UNIVERSITY
of
GLASGOW

Trademark Image Retrieval by Local Features

Paijit Kochakornjarupong

B.Sc. (Computer Science)
M.Eng. (Electrical Engineering)

Submitted in fulfilment of the requirement for the Degree of Doctor of Philosophy

College of Social Sciences
University of Glasgow

June 2011

Abstract

The challenge of abstract trademark image retrieval as a test of machine vision algorithms has attracted considerable research interest in the past decade. Current operational trademark retrieval systems involve manual annotation of the images (the current 'gold standard'). Accordingly, current systems require a substantial amount of time and labour to access, and are therefore expensive to operate. This thesis focuses on the development of algorithms that mimic aspects of human visual perception in order to retrieve similar abstract trademark images automatically. A significant category of trademark images are typically highly stylised, comprising a collection of distinctive graphical elements that often include geometric shapes. Therefore, in order to compare the similarity of such images the principal aim of this research has been to develop a method for solving the partial matching and shape perception problem.

There are few useful techniques for partial shape matching in the context of trademark retrieval, because those existing techniques tend not to support multi-component retrieval. When this work was initiated most trademark image retrieval systems represented images by means of global features, which are not suited to solving the partial matching problem. Instead, the author has investigated the use of local image features as a means to finding similarities between trademark images that only partially match in terms of their sub-components. During the course of this work, it has been established that the Harris and Chabat detectors could potentially perform sufficiently well to serve as the basis for local feature extraction in trademark image retrieval. Early findings in this investigation indicated that the well established SIFT (Scale Invariant Feature Transform) local features, based on the Harris detector, could potentially serve as an adequate underlying local representation for matching trademark images.

There are few researchers who have used mechanisms based on human perception for trademark image retrieval, implying that the shape representations utilised in the past to solve this problem do not necessarily reflect the shapes contained in these image, as characterised by human perception. In response, a

practical approach to trademark image retrieval by perceptual grouping has been developed based on defining meta-features that are calculated from the spatial configurations of SIFT local image features. This new technique measures certain visual properties of the appearance of images containing multiple graphical elements and supports perceptual grouping by exploiting the non-accidental properties of their configuration.

Our validation experiments indicated that we were indeed able to capture and quantify the differences in the global arrangement of sub-components evident when comparing stylised images in terms of their visual appearance properties. Such visual appearance properties, measured using 17 of the proposed meta-features, include relative sub-component proximity, similarity, rotation and symmetry. Similar work on meta-features, based on the above Gestalt proximity, similarity, and simplicity groupings of local features, had not been reported in the current computer vision literature at the time of undertaking this work.

We decided to adopted relevance feedback to allow the visual appearance properties of relevant and non-relevant images returned in response to a query to be determined by example. Since limited training data is available when constructing a relevance classifier by means of user supplied relevance feedback, the intrinsically non-parametric machine learning algorithm ID3 (Iterative Dichotomiser 3) was selected to construct decision trees by means of dynamic rule induction. We believe that the above approach to capturing high-level visual concepts, encoded by means of meta-features specified by example through relevance feedback and decision tree classification, to support flexible trademark image retrieval and to be wholly novel.

The retrieval performance the above system was compared with two other state-of-the-art image trademark retrieval systems: Artisan developed by Eakins (Eakins et al., 1998) and a system developed by Jiang (Jiang et al., 2006). Using relevance feedback, our system achieves higher average normalised precision than either of the systems developed by Eakins' or Jiang. However, while our trademark image query and database set is based on an image dataset used by Eakins, we employed different numbers of images. It was not possible to access to the same query set and image database used in the evaluation of Jiang's trademark

image retrieval system evaluation. Despite these differences in evaluation methodology, our approach would appear to have the potential to improve retrieval effectiveness.

TABLE OF CONTENTS

List of Tables

List of Figures

Acknowledgements

There are many people without whom this thesis could not be finished. I am taking this chance to indicate my deepest gratefulness to these people. I attempt my best and apologise dearly for those whom I overpass.

I would first like to take this opportunity to articulate my sincere gratitude to my supervisors, Dr. Paul Siebert and Prof. Paul Brna for their kindly help, advice, support, and encouragement over the years. I also would like to thank Prof. John Eakins, Dr. Paul Vickers, and Jon Riley who supervised and supported in the initial years of this research at University of Northumbria. I would like to thank Prof. Ela Claridge and Prof. Joemon Jose for their suggestions. Thank you very much for their patient to my poor English.

I would like to thank Royal Thai government for financial support, and Computer Engineering department at Prince of Songkla University for allowing me to be on leave to follow this PhD research.

I also thanks to all my Thai friends at University of Glasgow, University of Edinburgh, University of Northumbria, and my colleague in Thailand who have supported me and help me in any forms. I would like to thank Dr. Andrew Davison, Dr. Robert Elz, and Dr. Pichaya Tandayya for their proofreading through this thesis.

Especially, I would like to express my deepest thanks and appreciations to my mother, father, brothers, and sister. They give me constant encouragement, selfless support, and kindly solicitude. Here, I would like to share all my achievements with them.

Lastly I would like to thank my wife Duenpen for the constant encouragement, and immense support.

Thank you all,

Paijit Kochakornjarupong

Author's Declaration

I declare that, except where explicit reference is made to the contribution of others, that this dissertation is the result of my own work and has not been submitted for any other degree at the University of Glasgow or any other institution.

<div align="right">

Paijit Kochakornjarupong
June 2011

</div>

Chapter 1

Introduction

An important task of computer vision is to produce machines that perceive images. The search for methods that analyse images in an accurate manner has led the author to examine issues connected with the recognition of abstract trademarks. The approach has been made more specific by embedding the recognition problem within a context that requires that images be compared with one another to identify which images are similar and which are not. This context provides some advantages owing to its specificity. This chapter provides an outline of the problem, its context, and the background that led up to our approach. The chapter also features a brief explanation of the nature of the contributions made, and concludes with a description of how the remainder of the thesis is organized.

1.1 Motivation

The problem addressed here can be phrased as the question "how can we help people identify a putative trademark as being sufficiently original?" To do this, there is a need to address how to analyse a proposed trademark, how to identify those trademarks that are most similar, and how to organize their presentation to the best effect. Examples of abstract trademark images are given in Figure 1.1, along with similar trademark images.

Pattern recognition is a fundamental problem in computer vision, and trademark image retrieval is one of the most challenging in the area of Content-based Image Retrieval (Eakins et al., 2001). The particular problem of trademark image retrieval has been investigated for over two decades, and continues to attract research interest. There are two main reasons: the commercial potential offered by practical recognition systems, and the challenges that trademark

images provide as a test of machine vision algorithms. The potential applications are appealing, for example, machines could interpret their surroundings via cameras, making sensible human-like decisions. At the centre of interest is the development of visual perception by computer, where a computer can recognize images and select similar images. The list below gives examples of potential applications of computer perception technology.

Patent control: To register a trademark gives an advantage to both traders and customers; traders can protect their goods by a distinct mark; customers can recognize a genuine product by its trademark. The number of trademarks vary in each country from thousands to hundreds of thousands, and is gradually rising. A system utilizing this technology could help patent offices distinguish putative trademarks.

Image database retrieval: Many images in offices, web sites, and home computers are stored without manual annotation. It is difficult for a human to search such images, so a system using this technology could search the required images faster.

Image design: To understand how to perceive similar images by computer could also be used to measure how distinct the images are. Such a system could help image designers create more distinctive images.

(a)

(b)

(c)
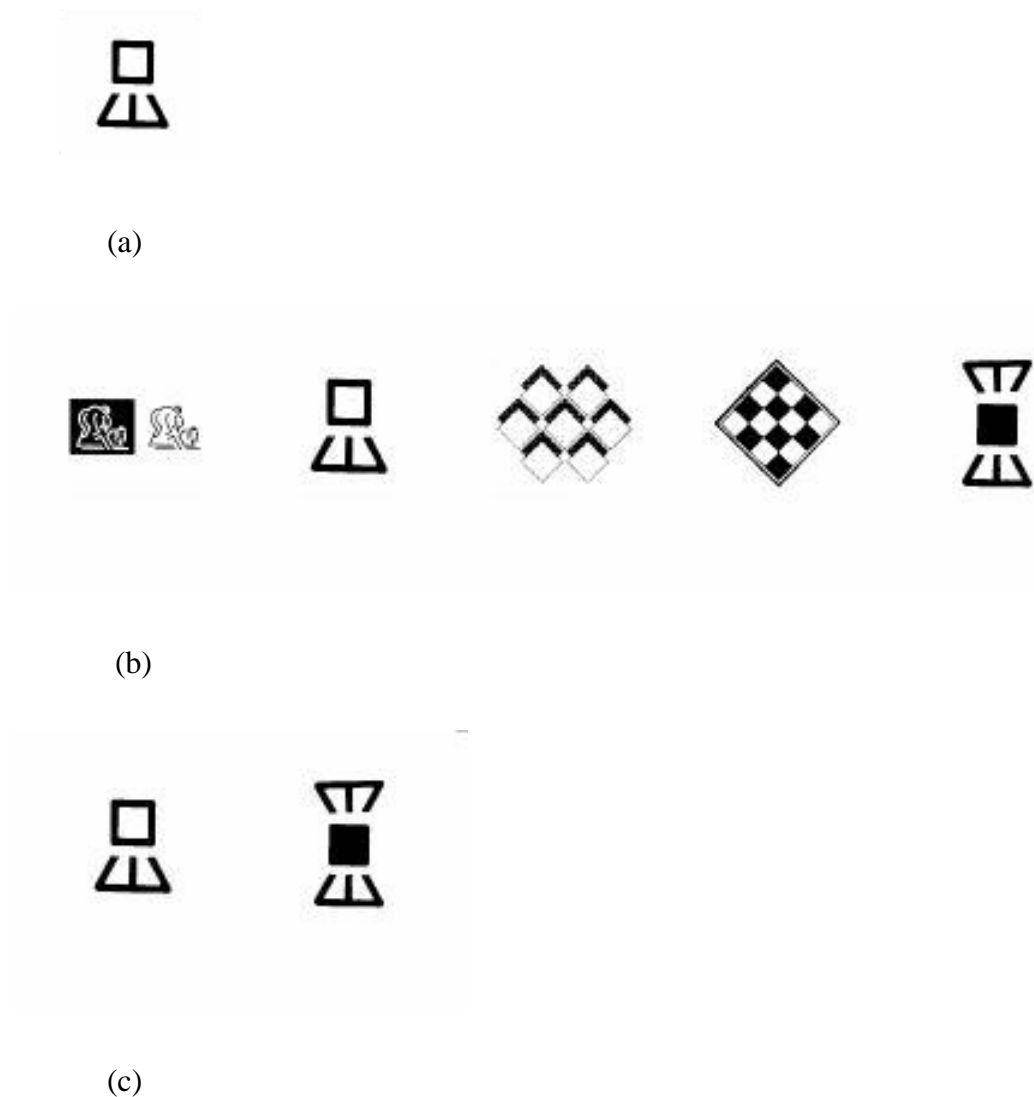
Figure 1.1 An illustration of the situation in which the work reported in this thesis is embedded. Similar trademark images are retrieved based on the analysis of a query image. (a) an example query image, (b) a simple dataset of trademark images, and (c) the corresponding similar trademark images that might be retrieved.

