||\mathbf{v}^m_{I,J} - \mathbf{v}^d_{i,j}||)\right) \tag{4.5}$$

This equation is similar to equation 4.3, which was derived from the consistency measure of Kittler and Hancock (Kittler and Hancock, 1989). This equation may also be seen as an example of fuzzy composite operation that is similar to those described in the review of Bloch (Bloch, 1996) on Fuzzy Operator for computer vision and image processing. This particular kind of fuzzy operator is referred to as context independent constant behaviour operator.

Finally, when recognition is being attempted over a large database of line patterns, the model is taken to associate with the minimum Hausdorff distance set of data. The data item associated with the model is

$$\theta_m = \arg\min_{d \in \mathcal{D}} H(G_d, G_m)$$

## 4.5 Experiments

The practical goal in this work is to incorporate the distance measures defined in the previous section into a hierarchical object recognition system. The overall goal is the recognition of complex line patterns from large databases of alternatives. The architecture that we have in mind is as follows. Recognition commences by comparing a compact shape representation to deliver a set of candidate objects. In our present work this compact object representation is a multidimensional histogram of pairwise Euclidean invariant geometric attributes. The candidates are then subjected to a more detailed comparison based on the attributes of the individual object primitives. It is here that we need a distance measure that captures the closeness and saliency of the pairwise attributes for the line patterns. Once the closest distance pattern has been identified, then detailed verification can be attempted by looking for individual correspondences between the primitives and assessing their relational consistency. Here we envisage using our recently reported framework for graph-matching (Wilson and Hancock, 1997).

We are interested in assessing the first two levels of this architecture. We use our recently reported work on compact line-pattern indexing to deliver a set of candidate patterns. Here the patterns are represented using a two-dimensional pairwise geometric histogram of the directed relative angles and lengths described in Section 2. The contributions to the histogram are gated using relational constraints. In other words, a histogram entry is only made if a line-segment pair is connected by an edge in the nearest neighbour graph (Huet and Hancock, 1998c). We compare the histogram bin contents using the Bhattacharyya distance (Huet and Hancock, 1996a). Based on the ordered set of histogram distances we select the N-best matches for more detailed comparison.

### 4.5.1 Distance Measures and Robust Weighting Kernels

We have conducted our recognition experiments with a database of 2500 line-patterns each containing over a hundred lines as described in Chapter 2 Section 2.3. The line-patterns have been obtained using the methodology described in Chapter 2 Section 2.4.

(a) Digital Map (b) Target 1 (c) Target 2

Figure 4.1: Images from the database.

The recognition task is posed as one of recovering the line-pattern, which most closely resembles a digital map. The original images from which our line-patterns have been extracted have been obtained from a number of diverse sources. However, a subset of the images is aerial infra-red line-scan views of southern England. Two of these infra-red images correspond to different views of the area covered by the digital map (see figure 4.1). These views are obtained when the line-scan device is flying at different altitudes. The line-scan device used to obtain the aerial images introduces severe barrel distortions.

In order to explore the sensitivity of our recognition method to segmentation systematics, we have introduced multiple segmentation of the target images into the database. These different segmentations have been obtained by maliciously adjusting the control parameters of the feature extraction algorithm. In total there are

ten different segmentations for each of the two target images.

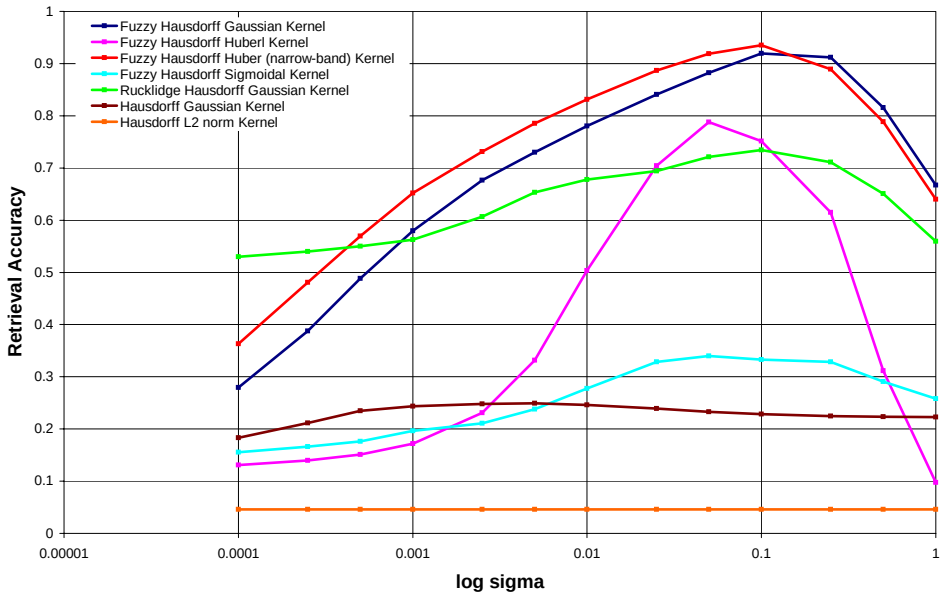



Figure 4.2: Relative recognition performance for various distance measures.

Our first set of experiment aims to illustrate the relative recognition performance of the different distance measures. The performances of the system in terms of retrieval accuracy are assessed using the standard normalised IAVRR / AVRR recall metric (Pentland et al., 1996) which is equal to $1$ for perfect retrieval accuracy.

For this experiment a database composed of 850 line patterns is used and the result shown represents the average retrieval accuracy of 100 distinct queries. Figure 4.2 shows the recognition performance as a function of the control parameter $\sigma$ for each of the distance measures presented in Section 4.3 in turn. From the figure it is clear that the best performance is obtained when the weighting kernel is either Gaussian (black) or a modified narrow-band Huber (red). The poorest performance is obtained with the crisp Hausdorff distance coupled with the $L2$ norm (orange). Rucklidge's modified Hausdorff distance (using median instead of $\max$ comparator (Rucklidge, 1995) and a Gaussian kernel) does not provide an optimal recall performance for this particular task but presents an obvious improvement (green) over

the standard Hausdorff distance. 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 $\sigma$. From this graph it can also be seen that an average correct retrieval rate of 94% is achievable.

### 4.5.2 Structural Sensitivity

In the next set of experiments we illustrate the effect of relational structure on the recognition process. Figure 4.3 shows the recognition accuracy as a function of the width parameter $\sigma$ for a number of different graph-structures. In this experiment

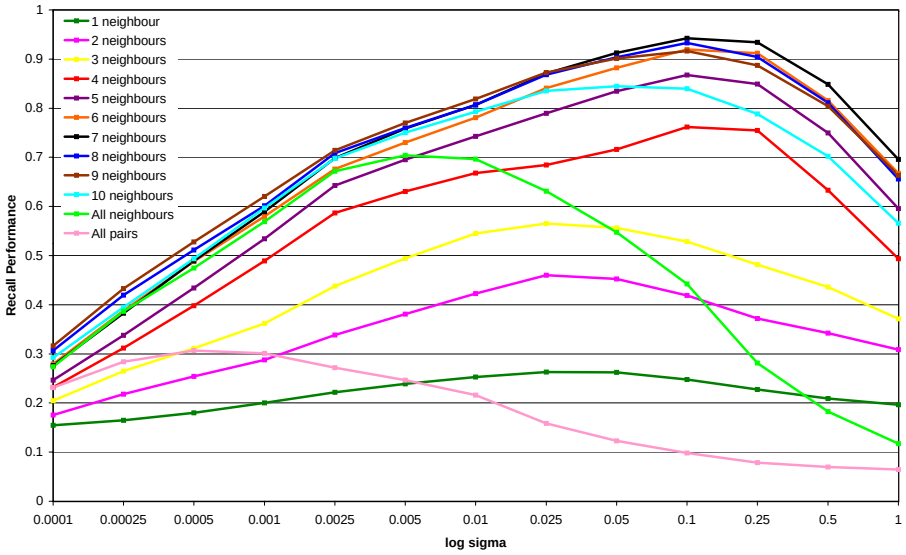



Figure 4.3: Relative recognition performance for various relational structures.

we have used the similarity measure of equation (3) with the pairwise attributes weighted using the Gaussian error-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. In the first example we relax the requirement for neighbourhood structure,

and evaluate the similarity measure over the complete space of edge-wise associations (light green).

In the second example we remove the edge-structure and compute the similarity measure over the complete space of pairwise associations (light pink). The first observation that can be drawn from this set of experiments is that the best recognition performance is obtained when the order of the nearest neighbour graph is seven (black).

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

### 4.5.3 Recognition Experiments

We now provide some examples to illustrate the qualitative ordering that results from the two recognition experiments. Figures 4.4, 4.5 and 4.6 compare the recognition rankings obtained from the database. In each case the left-hand panel is the result of using relational histograms while the right-hand panel is the result of using feature-sets. In each panel the thumbnails are ordered from left-to-right and from top-to-bottom according to decreasing rank. In Figure 4.4 we show an example of querying the database with the letter A. In the case of the feature-sets, the 12 occurrences of the letter A are ranked at the top of the order. It is interesting to note that the noisy versions of the letter are ranked in positions 11 and 12. In the case of the relational histograms the letter A's are more dispersed. The letters K and V disrupt the ordering.

Figure 4.5 shows the result of querying with one of the logos. There are four versions of this logo in the database. Each has the same overall shape, but has a different caption. Both the relational histogram and the feature-sets give the exact query in first position and the three similar logos in the next three positions. However, when feature sets are used, there is a greater tendency to cluster logos with a

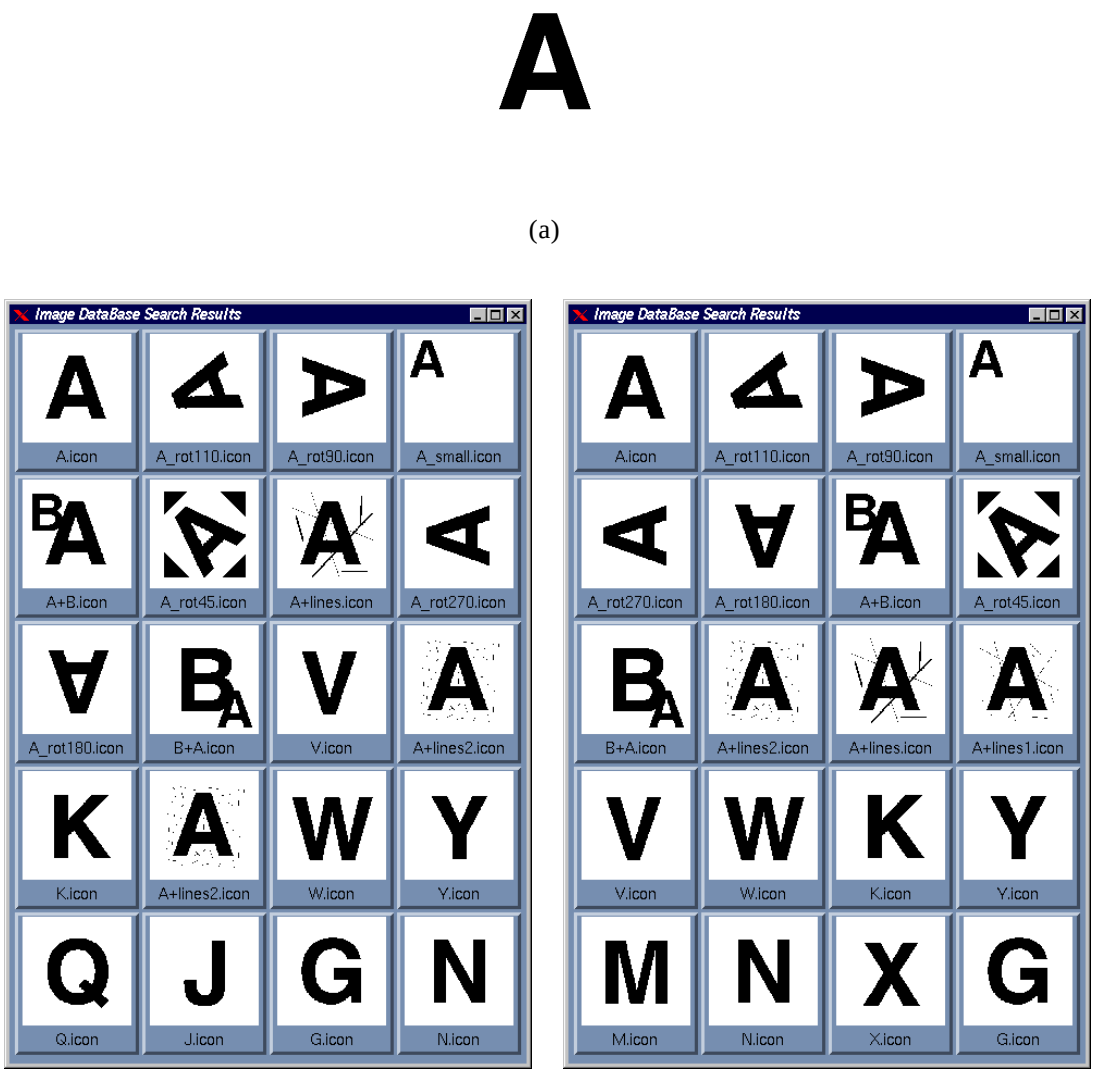



Figure 4.4: The result of querying the database with the letter “A”: The left-hand panel is the result obtained with the relational histogram. The right-hand panel is the result obtained when feature-sets are used. The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image, which is shown at the top of the figure.

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, Figure 4.6 shows the result of querying the database with the digital map. In the case of the feature-sets, the eight segmentations of the two images containing the road-pattern are recalled in the top-ranked positions. In the case of the relational histogram, five of the segmentations are top-ranked. Another segmentation is ranked ninth and one segmentation falls outside the top 16.

In the next sequence of experiments, we investigate the interplay between the histogram and the relational similarity measure for a number of queries.

Figure 4.7 shows that the retrieval accuracy using the coarse-grained two dimensional pairwise geometric histogram distance (y-axis) is improved when coupled with the more refined relational distance.