1.2 Problem of existing systems

The challenge of trademark image retrieval has been discussed by many researchers (Mehrotra & Gary, 1995; Wu et al., 1996; Eakins et al., 1998; Jain & Vailaya, 1998; Alwis & Austin, 1999; Chan & King, 1999; Ravela & Manmatha, 1999; Safar et al., 1999; Shih & Chen, 2001; Yin & Yeh, 2002; Gori et al., 2003). However, there is no completely satisfactory system that is currently in use in a patent office.

Due to the complexity of the task, we investigate the approaches taken in

the past and simplify the problem of trademark registration. These include the following limitations:

- Current trademark registration uses a lot of time and labour, because it uses a keyword search system to classify the trademark, and employs examiners to distinguish trademarks. Most trademarks are stored in electronic files (Claus, 2002), but many registries only use manual file search, and some use automated name or coding search. A traditional trademark retrieval system uses keyword search, for example, Tess, and the UK trademark search (The UK patent office, 2001; United state patent and trademark office, 2004). Therefore, a more effective and automatic system is needed.

- An abstract trademark is difficult to classify by keywords because it consists of complex visual elements. This kind of trademark is usually a geometric figure, known as an abstract type. Such trademarks are well distinguished by their elements rather than their meaning, so content-based image retrieval is required. Significantly, abstract trademarks are better suited for image content-based classification rather than manually-based classification (Eakins et al., 1998).

- Abstract trademark image retrieval by content-based image retrieval is obviously important, because it requires that the database be searched by means of an image since the search problem cannot be solved by textual queries alone. Visual perception is important for distinguishing shapes in abstract trademarks because they contain graphical elements. However, there is no current technique for computational vision perception that is suitable for abstract trademark retrieval. This is the main problem to be investigated in this research. From the Gestalt laws of perceptual grouping, shape is very important in human visual judgment. Furthermore, Biederman (1987) stated that humans recognize shape only by distinctive elements. Hence, shape similarity matching is essential in trademark image retrieval. However, few methods have yet been applied to find the shape similarity of trademark images by computer.

1.3 Objectives

From the description above of the problems, the principal drawback of current systems is their need to describe abstract trademark images by keywords. Content-based image retrieval is a possible solution, but the literature shows that there are knowledge gaps that need to be addressed. First, few techniques are suitable for partial shape matching because they do not support multi-component retrieval. Second, many techniques need an exact image segmentation, which continues to be an unsolved problem. Third, there are few researchers attempt to apply principles derived from human perception for shape retrieval and without which their adpoted shape representations may not reflect percpetually meaningful configurations of the image's component. Finally, global features are not suitable for retrieving occluded or connected components in an image.

Image similarity estimation based on *interest points* has the potential to be used in abstract trademark retrieval since it has also been used successfully in image retrieval (Schmid and Mohr,1997; Lowe,1999; Wolf,2000; Sebe,2001). Importantly, interest points support partial matching and local features and are therefore potentially roboust to partial occlusions within the compared images. This addresses the main aim of this research, which is to develop a method for solving the partial matching and shape perception problems. We believe that using interest points in trademark image retrieval can improve the efficiency of an abstract trademark image retrieval system. The main objectives of this research are to investigate:

1. How we can use interest points to distinguish trademark images?
2. Which interest point techniques are most accurate when applied to distorted trademark images (noise, rotation, and scale)?
3. How we can use perceptual grouping methods to group interest points and represent these as a shape descriptor?
4. What techniques can exploit shape descriptors when retrieving abstract trademark images?

We explain how to fulfil our objectives in Chapter 2. In the next section, we summarise the topics involved in building a trademark image retrieval system.

1.4 Constructing a trademark image retrieval system

Our system measures the similarity of abstract trademarks, which raises the problems of partial matching and shape perception by computer. The details of proposed approaches are explained in Chapter 3, but are summarized here.

- The Gestalt laws of organization show that shape is very important in human visual similarity judgment. Also Biederman (1987) showed that humans can recognize shapes by only distinctive elements, which indicates that shape similarity judgement could plays a vital role in trademark image retrieval. However, there is no evident method for finding the shape similarity of trademark images by computer. Human visual perceptual concepts suggest how to imitate human image similarity by computer, and non-accidental properties have potential benefits, which provide the motivation for interest point extraction.

- Biederman's concepts motivate the idea of using interest points in shape retrieval because they have high information content and are robust in relation to partial visibility. Interest points should offer advantages when supporting human perception of shape and for specifying local features. This view is supported by many researchers who have used interest points for successful object recognition. For example, the SIFT detector is both robust detector and can generate local features (Lowe, 1999). We investigate interest point detectors in Chapters 4 and 5.

- We want to find techniques that use interest points and local features to evaluate similar images, which introduce local features that reflect human perception. We propose 27 local features for measuring perceptual grouping, and select appropriate features for calculating similarity according to relevance feedback. The technique is described in more details in Chapters 6 and 7.

- Trademark retrieval requires user to judgement of similar images. To help the user, the system should only require necessary user feedback. We investigate processes for applying relevance feedback to forming feature vectors that encapsulate the visual perception in Chapters 6, 7, and 8.

The contributions of the thesis are summarized in the next section.

1.5 Contributions of this thesis

This thesis addresses various specific problems in computer vision - partial matching, shape perception, and similarity judgement, offering methods that can benefit the computer vision communities. The main contributions of this thesis are as followings:

1.5.1 Analysis and application of point matching to trademark image matching and retrieval

We examine the application of interest points to abstract trademark image retrieval. Interest points have been successfully used to recognise objects (Lowe, 1999; Wolf, 2000; Sebe & Lew, 2003) but there is no research on how to retrieve similar abstract trademark images by interest points since most systems concentrate on global features rather than local features. Local features can reflect both local and global image characteristics, but global features are preferred since local features may need more computation time and complicated methods. The problems are where to apply local feature extraction, what local features should be utilized, and how to use the extracted local features. In the first case, we propose to use interest points because they support Biederman's concept that humans recognize images by distinctive elements. We will show in Chapter 4 that many detectors can extract the same areas in transformed trademark images, and the system can retrieve similar trademark images based on interest points. The technique is overviewed in Chapter 2, and evaluated in Chapter 8.

1.5.2 Point-based grouping of local features for trademark image retrieval

We will represent shapes in abstract trademark images by using local features. Visual perception plays an important role in human similarity judgment (Goldmeier, 1972; Eakins, 1997). In addition, shape is important for identifying abstract trademark images which contain multiple graphical elements. In Chapter 6, we propose 27 features based on interest points which utilize the computational vision perception of shape. The features are also grouped by Principal Component Analysis (PCA) to eliminate redundant ones. The feature types are global rotation, global pattern similarity, global pattern overlap, spread of the match points, structural configuration of the match points, scale different of matched pattern,

self similarity, symmetry, and sub-component similarity.

1.5.3 Relevance feedback based on point-based grouping of local features

We propose and evaluate our technique based on the use of relevance feedback and rule tree classification, generated by dynamic rule induction, in combination with algporithsm for simulating aspects of visual perception used for measuring image similarity. In other words, relevance feedback and decision tree classification can serve to imitate visual perceptual judgement by machine.

Minor contributions include:

1.5.4 Study of interest point detectors

The evaluation of interest point detectors is reported in Chapter 4. We investigate and evaluate interest point detectors that calculate interest points directly from an image. We choose four effective detectors for our experiments. They are the Harris detector, Chabat detector, SUSAN detector, and Wavelet-based detector. We measure the repeatability of these interest point detectors with the transformed trademark images. The results show that the Harris detector has the best repeatability and Chabat detector also offers good results, with more than 50% repeatability. The two detectors have the potential to be used in trademark image retrieval.

1.6 Thesis organization

The remainder of the thesis is organised as follows:

Chapter 2 reviews methods of trademark image retrieval, and discusses how trademark image retrieval can be improved.

Chapter 3 describes our proposed research for constructing a trademark image retrieval system.

Chapter 4 explains interest point detectors, and measures their suitability for trademark image retrieval.

Chapter 5 examines the SIFT detector for trademark image retrieval.

Chapter 6 proposes local features, and outlines the relevance feedback approach for perceptual grouping.

Chapter 7 reports system implementation for measuring perceptual grouping.

Chapter 8 investigates system efficiency for measuring perceptual grouping.

Chapter 9 summarises the contributions of this thesis, and discusses future work.

Chapter 2

Related work

This chapter explains the current techniques for trademark image retrieval, and investigates potential methods for improving an abstract trademark image retrieval system. We are interested in using our understanding of human visual perception in order to improve the retrieval of similar images. In this research, we concentrate on abstract trademark image retrieval since such images contain rich geometric elements that are grouped in significant ways.

2.1 Introduction

Worldwide, countries have increased their activities in relation to trademark registration (Claus, 2002). Trademarks have been used for a long time, the first was organized in London in 1876 (Winterfeldt et al., 2002; The UK patent office, 2003). Trademark registration protects goods and services e.g. distinguishing their owners from, and making it easier for recognize products. In other words, a trademark is both a marketing tool and form of intellectual property. The number of trademarks in each country varies from thousands to hundreds of thousands, and is rising gradually. Currently, all trademark retrieval systems in practical use are manual systems, so are both labour-intensive and time-consuming. The following sections describe the problems of trademark registration and trademark image retrieval.

Trademark image retrieval finds the similarity between trademark images by extracting and matching shape features of each trademark. Shape features are used in many trademark retrieval systems, because they are graphical figures to be distinguished. Also, local features may be useful because many trademarks consist of multiple components. Local features can be used in partial matching which supports the occlusion of multiple components. However, the location of local features is important for matching, and can affect retrieval efficiency.

Finding the most suitable method for extracting those points is vital.

2.2 Trademark registration

A trademark is any sign that uniquely distinguishes it from other trademarks – it may consist of words, symbols, abstract designs, or a combination of these (The UK patent office, 2004). Non-traditional trademarks can be an appearance, a shape, sounds, scents, taste, or even touch (INTA, 2003), and many trademarks are images (Eakins et al., 1996; Eakins et al., 1996).

A trademark image can be divided into three categories: words, devices, and composition trademarks. The word trademark consists of words only, for example as show in Figure 2.1(a). The device trademark is a graphical design trademark that does not contain characters, such as Figure 2.1(b). The composition trademark contains both words and graphical design components, as shown in Figure 2.1(c).

| a | b | c |
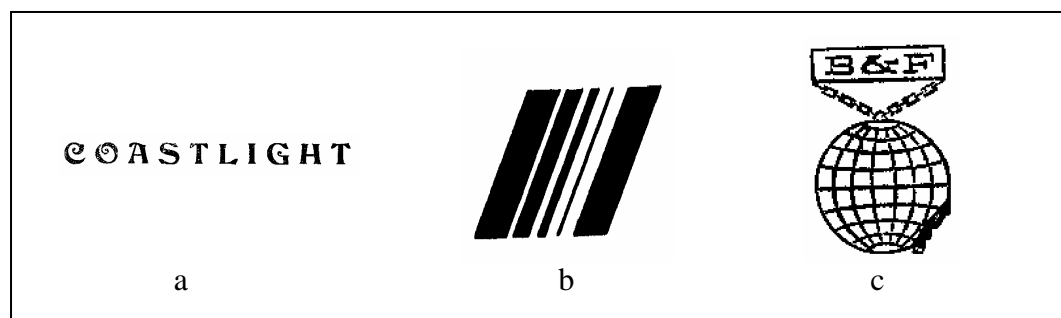
Figure 2.1 Examples of (a) word, (b) device, and (c) composition trademarks

Registering a trademark benefits both traders and customers: traders protect their goods by a distinct mark, while customers can recognize a genuine product by its trademark. Registering a trademark can be separated into five main steps: pre-applying, applying, examining, publishing, and certification.

Every trademark must specify its goods or services class. Most patent offices use the international goods and services Nice classification (Claus, 2002). It consists of 34 goods classes and 11 services made up of about 10,000 goods definitions and 1,000 services definitions. This classification is now up to its eighth edition since starting in January 2002 (WIPO, 2004). Goods' classes examples include chemicals used in industry, paints, machines and machine tools, vehicles, and rubber (WIPO, 2004). Services classes examples include

advertising, insurance, telecommunications, education, and medical services (WIPO, 2004).

The trademark class relates to its product type, and can be registered in many classes although that requires additional payment. After specifying a good's class, the mark property must be checked, which must distinguish it from other marks in the same class. The mark should not add any deceptive function to increase the product's value, nor reserved symbols, anything offensive or illegal (The UK patent office, 2001). There are private agents or advisory services associated with some patent offices, such as, the UK patent offices, which support the classifying process, but they are costly and limited to a text-based system. Automatic image classification is required.

Applying to a patent office for registration includes paying a fee and completing an application form. The registered mark is provided with national protection, but can be extended internationally by applying the Madrid protocol through WIPO (the World Intellectual Property Organisation)(The UK patent office, 2001). After submitting the mark, no changes to it are allowed. The patent office may acknowledge the application within two months.

The examining process is performed by the trademark registry office at the patent office, which is responsible for proving that a trademark is different from all others. Most patent offices use the Vienna classification to classify the figurative elements of a trademark. The classification consists of 29 categories, 144 divisions, and 1,667 sections, and is currently in its sixth edition since being created in January 2008. An example of the Vienna classification is shown in Table 2.1.

Table 2.1 The Vienna classification of the figurative elements of a trademark

The Vienna classification					
1. Celestial bodies	2. Human beings	3. Animals	…	26.Geometrical figures and solids	…
- Stars - Sun - Moon - …	…	…	…	- Circles, ellipses	…
				- Triangles	
				- One triangle - Two triangles, one inside the other - Several triangles, juxtaposed, joined or intersecting - …	…
				- … - Other polygons - Lines, bands - … - Geometrical solids	…

Each trademark is assigned component names selected by the examiners, coded by its shape features and object elements (Eakins et al., 1997). A logical combined code identifies a new trademark from all previous trademarks, so the processes involve much labour and time. The annotation may be a major problem for some types of trademark, such as device marks (It is called the identification process problem).

The patent office publishes approved trademarks in its official Gazette for public objection, which may take about three months.

If the published mark has no objections, the patent office will send a registration certificate to confirm the end of the process. The certificate is valid for ten years from the registered date, and can be renewed every ten year.

To sum up, in terms of processes, the goods' classification process is dependent on the applicant product. However, the identifying mark is more complicated than the classification, so may cause delays. The second process is submitting the document to the desire patent office, and has a specified response time. The third process is identifies the mark. The fourth process and fifth process have obvious time limits corresponded by objections caused by the similarity of the mask to others. There have been many comments from unsatisfied EU and USA applicants complaining about the long processing time (Annand, 2000), so

the identification process needs to be investigated in more detail.

Most trademarks are stored electronically (Claus, 2002), but many registries only use manual file search, perhaps with automated name or coding search. A traditional trademark retrieval system uses keyword searching, such as, Tess and UK trademark search (The UK patent office, 2001; United state patent and trademark office, 2004). It performs best on text-based marks (word and composition mark), and is poorly suited to trademarks that consist of complex visual elements and geometrical figures, often known as abstract types.

Though the Vienna classification contains of many keywords, it has a problem explaining abstract trademarks. Many elements are difficult to describe by words, indicating that keyword-based classification is inadequate and unsuitable. Gundersen (2000) states that an abstract design mark is the most difficult to represent by words. For example, Figure 2.2(a) consists of line segments and other components that are hard to explain by words. Figure 2.2(b) consists of one component that cannot be explained by words in a simple, clear meaning.
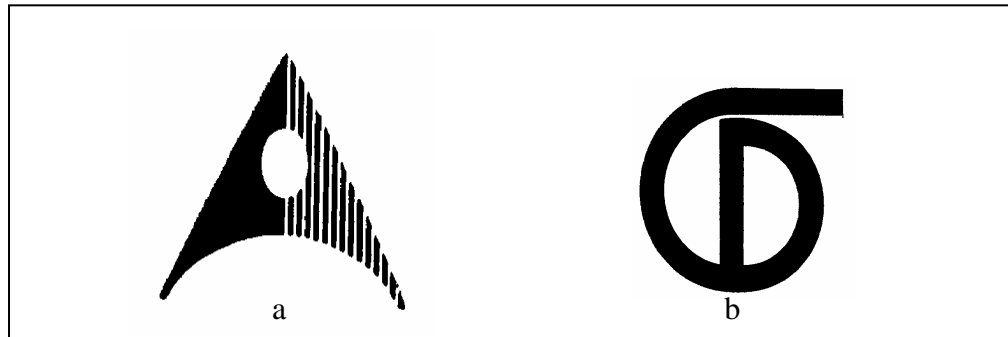
Figure 2.2 Examples of abstract trademarks

Device marks are better suited to content-based image retrieval, because they can be well distinguished by the primitive features within them, so it is important to find a technique for clearly identifying the abstract image in a device mark. The abstract class in the Vienna classification (class 26) is shown in Table 2.1.

Abstract trademarks are usually registered as two-dimensional black and white images, for maximum protection. If it is necessary, the mark may be registered in a colour series, and include monochrome colour.

Abstract marks are better suited for image content-based classification than for manual-based classification (Eakins et al., 1998). Since the image content is easier to identify than describing by keywords. Shape retrieval is important for distinguishing abstract trademarks because they contain graphical elements, thereby decreasing search time and making the process more automatic. In the next section, we will investigate content-based image retrieval for trademarks.

2.3 An Overview of Trademark image retrieval by image content

Trademark image retrieval systems continue to use keywords derived from trademark components. Most patent offices still use the Vienna classification to register trademark images, which consists of classes and sub classes of figurative elements for distinguishing trademark images (WIPO, 2004). However, a text-based system is time-consuming and intensively labour-intensive. Furthermore, an annotation-based system is not suitable for abstract trademark image retrieval because their content depends on intrinsic forms. Many abstract trademarks contain components that require additional keywords to explain.

Many researchers use image features to identify trademarks without requiring a textual description. However, each feature has different discrimination power to identify trademark images, which leads to the question of which features are most suitable. One answer is to consider how human perception and judgment utilizes those features.

2.3.1 Content-based image retrieval

Content-based image retrieval (CBIR) searches for desired images by their features which are automatically indexed (Eakins, 2001). CBIR eliminates the time needed to annotate keywords in text-based image retrieval, and improves system efficiency since it is not limited by the number of keywords.

Image representation and similarity measurement is the main focus of CBIR (Stanchev, 2001). A feature set is stored in a database, which represents the most important aspects of a set of images. Feature values are extracted from an image, and their similarity measured against the database set, as illustrated in Figure 2.3.
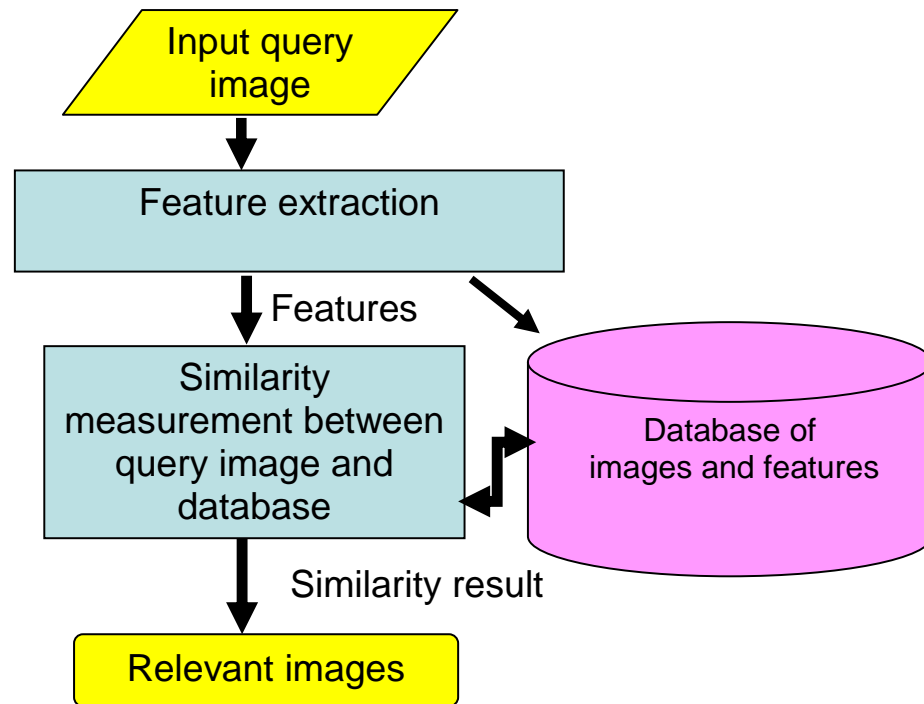
Figure 2.3 General content based image retrieval system

Trademark image retrieval systems are increasingly using CBIR (Mehrotra & Gary, 1995; Wu et al., 1996; Eakins et al., 1998; Jain & Vailaya, 1998; Alwis & Austin, 1999; Chan & King, 1999; Ravela & Manmatha, 1999; Safar et al., 1999; Shih & Chen, 2001; Yin & Yeh, 2002; Gori et al., 2003; Jiang et al., 2006). However, relevant trademark images cannot be the same or a modified image, which suggests that general CBIR may not be suitable for trademark image retrieval. The system requires additional techniques to extract the features and measure similar images.