The final set of experiments focuses on the distribution of the distance measures. We have extracted from the database the 1000 best histogram matches for the digital map query and have used these for more detailed recognition experiments. Figure 4.8 compares histograms of the various distance measures for the target images (red) and for the remaining line-patterns (blue). Figure 4.8(a) shows the distribution of Bhattacharyya distance between the histograms in the database and the query image. The remaining three plots show the distribution of the primitive-based distance measures. Figure 4.8(b) shows the distribution of Bayesian similarity measure, Figure 4.8(c) shows the distribution of standard Hausdorff distance while Figure 4.8(d) shows the distribution of Rucklidge’s median-based Hausdorff distance. The main feature to note from these distributions is that the Bayesian measure gives the best separation between the various segmentations of the two aerial images (shown in red) and the remainder of the database (shown in blue). In this case the ten different aerial image segmentations are ranked above the rest of the database. In other words, there is no overlap with the remaining patterns in the database. In each of

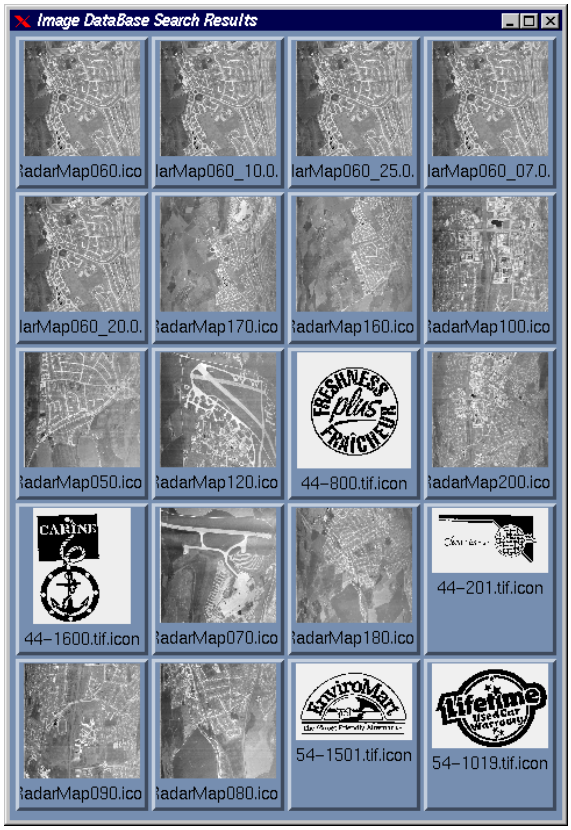

(a)

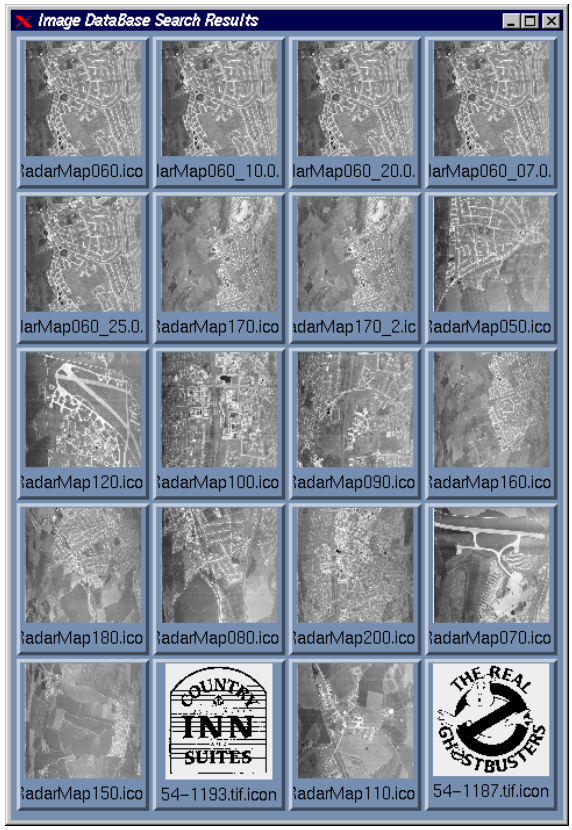


(b) (c)

Figure 4.5: The result of querying the database with the “Le Suites Days” logo: The left-hand panel is the result obtained with the relational histogram. The right-hand panel is the result obtained when feature-sets are used. The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image, which is shown at the top of the figure.

(a)




(b) (c)

Figure 4.6: The result of querying the database with the digital map (a): (b) is the result obtained with a relational histogram. (c) is the result obtained when the fuzzy hausdorff distance measure is used. The images are ordered from left-to-right and top-to-bottom in increasing distance from the query image.

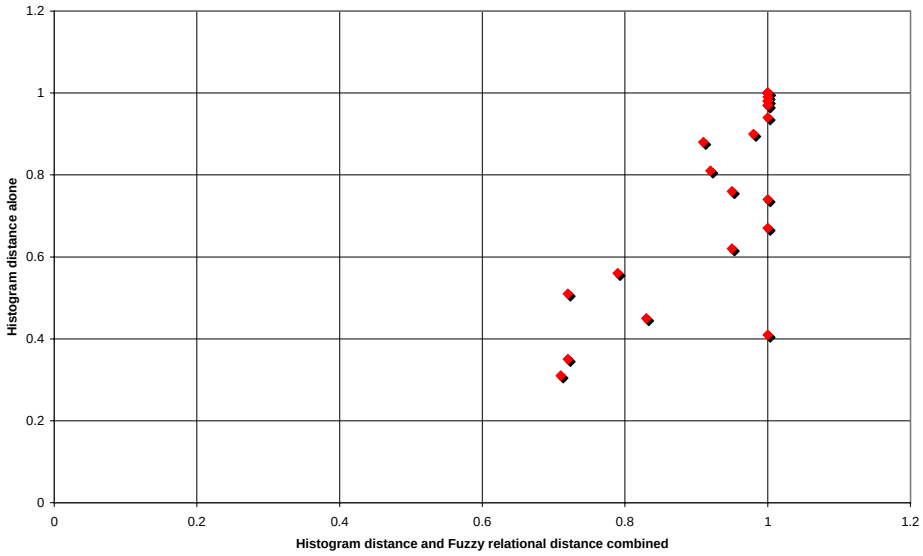
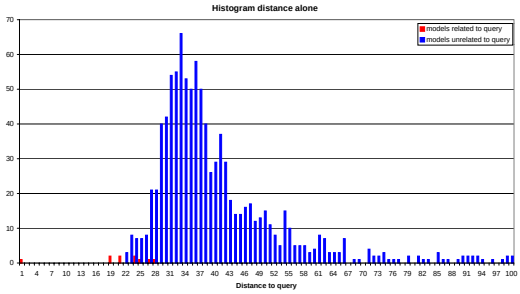
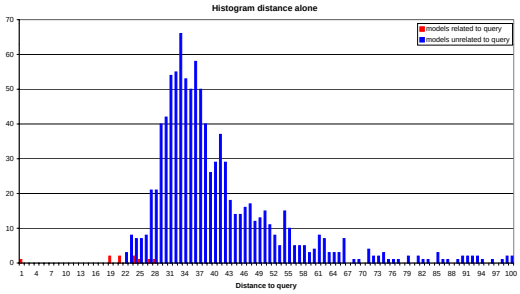



Figure 4.7: Interplay between the histogram and the fuzzy relational distance.

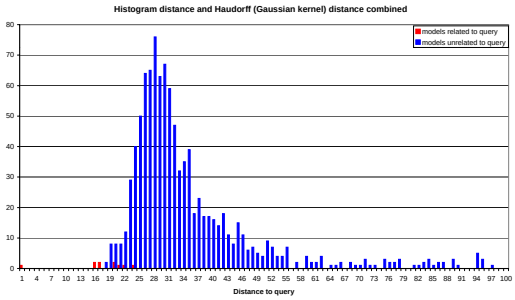
the remaining three cases (i.e. Figures 4.8(a), (c) and (d)) only five of the ten segmentations are top ranked. In other words, the primitive-based Hausdorff distances perform only as well as histogram-based comparison that overlooks the primitive structure of the line-patterns. Moreover, the Bayesian recognition process does not appear to be sensitive to the segmentation and polygonisation process used to extract the line-patterns from the two aerial images.



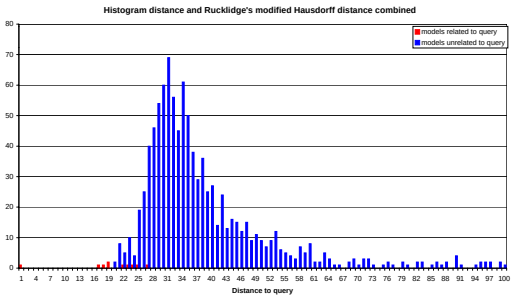

(a)




(b)




(c)




(d)

Figure 4.8: Distribution of the distance measures during retrieval using the digital map.

## 4.6 Sensitivity Analysis

Having established the most effective distance measure, i.e. the use of the so-called fuzzy Hausdorff distance in conjunction with the Huber kernel, we turn to measuring the sensitivity of the recognition scheme. The analysis focuses on the sensitivity to the systematics of the line-segmentation process. We repeat the sensitivity analysis described in Chapter 3 Section 3.4.

We have computed two performance measures for the algorithm under the different segment errors described above. The first of these is retrieval accuracy. This is the fraction of queries that return a correct recognition as described in Chapter 2 Section 2.5. The second measure is the average distance between the query and the target image.

In our first experiment, we query the database with a line pattern that is known to have a single precise counterpart. Figure 4.9 shows the retrieval accuracy as a function of the fraction of lines that are subjected to segmentation errors. From the plot it is clear that only line-deletion has any significant effect upon recall. Moreover, it is not until 75% of the lines are deleted that there is any significant effect on the retrieval accuracy. Turning our attention to the average distance, in Figure 4.10 we see that this poorer robustness to line-segment deletions is reflected by the fact that the average pattern similarity drops off more quickly with the fraction of segmentation errors.

We have repeated this experiment with an inexact query. Here the query pattern is a distorted version of the target in the database. An example is provided by the digital map described earlier, which is a barrel-distorted version of the target. Figures 4.11 and 4.12 show the retrieval accuracy and the average recall distance as a function of the fraction of segmentation errors. A more complex sensitivity pattern emerges in this case. In the case of the retrieval accuracy (Figure 4.11) there is a marked difference in the different performance curves. The effect of line deletion is

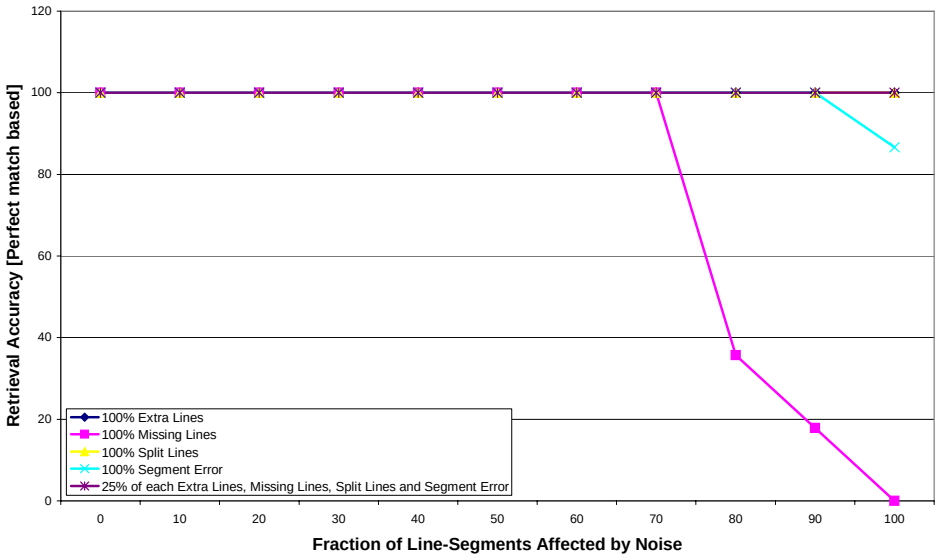



Figure 4.9: Effect of various kinds of noise to the retrieval performance using the feature-based relational similarity measure. The query is a noisy version of a unique target in the database.

again to rapidly degrade recall performance. 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 marked than in the case of line-deletions. This pattern is again reflected in the average similarity measure.

It is important at this stage to look back at the result obtained by the structurally gated histogram method described and evaluated in the previous chapter (see Chapter 3 Section 3.4). Comparing the retrieval accuracy performance in the case of exact queries (Figures 3.9 and 4.9) shows a dramatic improvement. Indeed the histogram based performance dropped off when the fraction of line-segment affected by errors exceeds 50%. However, the relational similarity measure still provides 100% recognition accuracy when 70% of the lines are subjected to errors. When the level

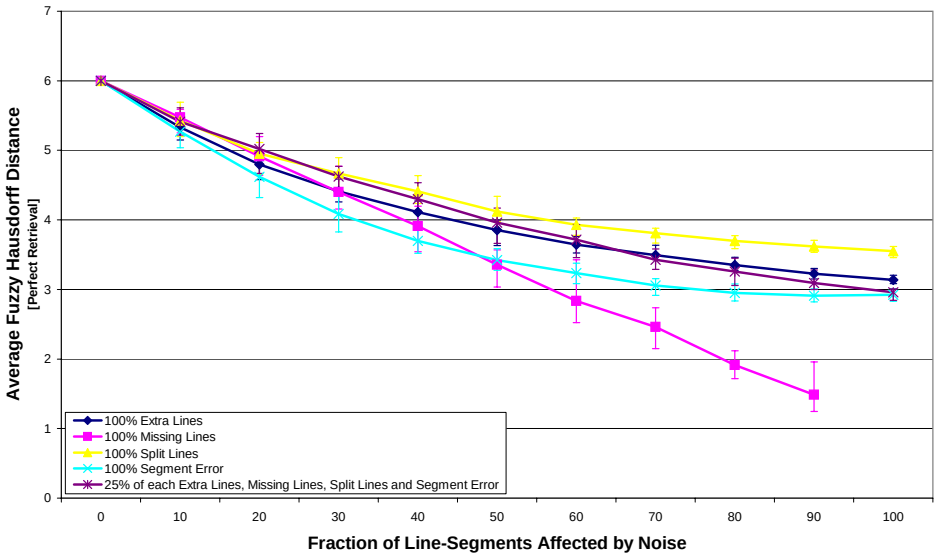



Figure 4.10: Effect of various kinds of noise on the fuzzy relational distance. The query is a noisy version of a unique target in the database.

of corruption exceeds 70%, then it is only the line deletion that significantly affects the performance. This is very understandable since most the evidence required to recognise the objects are likely to have been removed.

In the case of inexact queries (see Figure 3.11 and 4.11) a similar pattern emerges. For the relational histogram the performance starts to be affected when the line-segment error exceeds 15%. Accuracy becomes poor, approximately 50%, when the fraction of line-segments subjected to errors exceeds 40%. Turning our attention to the feature-set based relational similarity measure, performances of over 80% retrieval accuracy are obtained with up to 60% of the lines-segment corrupted. In fact, recognition accuracy never falls below 75%, apart from the case of line deletion. Again removing lines from the query pattern leads to the poorest recall rate. This is because of the lack of surviving evidence between the query and the model when only a small portion of the original line pattern remains.

To conclude the sensitivity study, we focus more closely on the role of segment end-point errors. The reason for this is that such errors will effect the accuracy of the

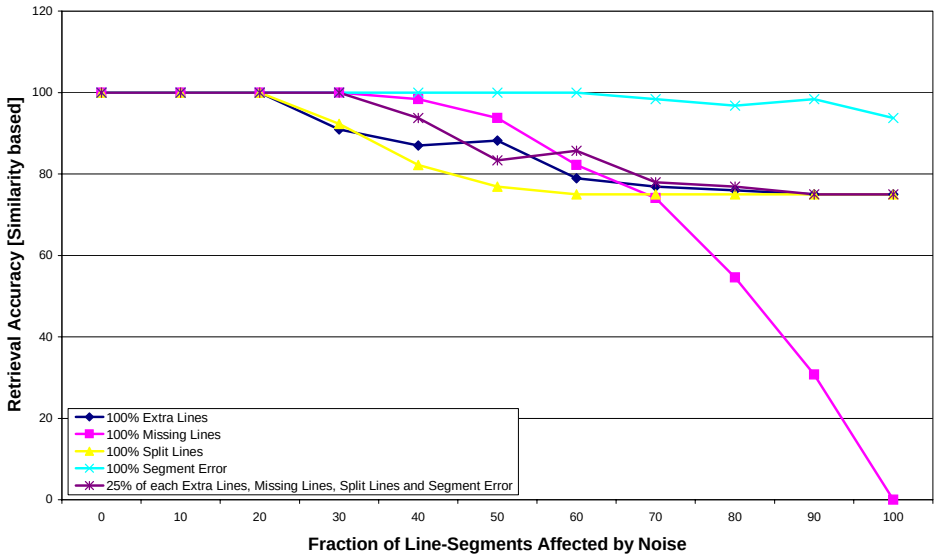



Figure 4.11: Effect of various kinds of noise to the retrieval performance using the feature-based relational similarity measure. The targets and the query are similar but not necessarily identical.

relational measurements. Figure 3.12 shows the average error in the relative angle attribute as a joint function of the fraction of lines affected by such errors and the standard-deviation of the Gaussian position error. The main feature to note from this plot is that the angle error increases with both the fraction of affected lines and the variance of the positional errors.

Figures 4.13 and 4.14 show the effect of line end-point position errors for an exact query. The different curves in the two plots correspond to different values of the standard deviation of the end-point position errors. In Figure 4.13 we show the retrieval accuracy as a function of the fraction of lines affected by end-point errors. As the standard deviation of the position error increases, the fraction of corrupt lines for which perfect recall is possible decreases. In Figure 4.14 we show the average similarity measure as a function of the fraction of corrupt lines. A similar pattern emerges. The average similarity measure decreases with increasing standard deviation of the end-point errors.