2.3.2 Reported trademark image retrieval systems

Existing trademark image retrieval systems apply different approaches to extending general CBIR, highlighting the challenge of trademark image retrieval in the past decade (Mehrotra & Gary, 1995; Wu et al., 1996; Eakins et al., 1998; Jain & Vailaya, 1998; Alwis & Austin, 1999; Chan & King, 1999; Ravela & Manmatha, 1999; Safar et al., 1999; Shih & Chen, 2001; Yin & Yeh, 2002; Gori et al., 2003). Most systems use edge detection to perform segmentation and extract components from the image. This reflects a major condition for retrieving relevant trademarks − an ability to specify the real shape of components in an image.

2.3.2.1 STAR

Wu et al (1996) developed the STAR system (System for Trademark Archival and Registration) to retrieve similar trademarks for both word and device marks. The word mark similarity utilizes text, phonetics, and interpretation, while device mark similarity is divided into graphic meaning and graphic similarity. For the graphic meaning device mark, it has a specific meaning. They used the shape interpretation or the graphic meaning to measure similarity. The system requires that the user annotates the mark using Vienna classifications, and gives the similarity in terms of a fuzzy factor in a thesaurus relationship. For device mark, STAR compares its features to measure component similarity using the spatial relationship between components namely the structural description, and visual features. The system uses colour segmentation to separate the components in each mark, and a first order Markov field and Gaussian distributed clustering to specify the spatial relations. The system requires that the user assign the major and group components to be the structural description. The visual features are a Fourier descriptor, seven invariant moments, and projections. The similarity between features is calculated by a weighted distance of the spatial relationship, structural description, and the visual features. An experiment using 3000 trademark images showed that the system is effective for retrieving similar trademarks and reducing searching time, but many processes require user interaction.

2.3.2.2 Jain and Vailaya

Jain and Vailaya (1998) developed a system to retrieve similar trademark images by shape features, by utilizing fast pruning followed by refined matching. The pruning removes non-relevant trademarks by comparing edge direction histograms and seven invariant moments. The edge histogram is extracted from the boundary image, normalized, and smoothed. Seven invariant moments are extracted from the raw image. The query image features are compared with database features to calculate dissimilarity values, which are combined and normalized in the range 0 to 1. However, these two features were not accurate in the case of line drawing image rotation, so some similar images could be spuriously eliminated. The second stage is matching to a deformable template by edge mapping. The verified images from the first stage are compared with the

query image. The deviation and the energy function of the deformed template are used to calculate a similarity value. It requires the specific pose and deformation template parameters by iteratively searching a gradient descent in Hough transform space. The experiment was implemented using 1100 trademark images, and showed that the system could not always detect semantically similar images. Furthermore, the system has problems segmenting multi-component images, and the authors suggest that local features would extend the matching performance.

2.3.2.3 ARTISAN

Eakins et al (1997, 1998) developed the ARTISAN system (Automatic Retrieval of Trademark Images by Shape Analysis) to retrieve the similarity of abstract geometric shapes of device trademarks based on human image perception from Gestalt theory. The system extracts the features at three image levels: the individual boundary, the perceptual region, and the entire image. An individual boundary is a closed region in the image that consists of line and arc segments. A perceptual region is grouped by the co-linearism and co-curvilinearism of segments in the image. The system uses proximity, parallelism, and concentricity scores to group region boundaries into a family boundary, and tracks a family contour by utilizing an external family boundary. The system extracts shape feature vectors at all three levels, including aspect ratio, circularity, transparency, relative area, right-angleness, sharpness, complexity, directedness, and straightness. The experiment was implemented using 10745 abstract trademark images, and used normalized precision (P_n), normalized recall (R_n), and last-place ranking (L_n) to measure the system effectiveness. The retrieval performance is P_n =0.63 $\pm$ 0.24, R_n = 0.90 $\pm$ 0.12, and L_n =0.56 $\pm$ 0.31, which suggests that perceptual grouping has the potential to act like human similarity judgment. This system can also be extended to handle multiple component retrieval by local features.

2.3.2.4 Soffer and Samet

Soffer and Samet (1998) proposed negative shape features for classifying logos. Negative shapes are calculated by adding a border to the logo and extracting the internal holes for each component. Negative shape features consist of four global shapes (the first invariant moment, circularity, eccentricity, and rectangularity)

and three local shape descriptors (horizontal gaps per total area, vertical gaps per total area, and the ratio of hold area to total area). The similarity is measured by comparing the Euclidean distance of each component in the query image with database images, and the minimum distances of all the components of the query image are averaged to make a similarity score. They tested the retrieval performance with and without using negative shape features, and using one component by user selection and automatic multiple components. The experiment was implemented using 130 trademark images, and showed that using both positive and negative shape features of the multi-component logo gave the best performance in classifying logos involving triangular, long text, and stripes classes. However, no evidence was presented showing that the system can retrieve similar trademarks.

2.3.2.5 Alwis and Austin

Alwis and Austin (1999) proposed a trademark image retrieval system using combined multiple features from several image types: a boundary, a Gestalt, boundary closed figures, and a Gestalt closed figures image. The Gestalt images were obtained by grouping co-linear and co-curvilinear boundary segments. The system extracts perceptual features of both Gestalt and boundary images. The features are endpoint proximity, parallelism, co-linearism, and co-curvilinearism. In addition, the system calculates circularity, directionality, straightness, complexity, right-angleness, aspect ratio, sharpness, and the stuffedness of both boundary and Gestalt closed figures images. The boundary and Gestalt multiple features are compared by graphs, and the close figure features are measured for similarity by a distance bin. The experiment was implemented using 1000 trademark images, and the system effectiveness measured by precision and recall distributions over 10 queries from Artisan's system evaluation. The result shows that boundary-based features give a better score than Gestalt-based features. However, the combined feature using Dempster-Shafer method gives the best score. This system also has the potential to be handle multi-component retrieval by local features.

2.3.2.6 Ravela and Mammatha

Ravela and Mammatha (1999) proposed a multi-modal system for the retrieval of

similar design trademark images. The system begins with user input text and uses a text search engine to retrieve matching trademark images. The images are matched by visual appearance, or a composition of text and visual appearance. The selected images are filtered with multi-scale Gaussian derivatives to describe the structure of their intensity surface, and the system extracts local curvature and local phase histograms. These local features are used to obtain global similarities by comparing their histogram vectors with normalized cross-covariances. The experiments used 2048 binary images and ranked 48 similar images; given an average precision of 61.1%. The system used whole image features to perform global matching, so there is evidence that it can be degraded by occlusion and missing parts; and human perceptual factors were not considered.

2.3.2.7 Chan and King

Chan and King (1999) proposed the genetic weighting of several features for the retrieval of similar trademark images, including Fourier descriptors, seven invariant moments, eccentricity, circularity, and Euler number. They use a closing operator to group connected components in the image, which may lead to wrong grouping, because it does not consider perceptual factors. They calculated the integrated dissimilarity value by weighting a combination of Euclidean feature distances. A genetic supervised learning algorithm was applied to similar trademark images, with iterative optimisation, to calculate the weighting values. The experiment used 1360 binary trademark images, and among 20 top rank images, the method retrieved all the similar images. However, there was no mention of the number of similar images.

2.3.2.8 Shih and Chen

Shih and Chen (2001) proposed a system employing semi-automatic trademark segmentation, image features, and user weighting feedback. The system selects each object region in a binary image by user intervention and the regions are extracted by the region-growing algorithm and line-connecting segmentation. The system extracts Hu seven invariant moments, the Fourier transforms of edges from polar-coordinate transforms, the first derivatives of edges from polar-coordinate transforms, and a histogram of edge directions. The system measures trademark similarity by feature distances, which are weighted by user feedback.

The experiment used 3543 device trademark images, and a combination of the four features achieves the best result when compared with each independent feature. However, the system requires user segmentation to perform human visual perception.

2.3.2.9 Yin and Yeh

Yin and Yeh (2002) proposed an automatic content-based trademark retrieval method, which extracts seven features from a trademark binary image: area, isolation, deviation, symmetry, centralization, complexity, and two-level contour representation strings. The system removes redundant feature values by correlation and entropy thresholds, and classifies all trademarks into several classes with a fuzzy c-mean algorithm. The candidate classes are matched to the query image by comparing the normalized distance, and each distance weighted by user feedback. The experiment used 1000 trademark images, and showed that user feedback improves the retrieval efficiency. The system requires user judgment, so it has the potential of using human visual perception to improve the system efficiency.

2.3.2.10 Gori et al

Gori et al (2003) proposed an edge-back propagation method to recognize logos under Baird and spot noise conditions, by adapting a back propagation neural network. The input of the neural network is a fixed size vector of averaging grey levels of pixel regions, which the regions being connected components segmented by morphological transforms. It requires a training stage and iteration processes, with prepared noisy images. The experiment was implemented on a database of 88 logos with added noise, with the results showing that the methods can recognition images with spot noise that do not have large occlusions. However, this method does not provide for similar image retrieval.