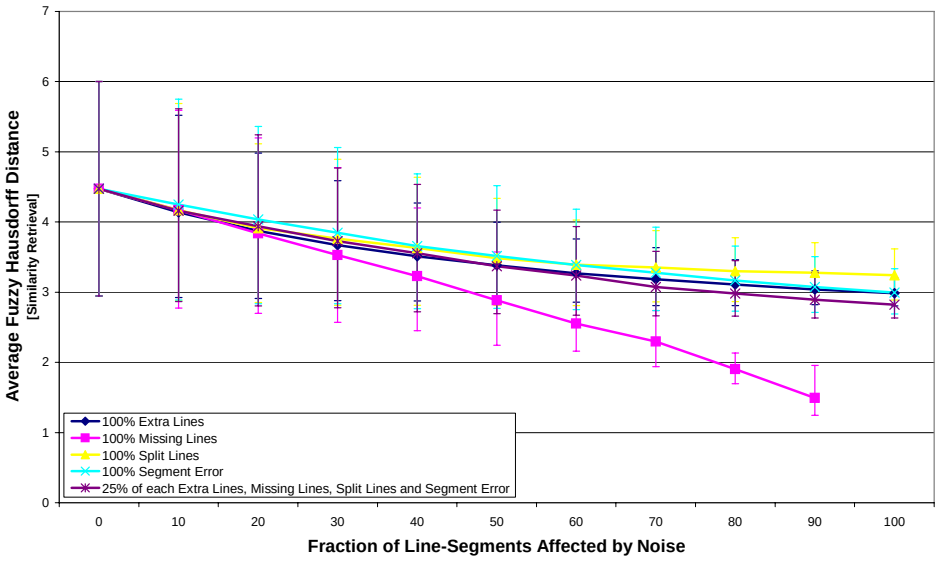



Figure 4.12: Effect of various kinds of noise on the fuzzy relational distance. The targets and the query are similar but not necessarily identical.

Figures 4.15 and 4.16 repeat these experiments for an inexact query. The same pattern of performance emerges. However, perfect recall is only achieved for lower levels of line corruption.

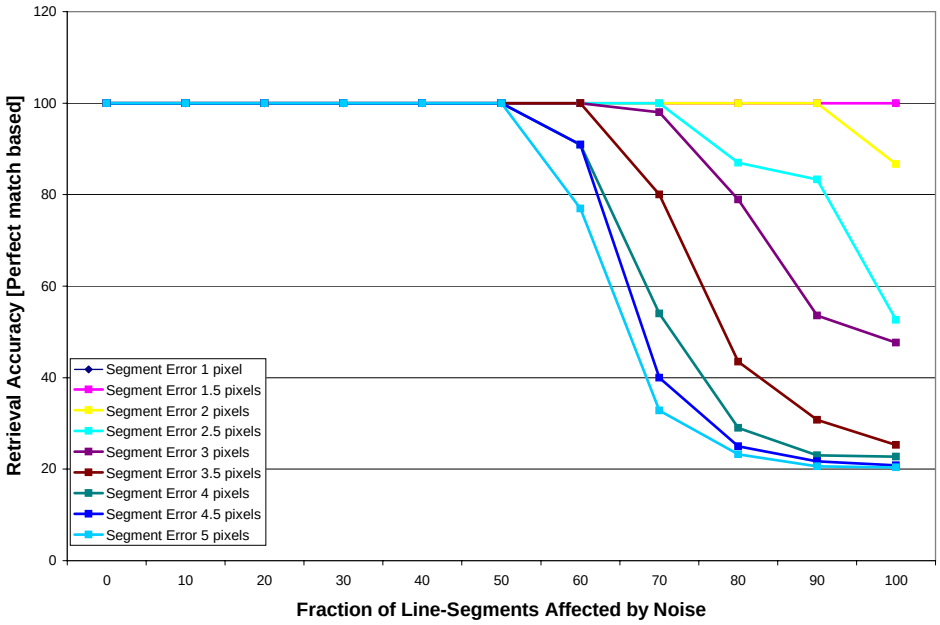



Figure 4.13: Effect of introducing segment errors to the retrieval performance using the feature-based relational similarity measure. The query is a noisy version of a unique target in the database.

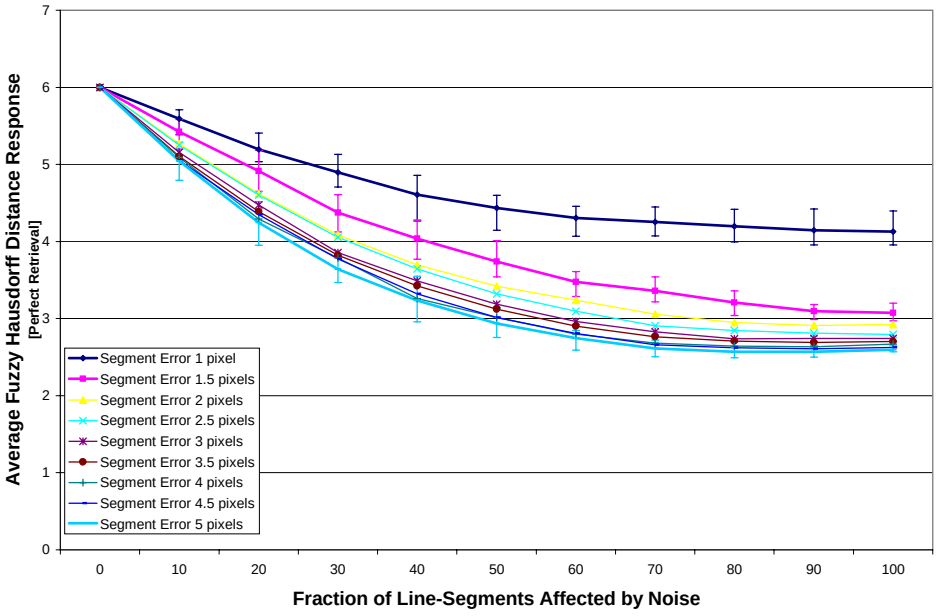



Figure 4.14: Effect of various kinds of noise on the fuzzy relational distance. The query is a noisy version of a unique target in the database.

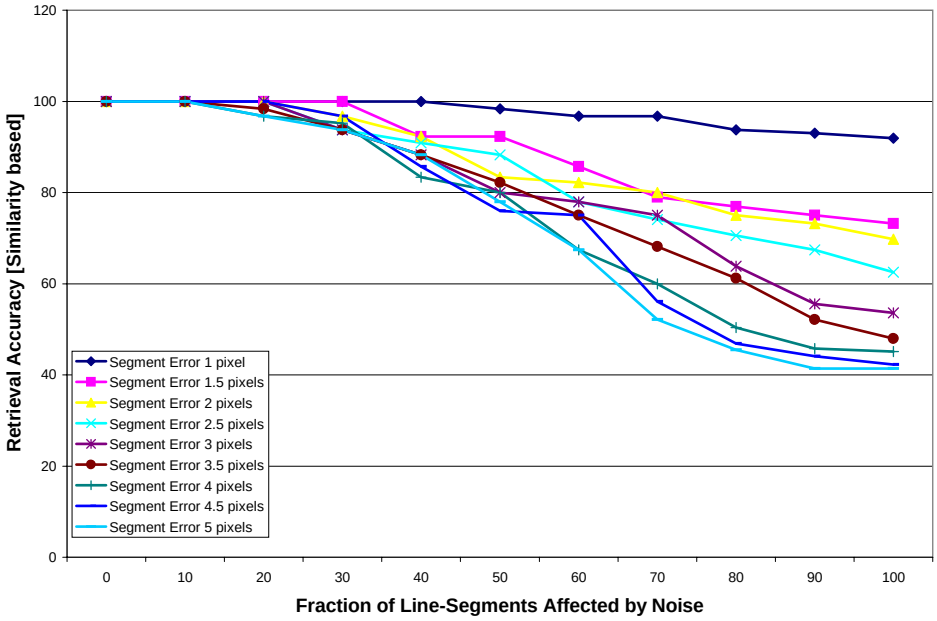



Figure 4.15: Effect of introducing segment errors to the retrieval performance using the feature-based relational similarity measure. The targets and the query are similar but not necessarily identical.

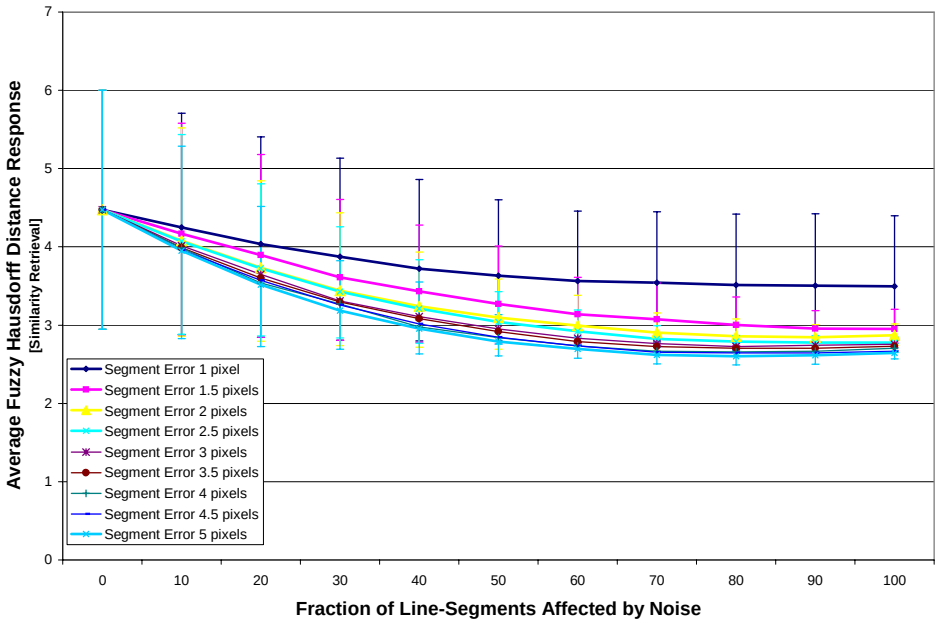



Figure 4.16: Effect of various kinds of noise on the fuzzy relational distance. The targets and the query are similar but not necessarily identical.

## 4.7 Summary

In this chapter 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 database 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 standard Hausdorff distance in terms of its recognition performance.

An important contribution in this chapter has been to demonstrate some of the noise sensitivity systematics that limit the retrieval accuracy. Our study reveals that this method offers noticeable recognition performance improvements over the method presented in the previous chapter. This allows our hierarchical methods to refine the selection of retrieved objects even further.

# Chapter 5

# Attributed Relational Graph Representation and Retrieval

The work in the previous two chapters was aimed at reducing the set of possible object models corresponding to a query object from a large image database. In the first instance (see Chapter 3, a structurally gated pairwise geometric histogram representation is employed to rapidly prune the large library of models. The method presented in Chapter 4 refines the set of object model hypotheses provided by the simple histogram based technique. It uses a more complex yet efficient relational similarity measure. The relational similarity measure operates by comparing local image feature arrangements between a query line-pattern and the set of patterns in the library. The consistency of the relational arrangement is not taken into account. The method presented in this chapter, aims to supplement the local feature structure and the attribute information with the more global (or inter-feature) constraints provided by an adjacency graph. Additionally, the technique can be used to find detailed correspondences between the query object and its conforming model description.

To meet this goal, this chapter describes a Bayesian graph matching algorithm for data-mining from large structural databases. The matching algorithm uses

edge-consistency and node attribute similarity to determine the maximum *a posteriori* probability of a query graph for each of the candidate matches in the reduced database. The node feature-vectors are constructed by computing normalised histograms of pairwise geometric attributes. Computing the Bhattacharyya distance between the histograms assesses attribute similarity. Recognition is realised by selecting the candidate from the database which has the largest *a posteriori* probability. We illustrate the recognition technique on a large database of line patterns extracted from real-world imagery. Here, the recognition technique is shown to significantly outperform a number of algorithm alternatives. Finally, a sensitivity study exposes the robustness of the method to a variety of line-segment errors.

## 5.1 Related Work and Motivation

Since Barrow and Popplestone (Barrow and Popplestone, 1971) first suggested that relational structures could be used to represent and interpret 2D scenes, there has been considerable interest in the machine vision literature in developing practical graph-matching algorithms (Sanfeliu and Fu, 1983; Shapiro and Haralick, 1985; Gold and Rangarajan, 1996; Wilson and Hancock, 1997). The main computational issues are how to compare relational descriptions when there is significant structural corruption (Sanfeliu and Fu, 1983; Shapiro and Haralick, 1985; Wilson and Hancock, 1997) and how to search for the best match (Gold and Rangarajan, 1996). Despite resulting in significant improvements in the available methodology for graph matching, there has been little progress in applying the resulting algorithms to large-scale object recognition problems. Most of the algorithms developed in the literature are evaluated for the relatively simple problem of matching a model graph against a scene known to contain the relevant structure.

A more realistic problem is that of taking a large number (maybe thousands) of scenes and retrieving the ones that best match the model. Although this problem

is key to data-mining from large libraries of visual information, it has invariably been approached using low-level feature comparison techniques. Concrete examples include the use of colour (Swain and Ballard, 1991), receptive field (Schiele and Crowley, 1996) or edge orientation (Jain and Vailaya, 1996) histograms. Very little efforts (Sengupta and Boyer, 1995) have been devoted to matching higher-level structural primitives such as lines, curves or regions. Moreover, because of the perceived fragility of the graph matching process, there has been even less effort directed at attempting to retrieve shapes using relational information.

We aim to fill this gap in the literature by using graph matching as a mean of retrieving the shape from a large database that most closely resembles a query shape. Although the indexation images in large databases is a problem of current topicality in the computer vision literature (Niblack et al., 1993; Pentland et al., 1994; Gevers and Smeulders, 1992; Swain, 1993; Picard, 1995), the work presented in this chapter is more ambitious.

Firstly, we adopt a structural abstraction of the shape recognition problem and match using attributed relational graphs. Each shape in our database is a pattern of line-segments. The structural abstraction is a nearest neighbour graph for the centre-points of the line-segments. In addition, we exploit attribute information for the line patterns. The geometric arrangement of the line-segments is encapsulated using a histogram of Euclidean invariant pairwise (binary) attributes. For each line-segment in turn we construct a normalised histogram of relative angle and length with the remaining line-segments in the pattern. These histograms capture the local geometric context of each line-segment. Moreover, we interpret the pairwise geometric histograms as measurement densities for the line-segments, which we compare using the Bhattacharyya distance. The choice of the Bhattacharyya distance measure for histogram comparison was guided by the results obtained while investigating a number of histogram distance measures (see Chapter 3 Section 3.2.2).

Once we have established the pattern representation, we realise object recogni-

tion using a Bayesian graph-matching algorithm. This is a two-step process. Firstly, we establish correspondence matches between the individual tokens in the query pattern and each of the patterns in the database. Since this is a computationally intensive process, we first filter the set of potential patterns using a rapid histogram-based recognition technique as described in chapter 3. The resulting set of hypothetical model candidates is further reduced using to the simple feature based relational similarity measure presented in Chapter 4. The correspondence matches are sought so as to maximise the *a posteriori* measurement probability. Here we use an extension of the graph-matching technique recently reported by Wilson and Hancock (Wilson and Hancock, 1997) in which we use the correspondence matches residing on the edges rather than the nodes of our attributed relational graphs to assess consistency. Once the MAP correspondence matches have been established, then the second step in our recognition architecture involves selecting the line-pattern from the database, which has maximum matching probability.