2.3.2.11 Jiang et al

Jiang et al (2006) proposed the adaptive selection of visual features from five types of Gestalt principles: symmetry, continuity, proximity, parallism, and closure. They employ Hough transforms (Ballard, 1987) to detect line, circle, and arc segments in a boundary image, from which parallel lines, concentric circles,

and arcs are calculated. They claim that Hough transforms automatically utilize Gestalt principles such as continuity, proximity, and parallelism, and can detect occluded and confused components. The system also detects polygons using end-to-end distances to group near neighbour line segments, but with the polygon limited to triangles, squares, and rectangles. Hough transforms require huge computation time, so they limit the input image resolution to 100 x 100 pixels. The system generates Zernike moments for integration with previous features by thresholding on saliency degree, and relevant images are extracted by maximumWBG matching. The system filters irrelevant features before matching because all the features, except Zernike moment, are not transform invariant. The system was evaluated with trademark images in a MPEG-7 dataset with 50 queries, with the performance measured by normalized precision (P_n), normalized recall (R_n), and last-place ranking (L_n). The retrieval performance is P_n =0.66 $\pm$ 0.18, R_n = 0.87 $\pm$ 0.11, and L_n =0.61 $\pm$ 0.28, which suggests that perceptual grouping has the potential to match a human similarity judgment. This system also has the potential to be extended to multiple component retrieval by local features.

2.3.2.12 Conclusion of reported trademark retrieval systems

A comparison of the discussed trademark image retrieval systems appears in Table 2.2.

Table 2.2 Properties of the trademark image retrieval systems

System	Required Segmentation	Provide Perceptual grouping	Provide Shape Similarity	Used Local features	User feedback	Performance
STAR	Yes	No	Yes	Partial	Yes	Good
Jain and Vailaya	Yes	No	Yes	No	No	Bad
ARTISAN	Yes	Yes	Yes	Partial	No	Good
Soffer and Samet	Yes	No	N/A	Partial	No	N/A
Alwis and Austin	Yes	Yes	Yes	Partial	No	Good
Ravela and Mammatha	No	No	Yes	Partial	Yes	Good
Chan and King	Yes	No	Yes	No	No	N/A
Shih and Chen	Yes	No	Yes	No	Yes	Fair
Yin and Yeh	Yes	No	Yes	No	Yes	Good
Gori et al	Yes	No	No	No	No	N/A
Jiang et al	No	Yes	Yes	Partial	No	Good

Most of the systems require image segmentation except Ravela and Mammatha's work. Only ARTISAN, Alwis and Austin, and Jiang et al. provide perceptual grouping. There are two systems by Soffer & Samet and Gori et al. that do not provide shape similarity. All the systems use global features, and some employ local features. The systems that provide user feedback perform well because the feedback reflects user judgement.

Many proposals capture different aspects of an image's appearance, such as texture, colour, structure, and shape (Rui et al., 1997; Bhattacharjee & Ebrhimi, 1999; Eakins, 2001; Eakins et al., 2001). An important aspect of image appearance is shape (Scassellati et al., 1994; Mehrotra & Gary, 1995; Safar et al., 1999; Eakins, 2001). In the real world, many applications rely on shape, for instance medical diagnosis, law enforcement, and trademark registration. Moreover, colour and texture do not have enough discriminating power to retrieve some types of abstract or grey scale images (Jain & Vailaya, 1996; Geradts et al., 2001). For this reason, using shape features in trademark image retrieval should be investigated in more detail.

2.4 Shape representation

An abstract trademark is a multi-component image that contains graphical designs that relate to shape retrieval. Generally, a shape is the representation of an object from the external form or appearance of a pattern. Shapes can be simple such as rectangles, circles, ellipses, triangles, or polygons, or more complex such as closed curves, or contours.

Human beings perceive shape by utilizing visual pathways from the eye to the brain (Levine, 1985). The pathways begin when the retina receives a pattern of light and its information is sent to the brain by the optic nerve (Bruce, 1996), where the perception process is initiated. To explore a shape by computer, we analyse its contents and convert this to digital features. In CBIR, a shape can comprise a configuration of binarised objects (depicted as either black or white) and be represented by 2D binary shape features.

Shape features are the descriptors that portray the appearance of an object in an image, and form the numerical data in the analysis process. Shape features should have good discrimination power, be reliable, independent, and compact (Ming, 1999).

Shape features can be extracted by coding or transform techniques (Marshall, 1989), and have many possible representations, including chain codes, edge direction histograms, scale space histograms, chord distributions, moments, scalar quantities values, line-sums/projections values, stochastic values, contour distributions, and transform values (Marshall, 1989).

Shape features can be categorized in a number of ways. Some features can be used to restore an original shape, and are therefore called information preserving. Examples include contours and Fourier descriptors. Other features have insufficient information to restore the original and so are not information preserving, such as perimeters, areas, and aspect ratios.

We can also categories shape extraction techniques into two types: boundary-based and region-based (Safar et al., 1999). Boundary-based techniques need an edge detector to transform the image into a shape boundary, and use only the contour or boundary of a shape to calculate the shape feature. However,

contour images require connected boundaries obtained by a segmentation process. On the contrary, region-based shape features are calculated from an entire image region.

Shape features can also be categorised into global and local features. A global feature is calculated using a whole image, and a local feature from some part of it. Many researchers have used global features, because of their high discrimination power. However, they cannot be used in the presence of occlusions, or for joined objects (Mehrotra & Gary, 1995). Local features are computed from local shape regions, and can potentially deal with occlusions. However, local features are sensitive to noise and rotation and by definition can comprise many thousands of instances in a single image. Therefore, pre-processing is required to handle noise reduction, point detection, and edge detection and the sheer numbers of local features detected. Thus, local features are more computationally expensive than global features.

An important requirement for shape similarity in image retrieval is invariance. Shape similarity should be invariant to translation, scaling, and rotation (Loncaric, 1998; Geradts, 2002).

2.5 Global features

Many researchers use colour, texture, and shape to retrieve images (Jain and Vailaya, 1995; Rui and Huang, 1999; Huang and Chang, 1997; Datta et al, 2008).

2.5.1 Colour features

Colour features are used widely, and include MPEG-7 colour descriptors (Manjunath et al, 2001), CIELAB colour descriptors (Othman and Martinez, 2008), and colour histograms (Jain and Vailaya, 1996). Colour features may also be useful at the semantic level in colour image retrieval (Stanchev, 2003).

However, colour features are not relevant to our work because trademark images are predominantly reproduced as gray tone images.

2.5.2 Texture features

Textures are measured by many features, including MPEG-7 texture descriptors

(Manjunath et al, 2001), Gabor-based features (Simona et al, 2002), moment-based features (Robert et al, 1979), contrast (Robert et al, 1979), correlation (Robert et al, 1979), and entropy-based features (Robert et al, 1979).

2.5.3 Shape features

We divide this approach into global shape feature and local shape features.

Global shape features are calculated from the entire image object. Some global shape features are listed below.

Aspect ratio

$$y_{max} / x_{max} \qquad\qquad \text{(Equation \quad 2.1)}$$

where y_{max} and x_{max} are the maximum length of the shape along the y and x coordinates.

Fourier descriptors

$$\theta(t) = \mu_0 + \sum A_k \cos(kt - a_k) \qquad\qquad \text{(Equation 2.2)}$$

where A_k and a_k are the kth harmonic amplitudes and phase angles respectively.

Zernike moments

$$A_{nm} = \frac{n+1}{\pi} \sum_\rho \sum_\theta (R_{nm}(r) e^{im\theta}) * I(r, \theta) \, | r < 1 \qquad\qquad \text{(Equation 2.3)}$$

where n is the order of Zernike moment with m repetition for $I(r, \theta)$.

$R_{nm}(r)$ is the set of radial polynomials defined by Zernike.

Roundness or compactness

$$\gamma = (\text{perimeter})^2 / 4\pi \, (\text{area}) \qquad\qquad \text{(Equation 2.4)}$$

Invariant moments

The two-dimensional moments of order (p, q) of image (I) are defined by:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q I(x, y) \qquad \text{(Equation 2.5)}$$

The normalized central moments are defined as:

$$\eta_{pq} = \frac{\mu_{pq}}{(\mu_{00})^w} \;\; ;p,q=0,1,2,... \qquad \text{(Equation 2.6)}$$

where

$$w = \frac{1}{2}(p + q) + 1 \; ; p+q=2,3,... \qquad \text{(Equation 2.7)}$$

Seven invariant moments can be derived from the second and third order moments (Mehtre et al., 1997):

$$m_1 = \eta_{20} + \eta_{02} \qquad \text{(Equation 2.8)}$$

$$m_2 = (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2 \qquad \text{(Equation 2.9)}$$

$$m_3 = (\eta_{30} + \eta_{02})^2 + (3\eta_{21} + \eta_{03})^2 \qquad \text{(Equation 2.10)}$$

$$m_4 = (\eta_{30} + \eta_{12})^2 + (3\eta_{21} + \eta_{03})^2 \qquad \text{(Equation 2.11)}$$

$$m_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$
$$\text{(Equation 2.12)}$$

$$m_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$

$$\text{(Equation 2.13)}$$

$$m_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{30})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$
$$\text{(Equation 2.14)}$$

Local shape features can be calculated from local regions, such as corner points, boundary segments, curvature, and turning angles (Wang, 1999; Stanchev, 2001).

Global features are generally calculated from the whole image without utilizing multi-component contours, and therefore do not represent the underlying shape (Wang, 1999). Furthermore, global features do not remain consistent when

the image content is affected by occlusion or contains connected components that should also be compared in isolation (Mikolajczyk, 2002).Thus, global features cannot support image sub-part retrieval and can be confounded by occlusions. For example, the image in Figure 2.4 contains a triangle and a rectangle, but the global feature is a single polygon. In addition, many abstract trademarks are multi-component images, as in Figure 2.5. If any component is modified or removed, the global feature description will changed and so is not robust.
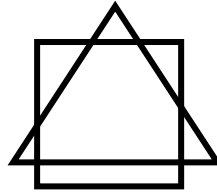
Figure 2.4 Occluded shapes
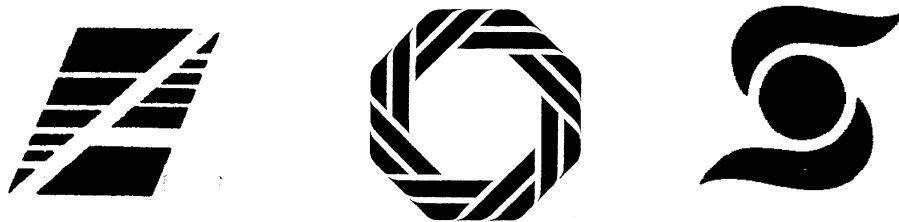
Figure 2.5 Multiple components in abstract trademarks

Experiments by Eakins show that component-based matching can be more effective for retrieving similar trademarks than whole-image matching (Eakins et al., 2001). Eakins (2003) states that using shape elements offers more discrimination power than a whole shape boundary.

2.6 Local features

Local features can be used to find part of an image and so are more suitable for multi-component image retrieval than global features since they have the advantage being able to match whole images or images parts, and thereby support multi-component image matching.