The outline of this chapter is as follows. Having described in the previous chapter (see Chapter 2 Section 2.1) the pairwise geometric attributes and the structural information that may be used to represent the shape content of the line-patterns, we may now proceed with the description of the rapid attributed graph matching algorithm for recognition and correspondence matching. In Section 5.2 we describe the maximum *a posteriori* probability (MAP) framework, which is a basis of our attributed graph matching method. This focuses on details of the structural comparison between a pair of graphs. Section 5.3 presents two distinct methodologies for representing and comparing the attribute associated with the nodes of the ARG. The representation can either be based on measurement vectors as described in Section 5.3.1 or as a histogram of attributes. This graph attribute description and the method used to measure consistency between nodes in terms of measurements are presented in Section 5.3.2. Experimental evaluation of the technique is presented in Section 5.4. This takes the form of a comparative study aimed at establishing the

effectiveness of the proposed algorithm and some of its variants. The goals of this experimentation are twofold. The first sensitivity analysis investigates the importance of the choice of attribute representations on the recognition effectiveness and accuracy. The second experiment is concerned with identifying the most effective graph structure. Section 5.5 addresses the sensitivity of the recognition methods in errors of the line-segmentation process. Then, we further our investigations about sensitivity to line segmentation systematics and present some examples using our proposed attributed relational graph matching algorithm in Section 5.5.1. Finally, Section 5.7 presents some conclusions and suggests directions for future investigation as far as this approach is concerned.

## 5.2 MAP Framework

Formally our recognition problem is posed as follows. Each Attributed Relational Graph (ARG) in the database is a triple, $G = (V_G, E_G, A_G)$, where $V_G$ is the set of vertices (nodes), $E_G$ is the edge set ($E_G \subset V_G \times V_G$), and $A_G$ is the set of node attributes. In our experimental example, the nodes represent line-structures segmented from 2D images. The edges are established by computing the N-nearest neighbour graph for the line-centres. Each node $j \in V$ is characterised by a vector of attributes, $\underline{\mathrm{x}}_j$ and hence $A_G = \{\mathrm{x}_j | j \in V\}$. In the work reported here the attribute-vector represents the contents of a normalised pairwise attribute histogram.

The database of line-patterns is represented by the set of ARG's $\mathcal{D} = \{G\}$. The goal is to retrieve from the database $\mathcal{D}$, the individual ARG that most closely resembles a query pattern $Q = (V_Q, E_Q, A_Q)$ and identify their correspondences. We pose the retrieval process as one of associating with the query the graph from the database that has the largest *a posteriori* probability. In other words, the class identity of the graph, which most closely corresponds to the query, is

$$\omega_Q = \arg\max_{G' \in \mathcal{D}} P(G'|Q) \tag{5.1}$$

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 database and the query $Q$. The set of correspondences between the query $Q$ and the ARG $G$ is a relation $f_G : V_G \mapsto V_Q$ 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.

$$f_G = \{(a, \alpha); a \in V_G, \alpha \in V_Q\} \subseteq V_G \times V_Q$$

Although this may appear to be a brute force method, it must be stressed that we view this process of correspondence matching as the final step in the filtering of the line-patterns. We provide more details of practical implementation in the experimental section (see Section 5.4).

With the correspondences to hand we can re-state our maximum *a posteriori* probability recognition objective as a two step process. For each graph $G$ in turn, we locate the maximum *a posteriori* probability mapping function $f_G$ onto the query $Q$. The second step is to perform recognition by selecting the graph whose mapping function results in the largest matching probability. These two steps are succinctly captured by the following statement of the recognition condition

$$\omega_Q = \arg \max_{G' \in \mathcal{D}} \max_{f_{G'}} P(f_{G'}|G', Q) \tag{5.2}$$

This global MAP condition is developed into a useful local update formula by applying the Bayes formula to the *a posteriori* matching probability. The simplification is as follows

$$P(f|G, Q) = \frac{p(A_G, A_Q|f_G)P(f_G|V_G, E_G, V_Q, E_Q)P(V_G, E_G)P(V_Q, E_Q)}{P(G)P(Q)} \tag{5.3}$$

The terms on the right-hand side of the Bayes formula convey the following meaning. The conditional measurement density $p(A_G, A_Q|f_G)$ models the measurement similarity of the node-sets of the two graphs. The conditional probability

$P(f_G|V_G, E_G, V_Q, E_Q)$ models the structural similarity of the two graphs under the current set of correspondence matches. The assumptions used in developing our simplification of the *a posteriori* matching probability are as follows.

We assume that the joint measurements are conditionally independent of the structure of the two graphs provided that the set of correspondences is known, i.e. $P(A_G, A_Q|f_G, E_G, V_G, E_Q, V_Q) = P(A_G, A_Q|f_G)$.

We assume that there is conditional independence of the two graphs in the absence of correspondences. In other words, $P(V_G, E_G, V_Q, E_Q) = P(V_Q, E_Q)P(V_G, E_G)$ and $P(G, Q) = P(G)P(Q)$.

The graph priors $P(V_G, E_G)$, $P(V_Q, E_Q)$ $P(G)$ and $P(Q)$ are taken as uniform and are eliminated from the decision making process.

To continue our development, we first focus on the conditional measurement density $p(A_G, A_Q|f)$, which models the process of comparing attribute similarity on the nodes of the two graphs. Assuming statistical independence of node attributes, the conditional measurement density $p(A_G, A_Q|f)$ can be factorised over the Cartesian pairs $(a, \alpha) \in V_G \times V_Q$, which constitute the correspondence match $f_G$ in the following manner

$$p(A_G, A_Q|f_G) = \prod_{(a,\alpha)\in f_G} p(\mathbf{x}_a, \mathbf{x}_\alpha|f_G(a) = \alpha) \tag{5.4}$$

As a result the correspondence matches may be optimised using a simple node-by-node discrete relaxation procedure. The rule for updating the match assigned to the node $a$ of the graph $G$ is

$$f_G(a) = \arg\max_{\alpha\in V_Q\cup\{\Phi\}} p(\mathbf{x}_a, \mathbf{x}_\alpha)|f_G(a) = \alpha)P(f_G|E_G, E_Q) \tag{5.5}$$

In order to model the structural consistency of the set of assigned matches, we turn to the framework recently reported by Finch, Wilson and Hancock (Finch et al., 1997b). This work provides a framework for computing graph-matching energies using the weighted Hamming distance between matched cliques.

Since we are dealing with a large-scale object recognition system, we would like to minimise the computational overheads associated with establishing correspondence matches. For this reason, rather than working with graph neighbourhoods or cliques, we chose to work with the relational units of the smallest practical size. In other words we satisfy ourselves with measuring consistency at the edge level. For edge-units, the structural matching probability $P(f_G|V_G, E_G, V_Q, E_Q)$ is computed from the formula

$$\ln P(f_G|V_G, E_G, V_Q, E_Q) = \sum_{(a,b)\in E_G} \sum_{(\alpha,\beta)\in E_Q} \left\{ \ln(1-P_e) s^G_{a,\alpha} s^Q_{b,\beta} + \ln P_e (1 - s^G_{a,\alpha} s^Q_{b,\beta}) \right\} \quad (5.6)$$

where $P_e$ is the probability of an error appearing on one of the edges of the matched structure. The $s_{a,\alpha}$ are assignment variables, which are used to represent the current state of match and convey the following meaning

$$s^G_{a,\alpha} = \begin{cases} 1 & \text{if } f_G(a) = \alpha \\ 0 & \text{otherwise} \end{cases} \quad (5.7)$$

## 5.3 Attributes Representation and Comparison

The methodology used to compute the pairwise attributes used in our representation is described in Chapter 2 Section 2.2. Both the directed relative angle and the directed relative position attributes are invariant to changes of scale, rotation and translation. We will investigate a number of alternative attribute representations. In the first instance, we treat the local pairwise measurements as a vector associated with each graph node. Secondly, we introduce the idea of combining local pairwise line-segments attributes in a sparse histogram. This may be viewed as having a histogram per node in the graph. In either case, we have the choice of either using a single attribute or combining both the relative angle and position attributes.

### 5.3.1 Vector-Based Consistency

We use only the node attributes rather than an attribute histogram to model the *a posteriori* matching probabilities. Both the relative angle $\theta_{i,j}$ and relative position $\vartheta_{i,j}$ attributes are available to describe a pair of line-segments. An attribute vector $\mathbf{x}_i(\theta_{i,j}, \vartheta_{i,j})$ defines the pairwise relations for each node $i$ and its set of connected edges $C_i = \{j|(i,j) \in E\}$. We use the work presented in the previous chapter on feature based representation and retrieval to access the probability of a node $i$ from the graph $G$ corresponding to node $\iota$ in the query graph $Q$. The attribute similarity between a pair of nodes can be computed by comparing their respective pairwise attribute vectors. However, the ordering of the attributes in the vector is not fixed. Comparing the vectors cannot be performed by simple distance between respective vector components. Ordering the attribute vectors using cyclic permutation would require more processing time without improving the comparison complexity. For example the ordering could be disrupted if some image features were removed. This cyclic problem is similar to the one we faced in chapter 4 when we compared local features. We will therefore use a similar approach to compute the *a posteriori* matching probabilities. The probability of match between the pattern vectors is computed using a weighted fuzzy Hausdorff distance between the sets of attribute vectors. As a result we note

$$P(f_G(i) = \iota|\mathbf{x}_i\mathbf{x}_\iota) = 1 - \frac{1}{n_{C_i^Q}} \sum_{C_i^Q} \min_{C_i^M} \left(1 - \Gamma_\sigma(||\mathbf{x}_\iota^G - \mathbf{x}_i^Q||)\right) \tag{5.8}$$

where $n_{C_i^Q}$ is the number of edges connected to node $i$ in the query graph $Q$ and $\Gamma_\sigma(||\mathbf{x}_\iota^Q - \mathbf{x}_i^G||)$ is a distance weighting function. In Chapter 4 Section 4.3.2, we considered a number of alternative robust weighting functions. The experiment presented in Section 4.5 shows that the weighting functions that provide the best recognition performance are the Gaussian and the Huber kernel. Again, it is the Gaussian kernel

of the form

$$\Gamma_\sigma(\rho) = \exp\left(-\frac{1}{2}\frac{\rho^2}{\sigma^2}\right)$$

that is going to be used in this attribute consistency measure. The *a posteriori* matching probability based on both the relative angle $\theta_(i,j)$ and the relative position $\vartheta_(i,j)$ can be written as follows.

$$P(f(i) = \iota|\mathbf{x}_i\mathbf{x}_\iota) = 1 - \frac{1}{n_{C_i^Q}}\sum_{C_i^Q}\min_{C_\iota^M}\left(1 - (\exp[-\frac{1}{2}\{\frac{(\theta_i - \theta_\iota)^2}{\sigma_\theta^2} + \frac{(\vartheta_i - \vartheta_\iota)^2}{\sigma_\vartheta^2}\}])\right) \quad (5.9)$$

where $\sigma_\theta$ and $\sigma_\vartheta$ are the measurement variances for the two pairwise geometric attributes and $C_I^Q$ is the set of edge connected to node $i$ in graph $G$.

In the experimental section, we will investigate several variants of the histogram per node attribute matching process. Again, we will experiment with the two geometric attributes taken alone and in combination. The representations will respectively make use of two-dimensional histograms or one-dimensional histograms. Then, we compare the effect of using the attribute histogram at every step of the iterative graph matching process with using it purely for the purposes of initialisation.

### 5.3.2 Histogram-Based Consistency

The angle and position attributes $\theta_{i,j}$ and $\vartheta_{i,j}$ are binned in a histogram. Suppose that $S_i(B_\theta, B_\vartheta) = \{(i,j) \in E \wedge \theta_{i,j} \in R(B_\theta) \wedge \vartheta_{i,j} \in R(B_\vartheta)\}$ is the set of nodes whose pairwise geometric attributes with the node $i$ are spanned by the range of directed relative angles $R(B_\theta)$ and the relative position attribute range $R(B_\vartheta)$. The contents of the histogram bin spanning the two attribute ranges is given by $H_i(B_\theta, B_\vartheta) = |S_i(B_\theta, B_\vartheta)|$. Each histogram contains $n_\theta$ relative angle bins and $n_\vartheta$ relative position bins. The normalised geometric histogram bin-entries are computed as follows

$$h_i(B_\theta, B_\vartheta) = \frac{H_i(B_\theta, B_\vartheta)}{\sum_{B'_\theta=1}^{n_\theta}\sum_{B'_\vartheta=1}^{n_\vartheta} H_i(B_\theta, B_\vartheta)} \quad (5.10)$$

The probability of match between the pattern-vectors $i$ and $\iota$ is computed using the Bhattacharyya distance $D_{h_i,h_\iota}$ between the normalised histograms. The Bhattacharyya distance measure was selected from a number of alternative histogram distance measures. The experiment presented in Chapter 3 Section 3.3.4 clearly indicated that statistical histogram distance measures outperform Euclidean norms.

$$P(f(i) = \iota|\mathbf{x}_i, \mathbf{x}_\iota) = \sum_{B_\theta=1}^{n_\theta} \sum_{B_\vartheta=1}^{n_\vartheta} h_i(B_\theta, B_\vartheta) h_\iota(B_\theta, B_\vartheta) = \exp[-D_{h_i,h_\iota}] \tag{5.11}$$

## 5.4 Experiments

The aim in this section is to evaluate the graph-based recognition scheme on a database of real-world line-patterns. We have conducted our recognition experiments with the database described in chapter 2.

In order to prune the set of line-patterns for detailed graph matching we select about 10% of the database using a two-step process. This consists of first refining the database using a global histogram of pairwise attributes (see Chapter 3). The top quartile of matches selected in this way are then further refined using a variant of the Hausdorff distance to select the set of pairwise attributes that best match against the query as described in Chapter 4.

The experimental study has two aims. Firstly, we focus on the issue of the best choice of recognition strategy. The aim here is to compare the performance delivered by the various components of our recognition process. Secondly, we address the choice of the adjacency structure. This takes the form of a comparison between a number of alternative neighbourhood graphs and the Delaunay graph.

We aim to assess the importance of different attribute representations on the retrieval process. To this end, we compare vector-based and the histogram-based attribute representations. We also consider the effect of taking the relative angle and relative position attributes both separately and in combination. The final aspect of

the comparison is to consider the effects of using the attribute consistency either purely for initialisation purposes or in a persistent way during the iteration of the matching process. To this end we consider the following variants of our algorithm.

- **Persistant Attributes:** Here we use both the attribute information and the structural constraints in order to find the best match between a pair of graph. The recognition is performed using the method described in section 5.2.

- **Non-Persistent Attributes:** Here we ignore the attribute information provided by either the node-histograms or the node-vectors after the first iteration. In other words we attempt to maximise only the structural congruence of the graphs. The recognition condition is

$$\omega_Q = \arg\max_{G' \in \mathcal{D}} \sum_{(a,b) \in E'_G} \sum_{(\alpha,\beta) \in E_Q} \left\{ + \ln(1 - P_e) s_{a,\alpha} s_{b,\beta} + \ln P_e (1 - s_{a,\alpha} s_{b,\beta}) \right\} \quad (5.12)$$

  In this mode we simply use the attribute histograms for the purposes of initialisation. The assignment variables are initialised as follows

$$s_{a,\alpha} = \begin{cases} 1 & \text{if } f_G(a) = \arg\min_\alpha D_{a,\alpha} \\ 0 & \text{otherwise} \end{cases} \quad (5.13)$$

In Table 5.1 we present the performance for each of the recognition strategies in turn. The table lists the recall performance together with the average number of iterations per recall for each of the recognition strategies in turn. The main features to note from this table are as follows. Firstly, the iterative recall using the full histogram representation outperforms each of the remaining recognition methods in terms of both accuracy and computational overheads. Secondly, it is interesting to compare the effect of using the histogram in the initialisation only and iteration persistent modes. In the latter case the recall performance is some 32% better than in the former case. In the non-persistent mode the best recognition accuracy that can

| Graph Matching Strategy | Retrieval Accuracy | Iterations per recall |
|---|---|---|
| Rel. Position Attribute (Initialisation only) | 39% | 5.2 |
| Rel. Angle Attribute (Initialisation only) | 45% | 4.75 |
| Rel. Angle + Position Attributes (Initialisation only) | 58% | 4.27 |
| 1D Rel. Position Histogram (Initialisation only) | 42% | 4.7 |
| 1D Rel. Angle Histogram (Initialisation only) | 59% | 4.2 |
| 2D Histogram (Initialisation only) | 68% | 3.9 |
| Rel. Position Attribute (Persistent) | 63% | 3.96 |
| Rel. Angle Attribute (Persistent) | 89% | 3.59 |
| Rel. Angle + Position Attributes (Persistent) | 98% | 3.31 |
| 1D Rel. Position Histogram (Persistent) | 66% | 3.46 |
| 1D Rel. Angle Histogram (Persistent) | 92% | 3.23 |
| 2D Histogram (Persistent) | 100% | 3.12 |

Table 5.1: Recognition performance of various recognition strategies averaged over 26 queries in a database of 260 line-patterns.

be obtained is 68%. Moreover, the recall is typically achieved in only 3.12 iterations as opposed to 3.9 (average over 26 queries on a pruned database of 260 images). Finally, the histogram representation provides better performance, and more significantly, much faster recognition than the attribute vector similarity measure. When the attributes are used on their own, rather than as a pair, then it is the relative angle that appears to be the most powerful. This result conforms with our findings in Chapter 3 Section 3.3.4).

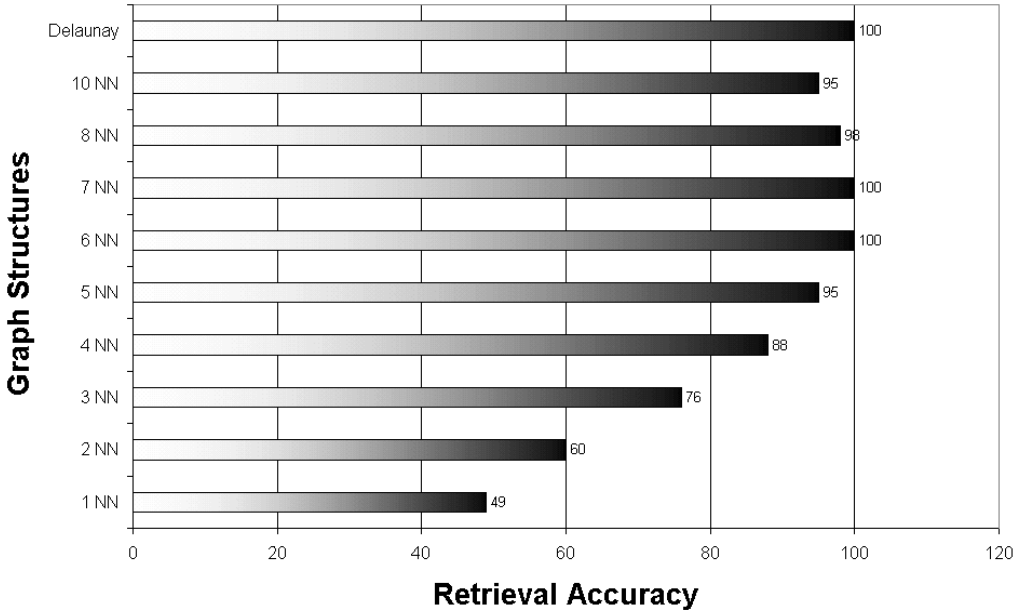



Figure 5.1: Effect of the choice of structural representation on retrieval accuracy.

The second set of experiments examines the role of the structural representation. Here we investigate the effect of the order of the N-nearest neighbour graph. We also compare the N-nearest neighbour graph with the Delaunay graph. In Figure 5.1 we show a plot of the recognition accuracy on a pruned database of 260 line-patterns. The main effects to note are that the optimal order of the nearest neighbour graph is 6. Again, similar findings where obtained while investigating structural representation for both the structurally gated histogram approach (Chapter 3) and the relational similarity measure (Chapter 4). Although the recent study of Tuceryan and

Chorzempa (Tuceryan and Chorzempa, 1991) indicates that the Delaunay graph offers greater robustness to noise and clutter, in the experiments reported here the performance advantage is not noticeable.

## 5.5 Sensitivity Study

The aim in this section is to investigate the sensitivity of the recognition strategy to the systematics of the line-segmentation process. Again, we investigate the five types of segment errors described in Chapter 3 Section 3.4.

In our first experiment, we query the database with a line pattern that is known to have an identical counterpart. However, a fraction of the lines have been subjected to the segmentation errors described above. Figure 5.2 compares the retrieval accuracy as a function of the fraction of lines that are subjected to segmentation errors. The recognition performance does not degrade until the fraction of errors exceeds 70%. The only destructive type of error is the line dropout. As the number of line-segments providing evidence about the similarity of the two object decreases, so does the probability of the two objects being identical. Removing line-segments from the line-pattern introduces changes in both the local histograms and the structure of the graph. However, the recognition accuracy is not affected by the addition of lines, the splitting of lines and the addition of end-point displacement errors. The combination of the various types of error results in perfect retrieval accuracy even when all the line-segments in the query line-pattern are affected. Recalling the results obtained by the fuzzy Hausdorff distance in Figure 4.9 of Chapter 4, it is clear that a slight improvement is provided by the more robust graph matching algorithm. Indeed, the recognition accuracy for line-patterns subjected to end-point displacement errors is not perfect in the case of the fuzzy Hausdorff distance measure. In contrast, 100% accuracy is achieved by the more computationally demanding attributed graph matching technique.

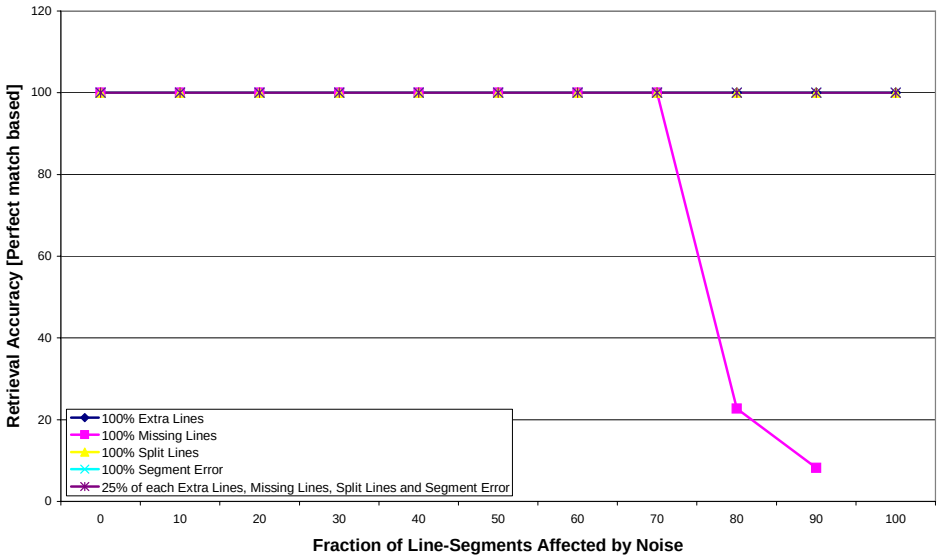



Figure 5.2: Effect of various kinds of noise to the retrieval performances using the standard pairwise geometric histogram representation. The query is a noisy version of a unique target in the database.

This experiment is now repeated for an inexact query. Here the query patterns are distorted versions of targets in the database. An example is furnished by the digital map (see Figure 2.5), which is a barrel-distorted version of the target. Important scale transformations are also noticeable between the target line-pattern and the corresponding query.

Figures 5.3 shows the retrieval accuracy as a function of the fraction of segmentation errors in the case of inexact query. A more complex sensitivity pattern emerges in this case. The removal of line-segments from the line-pattern has the greatest effect on the recognition performance. There is a rather sharp performance drop-off when more than 40% of the line-segments are missing. Extra line-segments have no effect on the representation, and therefore the recognition accuracy, until the level of added clutter exceeds 70%. Even then, the performance never falls below 90%. Robustness to the other error types is very satisfactory. Under line-segment splitting the level of recognition accuracy never drops below 80%. Similarly, for line-segment

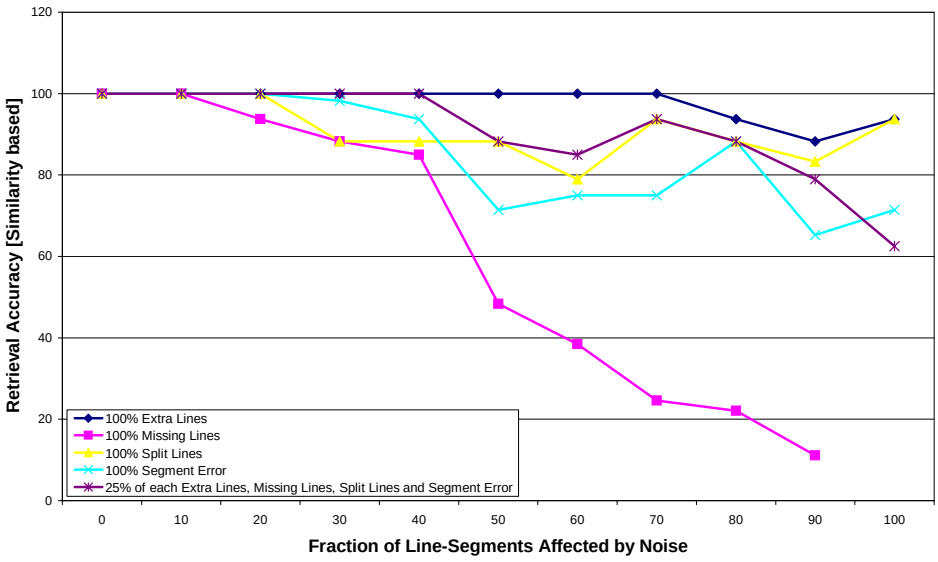



Figure 5.3: Effect of various kinds of noise to the retrieval performances using the relational pairwise geometric histogram representation. The targets and the query are similar but not necessarily identical.

end-point errors, 68% accuracy is achievable even when all the line-segment end-points have been displaced.

In terms of improvement compared with the feature-based fuzzy Haussdorff method presented in Chapter 4, the graph matching method is showing a lot more prudence in its recognition decisions. The recognition accuracy is improved in the case of line-segment addition, line-splitting and combined errors. When the number of line-segments composing an object decreases, the graph matching algorithm is not able to find enough structural and attribute evidence to provide a high matching probability. The random removal of line-segments has a greater effect on the graph matching algorithm than the feature-based fuzzy Hausdorff distance. This would not be the case if the missing line-segments were all deleted from the same area of the line-pattern. This is because neither the structure nor the attribute histograms of the remaining graph nodes (line-segments) would be affected. This situation is more likely to arise when using real world imagery because of object occlusion. The

deletion of random line-segments is also likely to be present in non-synthetic object recognition and cause perturbation but only in small amount.

To continue the sensitivity study, we focus more closely on the effects of segment end-point errors. The reason for this is that such errors will effect the accuracy of the relational measurements. Figure 5.4 shows the effect of line end-point position errors in the case where a model object is used as query and is also present in the database. The different curves in the plots correspond to different values of the standard deviation of the end-point position errors. They show the retrieval accuracy as a function of the fraction of lines effected by end-point errors. Perfect retrieval is achieved when the standard deviation is less than 5 pixels and fewer than 80% line-segments are displaced. Line-segment end-point displacements of less than 3 pixels have no effect on the performance. For higher levels of perturbation the recall accuracy falls gradually to about 40%.

Figure 5.4 shows the recognition accuracy for similarity queries. The results are averaged over ten queries. As the standard deviation of the position error increases, then so the fraction of corrupt lines for which perfect recall is possible decreases. The main point to note from these plots is that the ARG based method degrades less rapidly under line end-point errors than for both the relational histogram and the feature-based relational similarity measure. Additionally, there appears to be significant variation in the accuracy of retrieval. This is caused by small line-segments. For instance, the displacement of the end-points of a short line-segment has a dramatic effect on its orientation, even for small displacements. The line-patterns used for this experiment have many line-segments, which are under 10 pixels long.

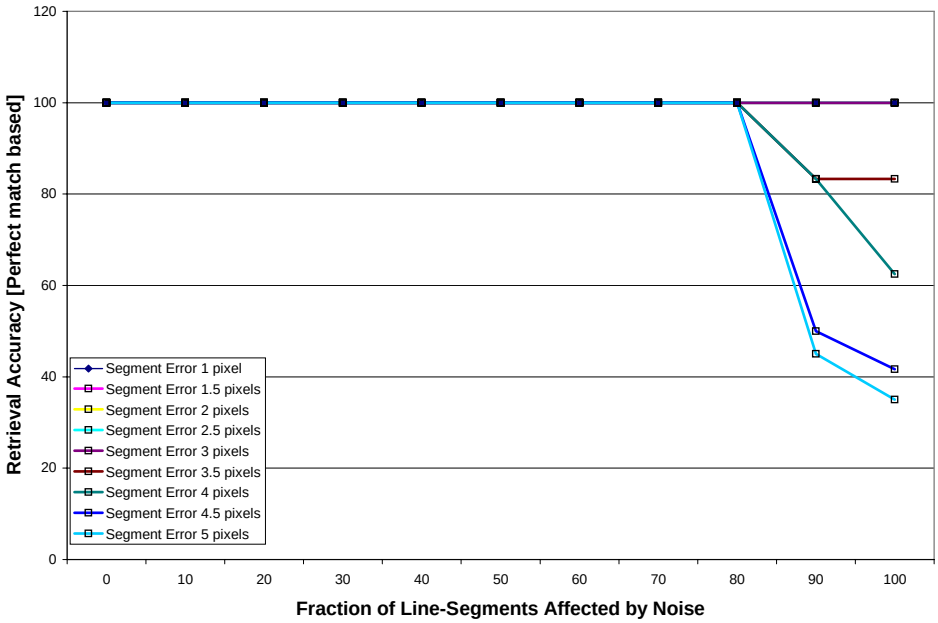



Figure 5.4: Effect of introducing segment end-point errors on the retrieval performance using the relational pairwise geometric histogram representation. The query is a noisy version of a unique target in the database.

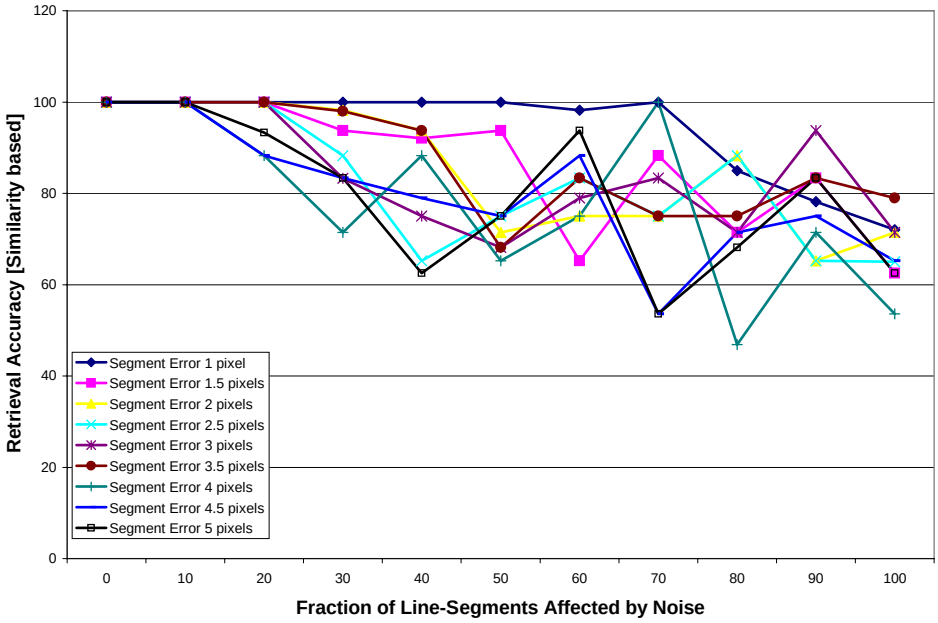



Figure 5.5: Effect of introducing segment end-point errors on the retrieval performance using the relational pairwise geometric histogram representation. The targets and the query are similar but not necessarily identical.