There are three aspects of retrieval by local features: feature location, desirable local shape feature properties, and the matching method.

We tend to focus on the salient local features in an image, so using all the pixels in an image to compute a feature is unnecessary. Proposed methods for manipulating the location include corner detectors, key point detectors, salient point detectors, and interest point detectors. Corner detectors label corner features while the other types of detector label salient locations in the image, such as curvature inflections, curvature maxima, and points resulting from image transforms (Schmid & Mohr, 1997; Loupias & Sebe, 1999).

Good local feature properties that reflect image similarity are required. For instance, local features representing shape should be invariant to translation, rotation, and scaling (Rui et al., 1997).

Local features can be used in part-image matching or in whole image matching. Matching results require a similarity ranking of database images compared to the query image.

We would like to use structural and appearance features for each interest point. The structural features represent the shape structure by quantifying the spatial configuration of interest points that a shape comprises. For example, two-largest-angle features of a Delaunay triangulation of interest points has been proposed for shape representation (Tao & Grosky, 1998); since it is highly efficient at distinguishing objects by means of feature point relationships. A Delaunay triangulation of interest points is created, and each point's angle is measured, so the two largest angles in the same triangle can be selected. Also, the local direction of SIFT (Scale Invariant Feature Transform), (Lowe, 2004) features can be used to contribute to the specification of the structure of a shape.

The local appearance of features is used to represent local properties of the interest points. Lowe (1999) developed SIFT features for object recognition, which are robust under partial occlusion in cluttered images and he reported more details of the SIFT approach in his later paper (Lowe, 2004). He claims these features are invariant to image scaling, translation, and rotation, when using invariant key points. The local direction of the SIFT features are the image gradient magnitude ($\| \nabla I(x,y) \|$) and orientation ($\theta(x,y)$) of Gaussian smoothing image:

$$\| \nabla I(x, y) \| = \sqrt{(I(x, y) - I(x+1, y))^2 + (I(x, y) - I(x, y+1))^2} \quad \text{(Equation 2.15)}$$

$$\theta(x, y) = \tan^{-1} \frac{I(x, y) - I(x+1, y)}{I(x, y+1) - I(x, y)} \quad \text{(Equation 2.16)}$$

The magnitude and orientation are computed at each key point. SIFT features also provide an appearance descriptor that consists of sixteen concatenated 8 element edge orientation histograms, (each element corresponding to one of 8 orientation directions) extracted from a 4x4 array of grids centred on the key point.

Schmid and Mohr (2003) have evaluated the performance of a variety of local descriptors. SIFT descriptors were the best, followed by steerable filters. However, steerable filters have the advantage of low dimensionality. They are calculated with Gaussian derivatives, and apply Gaussian kernels with σ=7 in an image patch of size 45. Each image patch performs a convolution with five different fourth Gaussian derivatives (Freeman & Adelson, 1991; Mikolajczyk & Schmid, 2003).

$$G_4 a = 1.246(0.75 - 3x^2 + x^4)e^{-(x^2+y^2)} \quad \text{(Equation 2.17)}$$

$$G_4 b = 1.246(-1.5x + x^3)(y)e^{-(x^2+y^2)} \quad \text{(Equation 2.18)}$$

$$G_4 c = 1.246(x^2 - 0.5)(y^2 - 0.5)e^{-(x^2+y^2)} \quad \text{(Equation 2.19)}$$

$$G_4 d = 1.246(-1.5y + y^3)(x)e^{-(x^2+y^2)} \quad \text{(Equation 2.20)}$$

$$G_4 e = 1.246(0.75 - 3y^2 + y^4)e^{-(x^2+y^2)} \quad \text{(Equation 2.21)}$$

Due to their desirable performance characteristics SIFT descriptors, and other related local features, will be investigated to identify suitable features for distinguishing shapes. Shape similarity judgment is an important aspect in abstract trademark image retrieval and is explained in next section.

2.7 Shape similarity judgment

Patent office examiners consider trademark images by sight, and so human image perception and similarity judgement play an important role in trademark identification. However, few researchers find image features by utilizing principles based on human visual perception.

Humans compare images using rules to evaluate their similarity, as in Gestalt theory (Wertheimer, 1923). Psychologists proposed the Gestalt laws for grouping perceptual organization in images based on proximity, similarity, continuity, co-linearity, co-curvilinearity, closure, parallelism, symmetry, and familiarity.

Murray et al (2002) states that the structure of the primary visual cortex (V1) reflect greatly an underlying use of local image features, and higher visual areas, including the lateral occipital complex (LOC), appear to group local image features into coherent objects. This implies that some local aspects of an image could potentially have a large impact on human shape similarity judgments.

A number of researchers have reported that Inferior Temporal (IT) neurons are involved in shape recognition (Schwartz et al, 1983; Logothetis and Sheinberg, 1996; Tanaka, 1996). Vogels et al (2001) report that IT neurons respond to non-accidental properties during shape similarity testing.

Biederman claims that humans perceive the non-accidental properties of an image, using them to fill and identify an object in memory (Biederman, 1987). The properties are smooth continuation, co-termination, parallelism, and symmetry, which support Gestalt laws. Figure 2.6 shows examples of recoverable images derived from the non-accidental properties. Biederman (1995) reported that a class of objects comprising distinct visual elements could be represented by "shape primitives", as opposed to amorphous objects which are not amenable to such decomposition. Trademarks are an exemplary instance of objects that can be decomposed into primitives, hence Biederman's recognition by components theory is highly appropriate here.

(a1)   (b1)   (c1)

(a2)   (b2)   (c2)
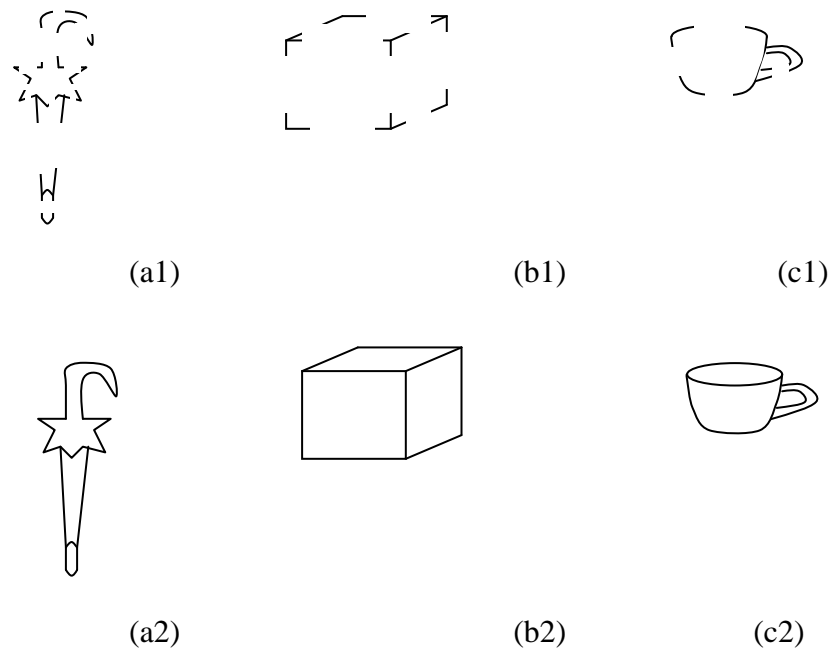
Figure 2.6 Recoverable images derived from the non-accidental properties in a1-c1 and the original images in a2-c2

2.8 Grouping and non-accidental properties by computer

Gestalt laws group a number of (typically small) image structures into perceptually significant larger components, which we can use to measure image similarity.

Gosselin and Schyns (2001) developed Bubbles, a technique for measuring human categorization performance for specific visual information. Bubbles are generated from holes punctured in an observed image by Gaussian windows. Gibson et al (2007) have shown that the human visual system is biased to recognizing objects from non-accidental properties by using Bubbles. This result is also supported by Biederman (2007), who observed that non-accidental image feature properties are vital for recognizing objects.

We believe that by applying perceptual grouping and non-accidental properties of local features, we can model human-like perception for retrieving images by computer.

The Gestalt laws show that shape is very important in human visual

similarity judgment. Biederman (1987) has shown that humans can recognize shape by distinctive elements. The success of human visual perception encourages imitation for image similarity by computer. Non-accidental properties are potentially of use in human shape similarity judgement, and this idea provides the motivation for interest point extraction.

2.9 Interest point detectors

Interest point features supports the machine implementation of visual perception concepts derived from human vision, including multiple-component matching, and interest points also have high information content, and are robust to partial visibility (Han & Guo, 2002). In mammalian vision systems, the broadly equivalent functionality is manifest within end-stopped cells which are believed to represent shape, and also maximally respond to corners or vertices (Biederman, 1995).

Interest points are required to specify the positions of local features, and so have been used by many researchers for object recognition for many years (Schmid & Mohr, 1997; Loupias & Sebe, 1999; Lowe, 1999; Jugessur & Dudek, 2000; Sebe, 2001). Interest point detectors are techniques to find the location of the most important image points that are also interest points (Schmid & Mohr, 1997; Sebe, 2001), which allows these to be used to compare the similarity of images (Schmid et al., 2000). Interest points can be corners, junctions, signal changing points, maxima curvature points, and points resulting from transforms (Schmid & Mohr, 1997; Smith & Brady, 1997; Loupias & Sebe, 1999; Schmid et al., 2000; Sebe, 2001; Sebe & Lew, 2003).

We divide interest point detectors into two categories in terms of their input. Intensity-based methods use an image directly and calculate interest points using every pixel an image. Boundary-based methods use shape boundaries of input images and calculate interest points using only a fraction of the pixels in an image, such as its silhouette. Intensity-based methods do not have the problems of segmentation or edge detection, and can extract occluded or connected objects. On the other hand, boundary-based methods have the potential to reflect the real shape of an object more accurately than intensity-based methods. In this research,

we focus on intensity-based methods because they do not require object boundary contours to be segmented and can be calculated from a raw image. Therefore, only interest point detectors based on intensity-based methods are investigated in more details.

Todman and Claridge (2000) state that low-level features, such as junctions, are important for perceptual grouping. Since perceptual grouping is required in trademark image retrieval, perceptual grouping is investigated in next section.

2.10 Perceptual grouping

The Gestalt laws of organization attempt to codify perceptual grouping mechanisms in human vision Many researchers have applied Gestalt laws to computer vision (Lowe, 1985; Mohan & Nevatia, 1992; Kang & Walker, 1994; Sarkar & Boyer, 1994; Havaldar et al., 1996; Boyer & Sarkar, 2000; Rome, 2001; Kruger & Worgotter, 2002).