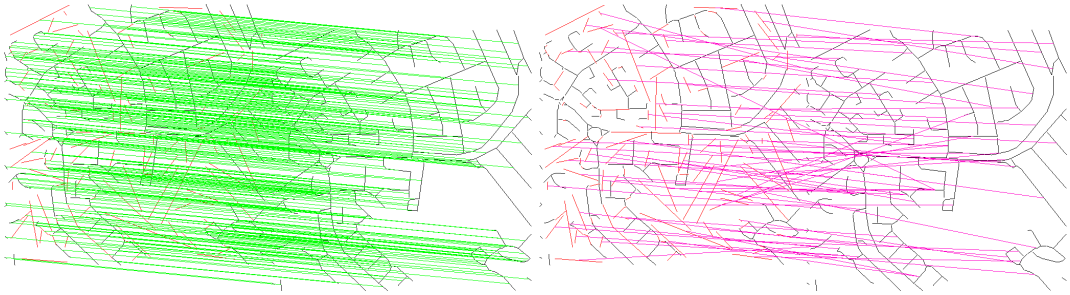
### 5.5.1 Matching Examples

In this section, we investigate the quality of the correspondences obtained by the attributed graph matching algorithm. The results presented here have been obtained by attempting to match the digital map (see Figure 2.5) against itself. In order to assess the sensitivity of the algorithm to line-segmentation errors when computing correspondences, noise is going to be added to the query line-pattern. Using the same line-pattern for both the query and target gives us ground truth. This allows us to compute performance statistics for the method.

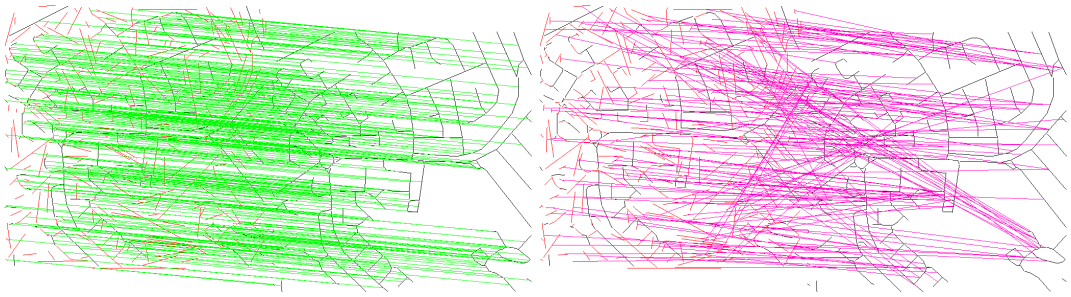
For all the matching examples provided in the remainder of this section, both the query and model line-patterns are shown side by side. The "noisy" query is shown on the left while the model is on the right. The line-segments highlighted in red in the query line-patterns have been affected by the addition of synthetic noise. Again we are going to investigate the standard five types of line-segmentation errors (i.e. extra lines, missing lines, line splitting, line-segment end-point errors and combined errors). We distinguish between correct and incorrect correspondences between the query and the model features by showing them in distinct figures. The left-hand panel of each figure depicts the correct matches. These are represented by green lines connecting the center-point of query line-segments to their accurately located model counterpart. The left-hand panel depicts the incorrect matches. These are shown by red connecting lines. These correspondences represent query nodes that have not been correctly associated with a model node.

We begin this study with the addition of extra line-segments to the query line-pattern. Figure 5.6 shows the matching results when 20% (top) and 50% (bottom) of the line-segments have been randomly added to the line-pattern. It is clear that as the amount of clutter increases, then so the number of correctly matched nodes decreases. For 20% extra line-segments the only incorrect matches come from added clutter. In other words, the original line-segments are all correctly matched. When 50% extra lines are added to the line-pattern (see Figure 5.6 (bottom)) the number

of incorrect matches increases. As reported earlier, the structure and local attribute information are not dramatically affected by the addition of extra clutter. The results summarized in Table 5.2 show a number of interesting features about the various matching examples. In the case of extra line-segments, the fraction of line-segments correctly matched is 82.5% for 20% added clutter and drops down to 65% for 50% extra line-segments. The number of iterations required is respectively three and six, while the matching probabilities are 0.61 and 0.41.



(a) 20% of the lines are affected by errors

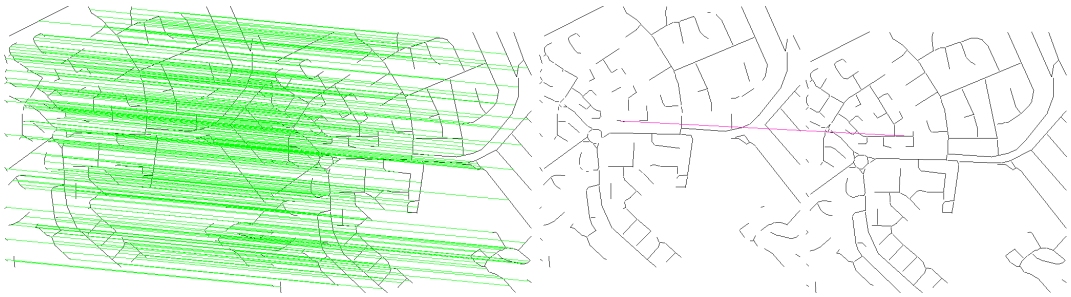


(b) 50% of the lines are affected by errors

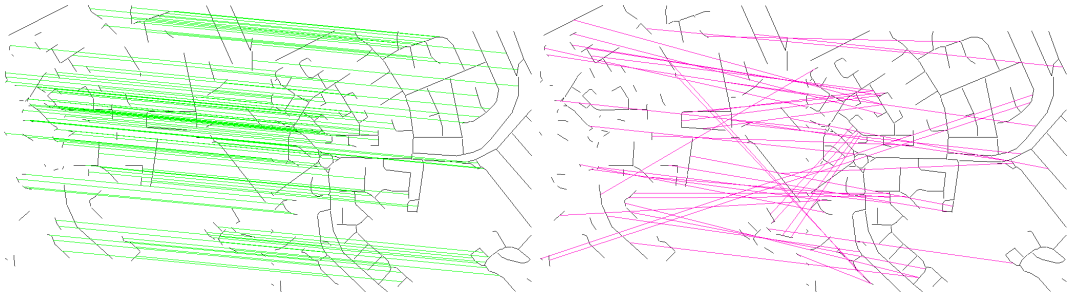
Figure 5.6: Effect of segment addition on the attributed graph matching algorithm. The lines in green show the correct correspondence while the red lines indicate a bad match.

We now turn our attention to the case where line-segments are removed from the original line-pattern. It was shown in the previous section that this type of error had some effect on the representation and the accuracy of retrieval. Figure 5.7 depicts

the correspondences found in this situation. At low levels of corruption the method provides good correspondences (99%) in three iterations. However, when only 50% of the query line-pattern, the number of mismatches is increased and the matching probability falls to 0.35. As explained earlier, this is due to both the structure and the attribute information being heavily distorted.



(a) 20% of the lines are affected by errors

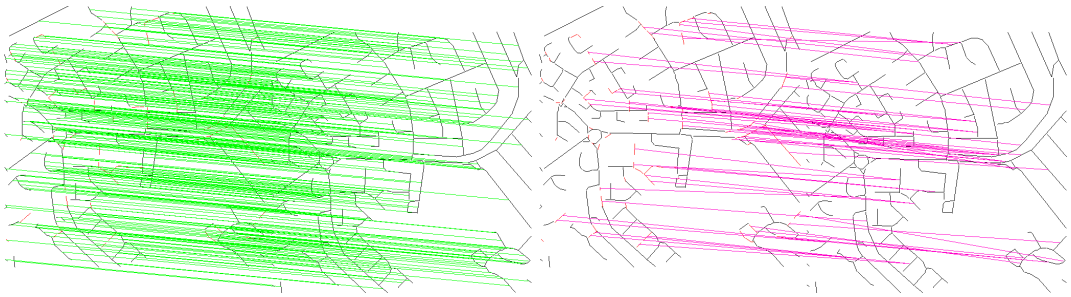


(b) 50% of the lines are affected by errors

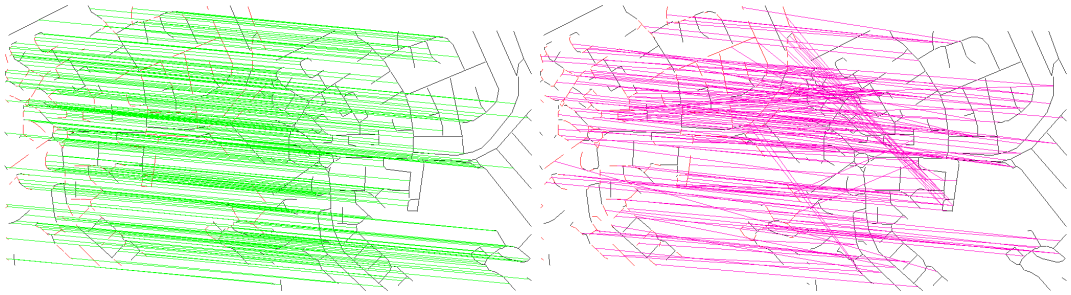
Figure 5.7: Effect of segment deletion on the attributed graph matching algorithm. The lines in green show the correct correspondence while the red lines indicate a bad match.

For the next matching example, shown in Figure 5.8, a fraction of the line-segments from the query object are split into two smaller and disconnected line-segments. This has the dual effect of adding clutter and changing the local attribute representation. Under small perturbation (see Figure 5.8 (top)) the level of accurately matched line-segments is 82.5%. Only a small portion of the noisy line-segments

causes incorrect matches. When the level of corruption is raised to 50%, it is again the "noisy" line-segments for which the correct correspondences have not been accurately computed (only 59.8% line-segment correctly matched). However, this does not have such a dramatic effect on the overall matching probability of the two line-patterns, which are respectively 0.67 for 20% corruption and 0.43 for 50% corruption.



(a) 20% of the lines are affected by errors



(b) 50% of the lines are affected by errors

Figure 5.8: Effect of segment splitting on the attributed graph matching algorithm. The lines in green show the correct correspondence while the red lines indicate a bad match.

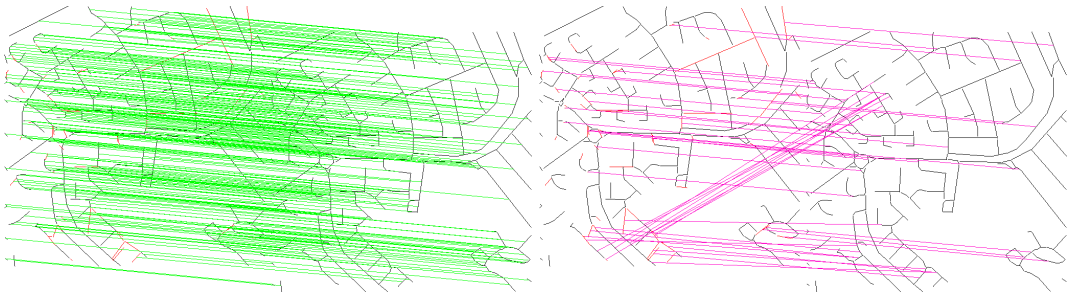
The next type of error to be investigated is the line-segment end-point displacement. Figure 5.9 shows the results of matching when 20% and 50% of the line-segments are randomly perturbed by Gaussian displacement errors of standard deviation $\sigma$ = 2 pixels. This type of error does not directly affect the structure of the graph. However, both the relative angle and relative position attributes (and there-

fore the node-histograms) are going to be affected. For 20% corruption the fraction of line-segments correctly matched is 86.3%. Incorrectly located correspondences originate mainly from noisy line-segments. When the corruption is increased to 50% the matching performance does not dramatically degrade. In fact, 65% of the line-segment correspondences are correctly identified. There is, however, an interesting observation to make at this point. The number of iterations required by the graph matching is increased to four in the case of the line-segment end-point errors, while only three iterations were required under the remaining types of error reported above. The higher level of node-histogram corruption is responsible for the poor level of correctly initialised feature-correspondences. The graph matching relies more heavily on the structural part of the representation and therefore requires more iteration to converge to the optimal set of correspondences.

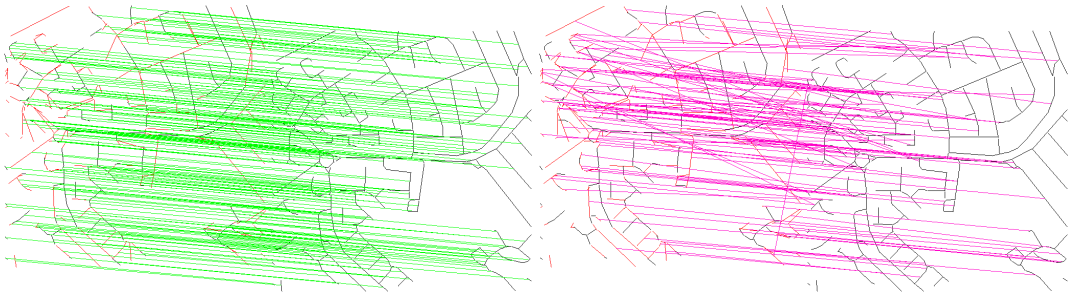
Finally, we combine all the types of error previously investigated in equal proportion. Figure 5.10 shows the results obtained for 20% corruption (top) and 50% corruption (bottom). Again, it is the corrupted line-segments for which the correspondences cannot be located. In many cases (extra lines, missing lines and split-lines) no corresponding line-segment actually exists in the model line-pattern. Combined errors simulate more closely what is likely to happen when processing real world object images than the individual error types. The histogram per node graph matching algorithm provides a good level accuracy combined with improved computational requirements when compared with the feature vector-based method. For a low level of corruption (20%), 86.3% of the line-segments are correctly matched in just four algorithm iterations. At the higher level of corruption (50%) only five iterations are required to recover 75.2% of the correspondences correctly.

| Error Type | Fraction of Segments Affected | Iterations Required | Matching Probability | Fraction of Segments Correctly Matched |
|---|---|---|---|---|
| Extra Lines | 20% | 3 | 0.61 | 82.5% |
| Extra Lines | 50% | 6 | 0.41 | 65% |
| Missing Lines | 20% | 3 | 0.7 | 99% |
| Missing Lines | 50% | 5 | 0.35 | 69% |
| Split Lines | 20% | 3 | 0.67 | 82.5% |
| Split Lines | 50% | 6 | 0.43 | 59.8% |
| End-point Errors | 20% | 4 | 0.71 | 86.3% |
| End-point Errors | 50% | 5 | 0.40 | 65% |
| Combined Errors | 20% | 4 | 0.63 | 86.3% |
| Combined Errors | 50% | 5 | 0.49 | 75.2% |

Table 5.2: Matching performance using the attributed graph matching algorithm.



(a) 20% of the lines are affected by errors



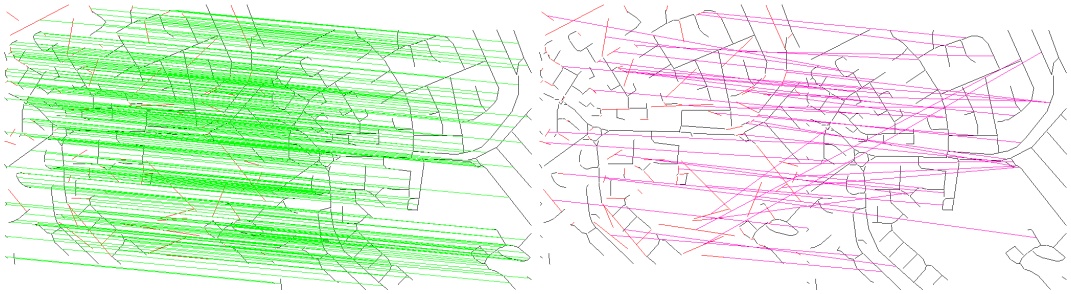
(b) 50% of the lines are affected by errors

Figure 5.9: Effect of introducing segment end-point errors on the attributed graph matching algorithm. The lines in green show the correct correspondence while the red lines indicate a bad match.