Eight perceptual grouping laws are widely used, as below:

- Similarity: groups image parts that have similar local features. We may use two features to measure this property (Jacobs, 2000).

- Proximity: groups image parts that have similar local features, which are close to each other. We may use two points to measure this property (Jacobs, 2000).

- Continuity: groups image parts that construct good continuations. We may use curve fitting to measure this property (Jacobs, 2000).

- Co-linearity: groups image parts that approximately lie on the same line. We may use angle and perpendicular distance between two lines, or local intensity gradients, to measure this property (Walker & Kang, 1994).

- Co-curvilinearity: groups image parts to produce longer curves. We may use curve fitting to measure this property (Kimia et al., 2000).

- Closure: groups image parts that produce closed curves. We may use curve fitting to measure this property (Kimia et al., 2000).

- Symmetry: groups image parts to produce parallel or super segments. We may use curve matching to measure this property (Kimia et al., 2000).

- Parallelism: groups two parallel line segments. We may use the angle between two lines to measure this property.

The fundamental laws that determine perceptual grouping are proximity, similarity, closure, and simplicity (Wertheimer, 1923). Simplicity tends to organize components into simple components according to symmetry, regularity, and smoothness (Wertheimer, 1923).

Ben Av and Sagi (1995) show that similarity and proximity properties can be measured based on intensity autocorrelations of element features. For that reason, SIFT features and related local features could be used to measure similarity and proximity properties. Structural features, such as moments, can also be used to measure the simplicity property. We propose to use local features to achieve perceptual grouping according to similarity, proximity, and simplicity laws.

Because local feature groupings can potentially indicate either the local or global characteristics of images (Lowe, 2001), the results of grouping will be used to measure image similarity.

Using the similarity law, we can see that component shape similarity results in the percept of horizontal rows in Figure 2.7 and Figure 2.7 (a) is more similar to Figrue 2.7 (b) than Figure 2.7 (c), in this case based on the degree of *local* shape similarity since the global spatial configuration of the feature locations in (a), (b) & (c) below is identical.
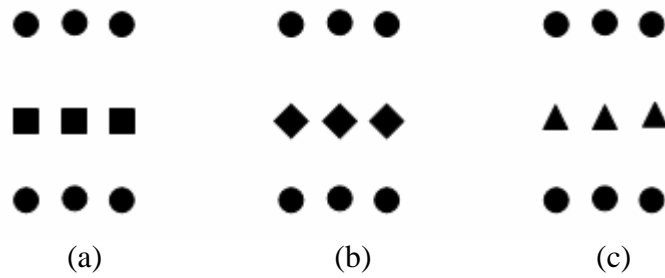
Figure 2.7 Similarity groupings suggest horizontal rows

Using the proximity law, closer components can be perceived as groups forming vertical columns in Figure 2.8. Figure 2,8 (a) is more similar to Figure 2.8 (b) than Figure 2.8 (c), based on the structure of the *global* configuration of the identical local elements in each example.
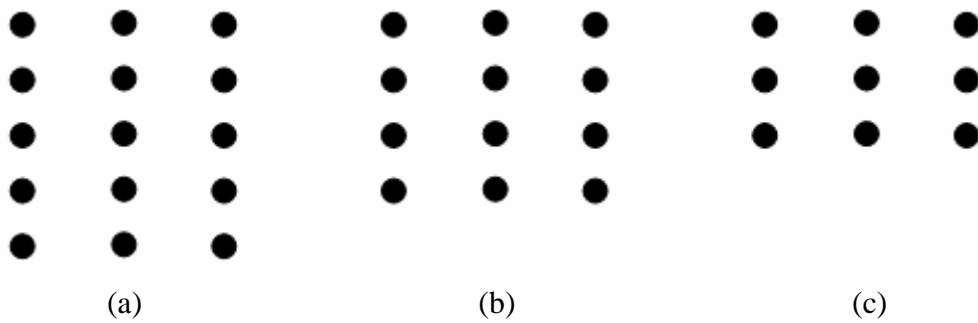
Figure 2.8 Proximity groupings suggest vertical columns

Using the simplicity law, the area which is enclosed by a symmetrical shape is perceived to be consistent in form. Figure 2.9 (a) is perceived to be more similar to Figure 2.9 (b) than to Figure 2.9 (c) based on the degree of symmetry of each element.
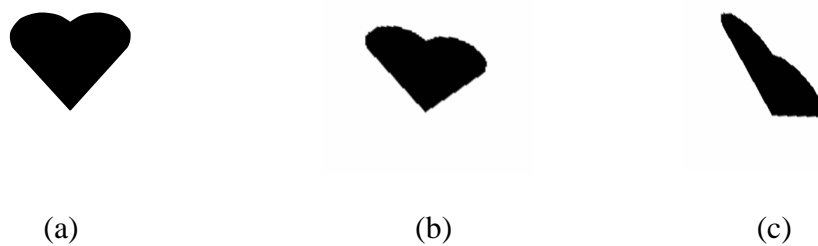
Figure 2.9 Symmetry grouping suggest (a) and (b) are more similar than (a) and (c)

However, it must also be noted that the shape similarity law could also be invoked here, as Figure 2.9 (b) could be considered to be a less distorted version of Figure 2.9 (a) than Figure 2.9 (c) without taking symmetry into consideration. Therefore, the perceptual influence of the different grouping laws cannot always be untangled, and may potentially operate simultaneously.

2.11 Discussion and conclusions

The literature shows that there are gaps in the reported trademark image retrieval research. There are few useful techniques for partial shape matching in the context of trademark retrieval, because those existing techniques tend not to support multi-component retrieval. Many techniques need image segmentation, which is an unsolved problem. Also, there are few researchers who use human perception for trademark image retrieval, which means that the shape representation does not reflect the shape as characterised by human perception. Finally, global features are not suitable for retrieving occluded or connected component images.

Human shape perception offers many advantages for trademark image retrieval. However, there are the problems of occluded and connected components, segmentation, and multi-component matching. Trademark image retrieval must support partial shape matching and multi-component retrieval, and can employ non-accidental properties of local features to avoid tackling segmentation issues.

We believe that using interest points in trademark image retrieval can improve the performance of systems for abstract trademark image retrieval. Interest points have been reported in the literature to have been used successfully in image retrieval (Schmid and Mohr,1997; Lowe,1999; Wolf,2000; Sebe,2001). Also, interest points are well suited to supporting partial matching and local feature extraction because they can be tolerant to the effects of local occlusions. Therefore, we propose to group the locations of detected local features loosely based on a subset of the Gestalt laws and potentially in a manner that exploits the non-accidental properties of these features. For example, by applying a transform space, or a robust affine estimation process, between the local features extracted

from compared images, we can characterise the goodness of fit of matching features. We can then evaluate the similarity of these features in higher semantic terms by evaluating the spatial configurations of matching feature constellations using standard techniques such as statistical moments. Therefore, we propose to compare images both in terms of the similarity of extracted local features and also in terms of different measures of the similarity of the spatial configurations of these local features. In the next chapter, we propose an approach for retrieving trademark images based on these ideas.

Chapter 3

A Principled Approach to Trademark Image Retrieval

This chapter provides an overview of a new approach to trademark image retrieval that tackles the complicated problem of dealing with similar images. The main goals of this research are examined, providing motivation for our system framework. We give an overview of the approach and of the system framework.

3.1 Summary of our Goals

Trademark retrieval systems typically utilize text-based retrieval, with keywords obtained from components of the trademark being investigated. The trademark's figurative elements are annotated using the Vienna classification to describe trademark images. However, text-based systems are time-consuming and (very labour) intensive. Furthermore, the use of annotations is unsuitable for abstract trademark image retrieval.

One of our aims is to develop new techniques for matching trademark image elements, which require new methods for solving partial matching and shape perception problems. The research literature highlights several issues with current types of matching processes and shape retrieval.

1.   There are few techniques aimed at partial shape matching, which is necessary for supporting multi-component retrieval.

2.   Many approaches require exact image segmentation, which is still an unsolved problem in the general case. Consequently, these techniques may not extract appropriate component shapes from an image.

3.   There is little work that reports employing human visual perception and shape similarity judgment for shape retrieval, which means that the chosen

shape representation may not reflect the real shape of the trademark.

4.   Global features are not suitable for retrieving occluded or connected component images.

The second and fourth issues can be addressed by *local features*, while the first and third issues can be dealt with by *interest points*.

In summary, our goals are:

- We will utilize certain principles employed in human visual perception in our system. The Gestalt laws of organization show that shape is very important in human visual similarity judgment, and have been applied to trademark image retrieval (Eakins, Boardman et al. 1998; Alwis and Austin 1999). However, additional segmentation and clustering processes are required to group image elements, so Gestalt grouping principles can be applied to obtain meaningful components. In addition, retrieval failure is a problem due to segmentation and clustering inadequacies. Biederman has shown that humans recognize shapes using distinctive elements (Biederman, 1987). Since trademark images contain multiple graphical shapes, shape similarity judgement has a vital role to play in trademark image retrieval. Non-accidental properties greatly assist human shape similarity judgement, and this observation provides the motivation for the inclusion of interest point extraction and local features in our system.

- Our system will support multi-component image retrieval. This will allow local features to be used to find image elements, and judge a shape more effectively. Local features are more suitable for multi-component image retrieval than global features, since they can be employed in both part and whole image matching, and support multi-component image matching.

We describe our proposed approach in more detail in the next section.

3.2 Overview of Our Approach

Our approach deals with the similarity problem between trademark images. Our

framework utilizes interest point extraction, local feature calculations, and decision making strategies. These capabilities are discussed briefly below, and we provide a roadmap to their detailed treatment in subsequent chapters.

Trademark registration is time consuming and labour intensive, so more effective, and automatic, trademark image retrieval would be beneficial. An important drawback of current systems is that abstract trademark images are difficult to describe by keywords, making content-based image retrieval an attractive solution.

Abstract trademark images are usually multi-component images, which present many problems for shape retrieval. Previous studies show that interest points provide useful information for image retrieval (Schmid et al., 2000; Harris, 1988; Chabat et al., 1999; Smith and Brady, 1997; Sebe and Lew 2003). They support partial matching and local features, thereby avoiding the occlusion problem. They may also reduce the amount of matched data.

Our system will utilize the following elements to support the efficient retrieval of trademark images:

- Interest points, to provide useful information, that can be used to specify the positions and provide the spatial configuration of extracted local featuers, in a form suitable for abstract trademark image retrieval.