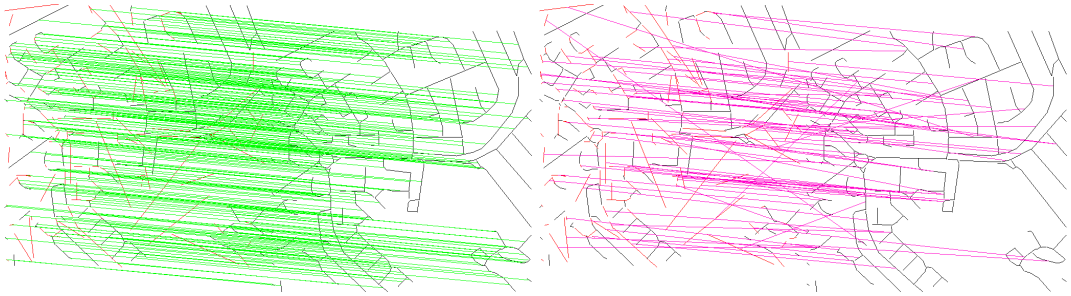
## 5.6 Hierarchical Integration

Finally, we turn our attention to how the three recognition strategies may be integrated. The idea is to use the relational histogram as a filter that can be applied to the database to limit the search via feature-set comparison. The important issue is therefore the rank threshold that can be applied to the histogram similarity measure. The threshold should be such that the probability of false rejection is low while the number of images that remains to be verified is small.

In Chapter 3 we showed the average result of querying the database with a number objects selected from a number of different categories. The main conclusion to



(a) 20% of the lines are affected by errors



(b) 50% of the lines are affected by errors

Figure 5.10: Effect of combining multiple error types on the attributed graph matching algorithm. The lines in green show the correct correspondence while the red lines indicate a bad match.

be drawn from this study is that, provided less than 20% of the line-segments are subject to error, then the database can be pruned to 1% of its original size using the relational histogram comparison. If a target pruning rate of 25% is adequate then the noise-level can be as high as 75%.

With such a level of pruning the more refined recognition process based on the fuzzy Hausdorff distance may be performed. The results presented in Chapter 4 indicate that recall accuracy of over 94% is achievable.

The limited set of model hypothesis provided by the feature-based recognition strategy provides the starting point for the graph matching algorithm. The output of this algorithm is further enhanced by a set of detailed correspondence matches

between query line-segments and their corresponding counterpart in the model.

## 5.7 Summary

We have presented a practical graph matching algorithm for data-mining in structural libraries. The main conclusion to be drawn from this study is that the combined use of structural and attribute histogram information improves both recognition performance and recall speed.

This method was designed as the last step of a hierarchical object recognition system. We showed how the histogram per node representation, combined with the graph matching method, improved on the recognition accuracy provided by the structurally gated histogram method and the local feature based fuzzy Hausdorff distance measure. The results obtained indicate that the proposed method is robust to a variety of segmentation errors.

Our study of the accuracy of the feature correspondences provided by the technique is accurate. This remains the case when a large number of the line-segments have been subject to corruption.

# Chapter 6

# Conclusions

Our aim in this thesis was to research and develop a technique for accurately and quickly retrieve and recognise objects from large libraries of line-pattern. First we investigated the state of the art in object recognition and content-based image retrieval in the literature. Our review identified a gap between retrieval and recognition techniques. Many object recognition techniques are able to perform with great accuracy at the expense of computational requirements. However, content-based retrieval techniques are able to cope with very large database. The improved computational performance is at the cost of accuracy. This is particularly true in the case of shape similarity.

Combining ideas from both image retrieval and object recognition would allow rapid retrieval and accurate recognition from large libraries of models. The solution is to devise a compact representation and a method to effectively compare large sets of object representations. The representation is in many cases limited to image or feature measurements. We believe that an object representation based on both feature attributes and relational constraints can improve the robustness and accuracy of the recognition process. Because of the complexity and computational requirements of graph matching algorithms, these are not a realistic proposition for recognition from large object libraries. However, they provide an effective way to find

correspondences between object features and should be used as such. It is from the content-based retrieval community that we have found inspiration to devise our fast line-pattern indexing algorithms. The overall idea is to employ a hierarchical recognition system. The hierarchy is based on three distinct levels each aimed at refining the indexing provided by the preceding method. In Section 6.1 we briefly describe the novel contributions of the work presented in the thesis. The main results obtained during our experiments are also mentioned here. Finally, in Section 6.2 we propose a number of future extensions which we think could further improve the effectiveness of the large scale recognition method presented here.

## 6.1 Summary

The idea of employing a hierarchy of algorithms to refine the retrieval and recognition presents a certain level of novelty. The use of hierarchy in computer vision task usually entails that a unique algorithm operates on multiple object representations (Ettinger, 1988). The hierarchy presented in this thesis is composed of three levels. The first level is aimed at coarsely pruning the large line-pattern library of the majority of unrelated or irrelevant models. This is performed by comparing structurally gated pairwise geometric histograms using the Bhattacharyya distance measure.

The use of attribute histograms has been shown to be very effective for indexing large image databases based on color (Swain, 1993; Finlayson et al., 1996) or texture (Gimelfarb and Jain, 1996). However, there has been only limited interest in the use of histograms for content-based retrieval according to shape similarity (Jain and Vailaya, 1996). It is the compactness of the histogram representations and its rapidity of comparison that makes them very attractive for large-scale object recognition. Our histogram representation differs from others (Thacker et al., 1995; DiMauro et al., 1996; Jain and Vailaya, 1996). This is because of the structural constraints placed on

the binning process during histogram creation. Additionally, the pairwise geometric attributes that compose the histograms are invariant to changes in scale, rotation and translation. A small amount of shear (change in the viewing direction) is also tolerated. Our experiments showed that the most effective combination of geometric attributes, structural gating and histogram distance measure is the following. The pairwise geometric attributes used for creating the histograms are the directed relative angle and the relative position. Restricting histogram entries to pairwise attributes that originate from connected line-segments in a six nearest-neighbours graph provides the best structure. Finally, the Bhattacharyya distance improves the level of accuracy during retrieval over the standard histogram distance measures. A number of experiments and sensitivity studies have shown how the structurally gated pairwise geometric histogram outperforms the un-gated version. Our content-based retrieval experiments indicate that a database reduction of 75% is achievable rapidly. This allows us to employ a more computationally demanding algorithm to further refine the model selection.

The second level of this hierarchy of algorithms is more closely connected to the local image features. The idea is to devise a fast method to compare attributed relational structures. This similarity measure is derived from a Bayesian consistency criterion developed by Wilson and Hancock (Wilson and Hancock, 1994) for graph matching. We have considerably simplified this measure so that it may be used in a non-iterative manner. This simplification was inspired by the work of Rucklidge and Huttenlocher on using the Hausdorff distance to locate and recognise objects based on sets of image primitives (Rucklidge, 1995). Our representation has two levels. The higher level corresponds to the set of features extracted from the image (line-segments). The lower level is the set of pairwise attributes associated with every pair of connected line-segments in the neighbourhood graph. Similarity between pairwise attributes is computed using robust error kernels. We conducted a serie of experiments aimed at assessing the recognition performance and noise sen-

sitivity of a number of algorithm alternatives. Our results show that using either the Gaussian or Huber's robust kernel with our novel relational similarity measure delivers up to 94% recognition accuracy when the neighbourhood graph is of order six. The method is shown to provide good robustness to various line segmentation errors. Moreover, when compared with both the standard and modified Hausdorff distances our measure shows important recognition improvements. This feature-based relational similarity measure allows us to further prune the set of model hypotheses to a bare minimum. This makes the use of a more elaborate iterative graph matching method possible.

The third and final level of the hierarchy is based on attributed relational graph matching. Again the line-patterns are represented using pairwise geometric attributes constructed from the edges of a neighbourhood graph. The novelty of the approach is that the pairwise attributes associated with each node in the graph are combined in a sparse two-dimensional histogram. This offers major advantages in terms of computational requirements and adaptability to other adjacency graph structures. The structural graph matching scheme employed here, is a simplified version of the technique recently reported by Finch, Wilson and Hancock (Finch et al., 1997a). The outcome of this study is an improvement on the computational requirement without any noticeable degradation of performance accuracy. An extensive set of experiments has been conducted to provide detailed information about the performance of the method. These include recognition performance evaluation for a number of alternative graph matching strategies. It was shown that the histogram per node approach provides enhanced accuracy and reduces the computational effort during matching. Again, the best structural representation for object retrieval is the neighbourhood graph of order six. The results obtained during the sensitivity study indicate adequate robustness to a number of segmentation errors. Finally, we showed that detailed correspondence matches are provided by the graph matching method even under non-negligible line-segment corruption.

## 6.2 Future Work

There are a number of ways in which the multi-level object recognition framework presented in this thesis can be extended.

The pairwise geometric attributes we have used for representing two-dimensional objects are not appropriate for representing three-dimensional objects. At each level of the hierarchy, the algorithms would benefit from using affine invariant attributes (Lamdan et al., 1988a; Rothwell et al., 1992; Maybank, 1998). Since we are interested in line-pattern based recognition the first example of an affine invariant that suggests itself is the cross ratio (Maybank, 1995). The end-point of collinear line-segments could provide one way to compute the cross-ratio. This would allow us to perform object recognition from large libraries of three-dimensional models.

At the moment, the histogram based retrieval performs a complete search of the database in order to provide a ranked set of most likely hypothesis. We believe that it is not necessary to do so. As the size of the database increases so will the recall time. The line-pattern library could be clustered into object groups according to their structurally gated pairwise geometric similarity. The query histogram would have to be compared only with a subset of the database. An alternative way of reducing the search space is to perform Eigen-analysis on the histograms. Rather than comparing the histograms directly, we could compare the first few eigenmodes.

As far as the relational similarity measure presented in Chapter 4 is concerned, an alternative local feature representation could be investigated. The results obtained during our experiments with the histogram per node graph matching algorithm suggest that local histograms of geometric attributes offer better accuracy and robustness than vector based representations. The feature-set based representation of our relational similarity measure could benefit from using a local pairwise geometric histogram for comparing the low level features.

The performance of our graph matching scheme (which uses a histogram per node) is very satisfactory. It provides a good efficiency to accuracy ratio. However, if higher levels of accuracy were required, there is a number of more computationally demanding algorithms that could be investigated. In particular, the work on Wilson and Hancock (Wilson and Hancock, 1995) on graph editing provides an improved way of dealing with clutter and noise problems by removing from the graph the nodes that have low correspondence probability.

The graph structure should also be the subject of further investigations. Firstly, we believe that a more perceptually meaningful representation of the line patterns would provide better discriminating power between objects. This could be addressed by using grouping principals derived from Gestalt psychology (Wertheimer, 1938). Alternatively, the relational structure could be automatically learned from multiple line-pattern examples representing similar objects.

Finally the bottom-up hierarchy between the various levels could be integrated more effectively. In the current recognition framework the interaction between the different levels of the hierarchy is very limited. Each algorithm simply prunes the number of hypotheses that need to be searched by the following level. It would be interesting to investigate the possibility of retaining some of the information about the ranking or similarity between objects in order to improve the retrieval. A possible approach would be to formulate the filtering of the line-patterns using Bayes decision trees.

# Bibliography

Arkin, E. M., Chew, L. P., Huttenlocher, D. P., Kedem, K., and Mitchell, J. S. B. (1991). An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):209–216.

Ashbrook, A. P., Fisher, R. B., Robertson, C., and Werghi, N. (1998). Finding surface correspondence for object recognition and registration using pairwise geometric histograms. In *Proceedings of the 5th European Conference on Computer Vision*, pages 674–686.

Ashbrook, A. P., Thacker, N. A., Rockett, P. I., and Brown, C. I. (1995). Robust recogniton of scaled shapes using pairwise geometric histograms. In *Proceedings of the 6th British Machine Vision Conference*, pages 503–512.

Ayache, N. and Faugeras, O. (1986). A new approach for the recognition and positioning of two-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):44–54.

Ballard, D. H. (1981). Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122.

Barrow, H. G. and Burstall, R. M. (1976). Subgraph isomorphism, matching relational structures and maximal cliques. *Information Processing Letters*, 4:83–84.

Barrow, H. G. and Popplestone, R. J. (1971). Relational descriptions in picture processing. *Machine Intelligence*, 5:377–396.

Bloch, I. (1996). Information combination operators for data fusion: A comparative review with classification. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 26(1):52–67.

Bray, A. J. and Hlavac, V. (1991). Properties of local geometric constraints. *Proceedings of the 2nd British Machine Vision Conference*, pages 95–103.

Bunke, H. and Messmer, B. T. (1995). Efficient attributed graph matching and its application to image analysis. In *Image Analysis and Processing, Lecture Notes in Computer Science 974*, pages 45–55. Springer-Verlag.

Canny, J. (1986). A computational approach to edge detection. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 8(6):679–698.

Christmas, W., Kittler, J., and Petrou, M. (1995). Structural matching in computer vision using probabilistic relaxation. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 17:749–764.

Cootes, T. F. and Taylor, C. J. (1995). Combining point distribution models with shape models based on finite-element analysis. *Image and Vision Computing*, 13:403–409.

Costa, M. S. and Shapiro, L. (1995). Scene analysis using appearance-based models and relational indexing. *IEEE Computer Society International Symposium on Computer Vision*, pages 103–108.

Cross, A. D. J. and Hancock, E. R. (1996). Inexact graph matching with genetic search. *Advances in Structural and Syntactic Pattern Recognition, Edited by P. Perner, P. Wang and A. Rosenfeld, in Lecture Notes in Computer Science, Springer-Verlag*, 1121:150–159.

Cross, A. D. J. and Hancock, E. R. (1998a). Graph matching with a dual-step EM algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1236–1253.

Cross, A. D. J. and Hancock, E. R. (1998b). Recovering perspective pose with dual step EM algorithm. *Advances in Neural Information Processing Systems*, 10:780–786. MIT Press.

Cross, A. D. J., Wilson, R. C., and Hancock, E. R. (1996). Genetic search for structural matching. *Proceedings of the Fourth European Conference on Computer Vision, Edited by B. Buxton and R. Cipolla, Lecture Notes in Computer Science*, 1064:514–525.

DiMauro, E. C., Cootes, T. F., Taylor, C. J., and Lanitis, A. (1996). Active shape model search using pairwise geometric histograms. *Proceedings of the 7th British Machine Vision Conference*, 1:353–362.

Dorai, C. and Jain, A. K. (1995). View organisation and matching of free-form objects. *IEEE Computer Society International Symposium on Computer Vision*, pages 25–30.

Dorai, C. and Jain, A. K. (1997). COSMOS: A representation scheme for 3D free-form objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(10):1115–1130.

Duda, R. O. and Hart, P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. *Communication in ACM*, 15:39–45.

Duda, R. O. and Hart, P. E. (1973). *Pattern Classification and Scene Analysis*. A Wiley-interscience publication.

Eakins, J. P. (1989). SAFARI - a shape retrieval system for engineering drawings. *Proceedings of 11th BCS Information Retrieval Specialist Group Research Group Colloquium on Information Retrieval*, pages 50–71.

Eakins, J. P., Shields, K., and Boardman, J. M. (1996). ARTISAN - a shape retrieval system based on boundary family indexing. *Storage and Retrieval for Image and Video Databases IV, Proc SPIE 2670, San Jose*, pages 17–28.

Ettinger, G. J. (1988). Large hierarchical object recognition using libraries of parameterized model sub-parts. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 32–41.

Evans, A. C., Thacker, N. A., and Mayhew, J. W. E. (1993). The use of geometric histograms for model-based object recognition. *Proceedings of the 4th British Machine Vision Conference*, pages 429–438.

Faloutsos, C., Barber, R., Flickner, M., Hafner, J., Niblack, W., Petkovic, D., and Equitz, W. (1994). Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3:715–729.

Farzin Mokhtarian, S. A. and Kittler, J. (1996). Robust and efficient shape indexing through curvature scale space. *Proceedings of the 7th British Machine Vision Conference*, 1:53–62.

Feng, J., Laumy, M., and Dhome, M. (1994). Inexact matching using neural networks. *Pattern Recognition in Practice IV*, pages 177–184.

Finch, A. M., Wilson, R. C., and Hancock, E. R. (1997a). Matching Delaunay graphs. *Patern Recognition*, 30:123–140.

Finch, A. M., Wilson, R. C., and Hancock, E. R. (1997b). Softening discrete relaxation. *Advances in Neural Information Processing Systems 9, Edited by M. Mozer, M. Jordan and T. Petsche, MIT Press*, pages 438–444.

Finlayson, G. D., Chatterjee, S. S., and Funt, B. V. (1996). Color angular indexing. *Proceedings of the 4th European Conference on Computer Vision*, pages 16–27.

Gevers, T. and Smeulders, A. W. M. (1992). ∑nigma: An image retrieval system. *Proceedings of the International Conference on Pattern Recognition*, pages 697–700.

Gimelfarb, G. L. and Jain, A. K. (1996). On retrieving textured images from an image database. *Pattern Recognition*, 29(9):1461–1483.

Gold, S. and Rangarajan, A. (1996). A graduated assignment algorithm for graph matching. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 18:377–388.

Gonzalez, R. C. and Wintz, P. A. (1977). Digital image processing. In *Addison-Wesley*.

Gool, L. V. V., Moons, T., and Ungureanu, D. (1996). Affine / photometric invariants for planar intensity patterns. *Proceedings of the 4th European Conference on Computer Vision*, 1:642–651.

Grimson, W. E. L. (1990). Object recognition by computer: The role of geometric constraints. *MIT Press*.

Grimson, W. E. L. and Lozano-Perez, T. (1984). Model-based recognition and localization from sparse range or tactile data. *The International Journal of Robotics Research*, 3(3):3–35.

Grimson, W. E. L. and Lozano-Perez, T. (1987). Localizing overlapping parts by searching the interpretation tree. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 9(4):469–482.

Grosky, W. I. and Mehrota, R. (1989). Image database management. *Computer*, 22(12):7–71.

Hamming, R. W. (1980). *Coding and Information Theory*. Prentice-Hall.

Hancock, E. R. (1993). Resolving edge-line ambiguities using probabilistic relaxation. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 300–306.

Hecker, Y. C. and Bolle, R. M. (1994). On geometric hashing and the generalized Hough transform. *IEEE Transactions on systems, man and cybernetics*, 24(9):1328–1338.

Henikoff, J. and Shapiro, L. G. (1990). Interesting patterns for model based machine vision. *Proceedings of the IEEE International Conference on Computer Vision*, pages 535–538.

Herault, L., Horaud, R., Veillon, F., and Niez, J. J. (1990). Symbolic image matching by similated annealing. *In Proceedings of the 1st British Machine Vision Conference*, pages 319–324.

Huet, B. (1996). Content-based retrieval in image databases. Technical report, University Of York. 1st Year Qualifying Dissertation.

Huet, B., Cross, A. D. J., and Hancock, E. R. (1998). Graph matching for shape retrieval. *Advances in Neural Information Processing Systems 11, Edited by M.J. Kearns, S.A. Solla and D.A. Cohn, MIT Press, (available May 1999),*. to appear.

Huet, B. and Hancock, E. R. (1996a). Cartographic indexing into a database of remotely sensed images. *Third IEEE Workshop on Applications of Computer Vision*, pages 8–14.

Huet, B. and Hancock, E. R. (1996b). A statistical approach to hierarchical shape indexing. *IEE Colloquium on Intelligent Image Databases*, pages 7/1–7/5.

Huet, B. and Hancock, E. R. (1996c). Structural indexing of infra-red images using statistical histogram comparison. *Third IEEE International Workshop in Image/Signal Processing*, pages 653–656.

Huet, B. and Hancock, E. R. (1997a). Pairwise representation for image database indexing. *Sixth International Conference on Image Processing and its Applications (IPA97)*, 2:492–496.

Huet, B. and Hancock, E. R. (1997b). Structurally gated pairwise geometric histograms for shape indexing. *Proceedings of the 8th British Machine Vision Conference*, pages 120–129.

Huet, B. and Hancock, E. R. (1998a). Fuzzy relational distance for large-scale object recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 138–143.

Huet, B. and Hancock, E. R. (1998b). Object recognition from large structural libraries. In *Advances in Pattern Recognition: Lecture Notes in Computer Science*, volume 1451, pages 190–199. Springer-Verlag. Sydney, Australia.

Huet, B. and Hancock, E. R. (1998c). Relational histograms for shape indexing. *Proceedings of the IEEE International Conference on Computer Vision*, pages 563–569.

Huttenlocher, D. P., Klanderman, G. A., and Rucklidge, W. J. (1993). Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863.

Huttenlocher, D. P. and Rucklidge, W. J. (1993). A multi-resolution technique for comparing images using the Hausdorff distance. *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 705–706.

Huttenlocher, D. P. and Ullman, S. (1987). Object recognition using alignment. *Proceedings of the IEEE International Conference on Computer Vision*, pages 102–111.

Huttenlocher, D. P. and Ullman, S. (1990). Recognizing solid objects by alignment with an image. *Intenational Journal of Computer Vision*, 5(2):195–212.

Illingworth, J. and Kittler, J. V. (1987). The adaptive Hough transform. *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 9(5):690–698.

Illingworth, J. and Kittler, J. V. (1988). A survey of the Hough transform. *Computer Vision Graphics and Image Processing*, 44(1):87–116.

Jain, A. K. and Vailaya, A. (1996). Image retrieval using color and shape. *Pattern Recognition*, 29(8):1233–1244.

Jong, K. A. D. and Spears, W. M. (1989). Using genetic algorithms to solve np-complete problems. *Genetic Algorithms*, pages 124–132.

Kittler, J., Christmas, W. J., and Petrou, M. (1992). Probabilistic relaxation for matching of symbolic structure. *Advances in Structural and Syntactic Pattern Recognition*, pages 471–480.

Kittler, J. and Hancock, E. R. (1989). Combining evidence in probabilistic relaxation. *International Journal of Pattern Recognition and Artificial Intelligence*, 3(1):29–51.

Knoll, T. F. and Jain, R. C. (1986). Recognizing partially visible objects using feature indexed hyphotheses. *IEEE Transactions on Robotics and Automation*, 21:3–13.

Koenderink, J. J. and vanDoorn, A. J. (1992). Surface shape and curvature scales. *Image and Vision Computing*, 10:557–565.

Koffka, K. (1935). Principles of gestalt psychology. *Harcourt Brace, New York*.

Kuner, P. and Ueberreiter, B. (1988). Pattern recognition by graph matching - combinatorial versus continous optimization. *International Journal of Pattern Recognition and Artificial Intelligence*, 2(3):527–542.

Kupeev, K. Y. and Wolfson, H. J. (1994). On shape similarity. In *Proceedings of the 12th. International Conference on Pattern Recognition*, pages 227–231. IEEE Computer Society Press.

Lamdan, Y., Schwartz, J. T., and Wolfson, H. J. (1988a). Object recognition by affine invariant matching. In *Proceedings of the conference on Computer Vision and Pattern Recognition*, pages 335–344.

Lamdan, Y., Schwartz, J. T., and Wolfson, H. J. (1988b). On recognition of 3D objects from 2D images. *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1407–1413.

Lamdan, Y. and Wolfson, H. J. (1988). Geometric hashing: A general and efficient model-based recognition scheme. *Proceedings of the IEEE International Conference on Computer Vision*, pages 238–249.

Lowe, D. G. (1987). Three-dimensional object recognition from two-dimensional images. *Artificial Intelligence*, 31(3):355–395.

Marsicoi, M. D., Cinque, L., and Levialdi, S. (1997). Indexing pictorial documents by their content: a survey of current techniques. *Image and Vision Computing*, 15:119–141.

Maybank, S. J. (1995). Probabilistic analysis of the application of the cross ratio to model-based vision. *International Journal of Computer Vision*, 16(1):5–33.

Maybank, S. J. (1998). Relation between 3D invariants and 2D invariants. *Image Vision and Computing*, 16(1):13–20.

Messmer, B. and Bunke, H. (1994). Efficient error-tolerant subgraph isomorphism detection. *Shape, Structure and Pattern Recognition*, pages 231–240.

Mumford, D. (1991). Mathematical theories of shape: Do they model perception? *Geometric Methods in Computer Vision*, 1570:2–10. SPIE.

Murase, H. and Nayar, S. K. (1995). Visual learning and recognition of 3-d objects from appearance. *International Journal of Computer Vision*, 14:5–24.

Nayar, S. K. and Bolle, R. M. (1993). Reflectance ratio: A photometric invariant for object recognition. *IEEE International Conference on Computer Vision*, pages 280–285.

Niblack, W., Barber, R., Equitz, W., Flickner, M., Glasman, E., Petkovic, D., Yanker, P., Faloutsos, C., and Taubin, G. (1993). The QBIC project: Querying images by content using color, texture and shape. *In Proc. SPIE Conf. on Storage and Retrieval of Image and Video Databases*, 1908:173–187. San Jose, California.

Ogniewicz, R. L. and Kubler, O. (1995). Hierarchic Voronoi skeletons. *Pattern Recognition*, 28(3):343–359.

Olson, C. F. (1994). Time and space efficient pose clustering. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 251–258.

Pentland, A. P., Picard, R. W., and Scarloff, S. (1994). Photobook: tools for content-based manipulation of image databases. *Storage and Retrieval for Image and Video Database II*, pages 34–47. San Jose, California.

Pentland, A. P., Picard, R. W., and Sclaroff, S. (1996). Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233–254.

Picard, R. W. (1995). Light-years from Lena: Video and image libraries of the future. *International Conference on Image Processing*, 1:310–313.

Picard, R. W. and Liu, F. (1994). A new Wold ordering image similarity. *IEEE Conference on Acoustics, Speech, and Signal Processing*, 5:129–132. Adelaide, Australia.

Picard, R. W. and Pentland, A. P. (1996). Introduction to the special section on digital libraries: Representation and retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):769–770.

Princen, J. P., Illingworth, J., and Kittler, J. V. (1992). A formal definition of the Hough transform: Properties and relationships. *Journal of Mathematical Imaging and Vision*, 1(1):153–168.

Procter, S. and Illingworth, J. (1997). Foresight: Fast object recognition using geometric hashing with edge-triple features. In *Proceedings of the International Conference on Image Processing*, pages I:889–893.

Prokop, R. J. and Reeves, A. P. (1992). A survey of moment-based techniques for unoccluded object representationand recognition. *Computer Vision Graphics and Image Processing: Graphical Models and Image Processing*, 54(5):438–460.

Reed, T. R. and DuBuf, J. M. H. (1993). A review of recent texture segmentation and feature-extraction techniques. *Computer Vision Graphics and Image Processing: Image Understanding*, 57(3):359–372.

Rigoutsos, I. and Hummel, R. (1995). A bayesian approach to model matching with geometric hashing. *Computer Vision and Image Understanding*, 62(1):11.

Rosin, P. L. and West, G. A. W. (1989). Segmentation of edges into lines and arcs. *Image and Vision Computing*, 7(2):109–114.

Rothwell, C. A., Zisserman, A., Forsyth, D., and Mundy, J. (1992). Canonical frames for planar object recognition. *European Conference on Computer Vision*, pages 757–772.

Rucklidge, W. J. (1995). Locating objects using the Hausdorff distance. *IEEE International Conference on Computer Vision*, pages 457–464.

Sanfeliu, A. and Fu, K. S. (1983). A distance measure between attributed relational graph. *IEEE SMC*, 13:353–362.

Santini, S. and Jain, R. (1995). Similarity matching. *Second Asian Conference on Computer Vision*, 3(2):544–548.

Scassellati, B., Alexopoulos, S., and Flickner, M. (1994). Retrieving images by 2D shape: a comparison of computation methods with human perceptual judg-

ments. *Storage and Retrieval for Image and Video Database II*, pages 2–14. San Jose, California.

Schiele, B. and Crowley, J. L. (1996). Probabilistic object recognition using multidimensional receptive field histograms. *Proceedings of the 13th International Conference on Pattern Recognition*, 2:50–54.

Sclaroff, S. and Pentland, A. (1993). A finite-element framework for correspondence and matching. *Proceedings of the 4th International Conference on Computer Vision*, pages 308–313.

Sclaroff, S. and Pentland, A. (1995). Modal matching for correspondence and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(6):545–561.

Scott, G. L. and Longuet-Higgins, H. C. (1991). An algorithm for associating the features of two images. *Proc. Royal Soc. London Series B-Biological*, 244:21–26.

Sengupta, K. and Boyer, K. L. (1993). Information theoretic clustering of large structural modelbases. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 174–179.

Sengupta, K. and Boyer, K. L. (1995). Organising large structural databases. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(4):321–332.

Shapiro, L. G. and Haralick, R. M. (1985). A metric for comparing relational descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):90–94.

Shapiro, L. S. and Brady, M. (1992). Feature-based correspondence: An eigenvector approach. *Image and Vision Computing*, 10(5):283–288.

Shapiro, L. S. and Brady, M. (1995). Rejecting outliers and estimating errors in an orthogonal-regression framework. *Phil. Trans. Royal Society A.*, 350:403–439.

Siddiqi, K., Shokoufandeh, A., Dickinson, S. J., and Zucker, S. W. (1998). Shock graphs and shape matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 222–229.

Stein, F. and Medioni, G. (1990). Efficient two dimensional object recognition. *in Proceedings of the International Conference on Pattern Recognition*, pages 13–17.

Stein, F. and Medioni, G. (1992). Structural indexing: Efficient 2D object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(12):1192–1204.

Stricker, M. and Swain, M. (1994). The capacity of color histogram indexing. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 704–708.

Suetens, P., Fua, P., and Hanson, A. J. (1992). Computational strategies for object recognition. *ACM Computing Surveys*, 24:5–61.

Swain, M. J. (1993). Interactive indexing into image databases. *Image and Vision Storage and Retrieval*, pages 95–103.

Swain, M. J. and Ballard, D. H. (1990). Indexing via colour histograms. *Third International Conference on Computer Vision*, pages 390–393.

Swain, M. J. and Ballard, D. H. (1991). Color indexing. *International Journal of Computer Vision*, 7(1):11–32.

Taubin, G. and Cooper, D. B. (1991). Recognition and positionning of rigid objects using algebraic moment invariants. *Geometric Methods in Computer Vision, SPIE*, 1570:175–186.

Thacker, N. A., Riocreux, P. A., and Yates, R. B. (1995). Assessing the completeness properties of pairwise geometric histograms. *Image and Vision Computing*, 13(5):423–429.

Tsai, F. C. D. (1994). Geometric hashing with line features. *Pattern Recognition*, 27(3):377–389.

Tuceryan, M. and Chorzempa, T. (1991). Relative sensitivity of a family of closest-point graphs in computer vision applications. *Pattern Recognition*, 24(5):361–373.

Turk, M. and Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–96.

Ullman, J. R. (1976). An algorithm for subgraph isomorphism. *Journal of the Association for Computing Machinery*, 9:757–768.

Ullman, S. (1979). The interpretation of visual motion. *MIT Press Cambridge, Mass.*

Viola, P. A. and Wells, W. M. (1997). Alignment by maximization of mutual information. *International Journal on Computer Vision*, 24(2):137–154.

Wang, H. and Brady, M. (1995). Real-time corner detection algorithm for motion estimation. *Image anf Vision Computing*, 13(9):695–703.

Wertheimer, M. (1938). Laws of organization in perceptual forms. *W.D. Ellis (ed.), A source book of Gestalt Psychology, Harcourt Brace, New York*, pages 71–88.

Wilson, R. and Hancock, E. R. (1997). Structural matching by discrete relaxation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):634–648.

Wilson, R. C., Cross, A. D. J., and Hancock, E. R. (1996). Sensitivity analysis for structural matching. *Proceedings of the 13th International Conference on Pattern Recognition*, 1:62–66.

Wilson, R. C. and Hancock, E. R. (1994). Relational matching by discrete relaxation. *Image and Vision Computing*, 13:411–422.

Wilson, R. C. and Hancock, E. R. (1995). Relational matching with active graph structures. *Fifth International Conference on Computer Vision*, pages 450–456.

Worthington, P. L., Huet, B., and Hancock, E. R. (1998). Appearance-based object recognition using shape-from-shading. In *Proceedings of the International Conference on Pattern Recognition*, pages 412–416.