- Shape descriptors, based on interest points and local features, to offer capabilities similar to a subset of those which are predominant in human visual perception. Shape descriptors also provide useful information for measuring shape similarity.

The main questions that need to be addressed are:

- Which interest point techniques can most effectively deal with distorted trademark images (e.g. with noise, rotation, translation, and scaling)? Interest point detectors can be divided into two main types. Intensity-based methods obtain interest points by pixels calculations on the entire image. Boundary-based methods obtain interest points by pixel calculated on

certain parts of an image, such as on silhouettes or contours. Intensity-based methods do not have segmentation problems or edge detection issues, and can extract occluded or connected objects. We investigate these questions in chapter 4.

- How can interest point techniques be applied to the domain of abstract trademark image retrieval? Interest points can greatly aid the creation of local features because they provide information content suitable for measuring image similarity (Schmid, Mohr et al. 2000; Han and Guo 2002). We look at applying interest points and local features to abstract trademark image retrieval in chapters 4, 5, and 6.

- How can perceptual grouping be carried out automatically? We believe a solution lies with shape descriptors that consist of local appearance features and the spatial locations of local features based on Gestalt principles to reflect certain aspects of human visual perception. Gestalt principles have been previously applied to extracting global features based on contours (Eakins, Boardman et al. 1998; Alwis and Austin 1999). Contour features require segmentation, and the major drawback is incorrect clustering (Eakins, Boardman et al. 1998). Hence, we propose to represent the relationship of image components by employing transformed shape descriptors (i.e. appearance and structural features) globally based on Gestalt principles. Further details can be found in chapter 6.

- How can shape descriptors be generated which distinguish between the component shapes in an abstract trademark image? Local features are more suitable than global features for multi-component image retrieval because of their robustness if any image components are lost. We utilize a vector of appearance and structural features as a shape descriptor (Datta et al., 2003). A vector can be used for training and determining shape similarity that can potentially reflect aspects of human visual perception. We evaluate this approach in chapters 6, 7, and 8.

3.3 System framework

We are motivated by the idea of grouping those parts of an image which contain non-accidental properties in order to recognise shapes. Recent research results show that non-accidental properties can potentially be used in object recognition (Biederman, 1987; Lowe, 1985; Draper et al., 2003). Non-accidental properties are also potentially implicated used in human shape similarity judgement.

Interest points are suitable for extracting important areas in image that result from the non-accidental properties of their parts. In addition, interest points can provide high information content and are inherently robust to partial occlusions (Schmid and Mohr, 1997). We apply an interest point detector to an image in order to extract interest points.

Local features have the potential to allow the system to achieve partial matching and they support multiple components matching (Lowe, 1999). We use interest points to specify the positions of local features. SIFT features are selected to be the primary features adopted by the system because they are robust and exhibit high discrimination power (Lowe, 1999; Schmid and Mohr, 2003). The best candidate match to each interest point is specified by locating smallest Euclidean distance between each feature extracted from the input image and each feature to which this is compared in the database of interest points. (Lowe, 2004). This process can eliminate insignificant points (Lowe, 2004).

We can measure perceptual grouping from local features supporting Gestalt laws and reflecting shapes (Liu et al., 2007). Appearance-based approaches have been used in object recognition (Hornegger et al., 2000). The appearance features can solve many problems such as obviating the need to segment image parts and can also facilitate geometric modeling of complex objects (Hornegger et al., 2000).

The system uses meta-feature vectors (also called shape descriptors, which contain structural features and appearance features) for training and determining similar trademarks. The meta-feature vector support human-like perception of shape, using appearance similarities and structural similarities (based on the

spatial configuration of local feature locations) and points matched using specific properties such as orientation and proximity. The global characteristics of an image such as global size and global orientation are determined using statistical measurements that describe the spatial distributions of the interest point locations (details in chapter 6). These global image characteristics are computed using local features to group similar parts of an image for example Figure 3.1 shows different global orientations of six hearts to which the same degree of rotation as been applied.

<div align="center">(a)   (b)</div>

Figure 3.1 images of six hearts (a) in global 0 degree and (b) in global 20 degrees

The Hough transform (HT) can also be applied to perform perceptual grouping. The Hough Transform is employed to support Gestalt principles such as continuity, proximity, and parallelism (Jiang et al., 2006). Draper et al. (2003) suggests that the Hough transform is suitable for representing appearance-based recognition. The Hough transform can also retrieve different shapes (Ballard, 1981). In addition it can be applied to measure non-accidental properties such as parallelism, and symmetry (Draper et al., 2003).

The interest points groups from Generalized Hough Transform (GHT) (Ballard, 1987) are successfully used in object recognition, panorama stitching, and 3D matching (Brown and Lowe, 2002). In this case Lowe applies the GHT to SIFT keypoints that match between compared images in order to determine the relative scale, rotation and offset between these images.

The GHT is in essence a voting space that records the probability density of the occurrence of matching local features between compared images. Where matching features have similar properties, clusters appear in Hough space.

Accordingly, by detecting these clusters we have a mechanism for binding the features associated with each cluster into a group that exhibits some common property, such as dominant orientation, spatial scale. In turn this mechanism provides a means of associating similar groups between compared images, and hence a means of comparing image similarity based on local property grouping.

Our proposed system uses the GHT to cluster matched interest points: for example, the percentage of matched point and total interest points could be used to measure overlap proximity (details in chapter 6). Figure 3.2 (b) shows overlap of three hearts from Figure 3.2 (a). Further processes based on the GHT can also be applied to measure other perceptual groupings in our system, such as the standard deviation of the difference between pairs of match points after an affine pose estimation process (details in chapter 6).

(a)   (b)
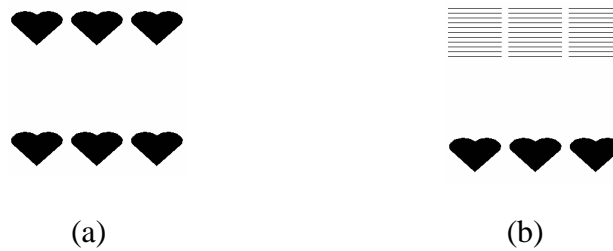
Figure 3.2 images of six hearts (a) with no overlap and (b) with 3 overlapped hearts

The system determines the similarity of images by employing meta-feature vectors and support three visual perception properties that are similar to Gestalt grouping based on proximity, similarity, and simplicity. These structural grouping features allow global comparisons to be made based on diagnostic summaries of local feature groupings captured over the whole image. The above is much less general than human vision, which can perform the same grouping tasks in a local and hierarchical basis, and also make comparisons based on these sub-groupings. However, we can decrease the gap of this semantic issue using high-level semantic-based retrieval techniques, as follows:

High-level semantic-based retrieval techniques are divided to five major approaches: (1) using object-ontologies such as keywords or qualitative definitions; (2) using machine learning methods such as decision trees (Mitchel,

1997) or support vector machines (Burges, 1998); (3) using relevance feedback (Zhou and Huang, 2003); (4) using semantic templates such as sample images or keywords; (5) using textual information and visual feature extracted from images (Liu et al., 2007). The high-level features have the potential to achieve higher performance than low-level features (Liu et al., 2007).

Only machine learning and relevance feedback approaches do not require a textual description. We aim to build the system without any keyword requirement. Hence, we employ machine learning by means of decision trees ID3 (Iterative Dichotomiser 3) and relevance feedback to introduce high-level semantics into our system. Decision trees are also included to reflect human judgement. In our system, they train the similarity vectors using relevance feedback (details in chapter 6). Using relevance feedback, a user might imply which of the global grouping properties are relevant by consistantly selecting as relevant image examples that exhibit the desired visual property. For example, if similar images are returned and those exhibiting a particular arrangement of components, say set in diagonal lines, are selected as relevant, then a decision tree will be formed that accepts this bias based on computing decision thresholds using the information contained in the meta-feature vector, (in this case most likely summed local feature orientation and/or global feature cluster orientation). Alternatively, a user might consistantly select images containing sub-components that are widely spaced apart, and in this case the spatial feature proximity (or cluster spatial variance) will indicate images containing similarly widely spaced out components.

We can summarise the system framework in Figure 3.3. The system is implemented in chapters 6 and 7, and evaluated in chapter 8. Our system is divided into five processes, linked together so the output from one process acts as input to the next.

```
                    ┌─────────────────────┐
                   /   Input query        /
                  /     image            /
                 └─────────────────────┘
                           │
                           ▼
        ┌───────────────────────────────────────┐
        │   Interest point extraction by interest │
        │           point detector                │
        └───────────────────────────────────────┘
                           │ Interest points
                           ▼
        ┌───────────────────────────────────────┐
        │   Calculate local features and matched  │
        │                points                   │
        └───────────────────────────────────────┘
                           │ Matched points
                           ▼
        ┌───────────────────────────────────────┐      ┌────────────────────────┐
        │   Calculate shape descriptors (query    │      │   Structural and        │
        │   and database meta-feature vectors)     │──────│   appearance meta-      │
        └───────────────────────────────────────┘      │   features to describe  │
                           │ Shape descriptors           │   an image              │
                           ▼                             └────────────────────────┘
        ┌───────────────────────────────────────┐      ╭────────────────────────╮
        │   Compute similarity between            │      │   Database of           │
        │   query meta-feature vector             │◄────►│   trademark images and  │
        │   and database meta-feature             │      │   meta-feature vectors  │
        │   vectors                               │      ╰────────────────────────╯
        └───────────────────────────────────────┘
                           │ Similarity scores
                           ▼
        ┌───────────────────────────────────────┐      ┌────────────────────────┐
        │   Relevance feedback and relevance      │      │   Relevance feedback and│
        │   classification to select similar      │──────│   decision tree (ID3)   │
        │   trademark images                      │      │   relevance classification of │
        └───────────────────────────────────────┘      │   database images       │
                           │ Similar images              └────────────────────────┘
                           ▼
                  ┌─────────────────────┐
                  │   Display result     │
                  └─────────────────────┘
```

Figure 3.3 System flowchart for similar trademark image retrieval.

A query image triggers the first process, which extracts interest points from the image. The second process computes local features situated at interest point locations and matches candidate points. The third process is to generate

descriptors from the interest points and local features, which are stored as meta-feature vectors (containing structural and appearance information). The query meta-feature vector is calculated from the query image, and the database of meta-feature vectors is computed from both the query and database images. The fourth process matches the query image's meta-feature vector against a database of existing meta-feature vectors. The user then supplies relevance feedback about similar trademarks, and the system judges the image set using decision trees, and outputs a set of similar trademarks.

3.4 Summary

Our main goal is to develop a new machine perception grouping technique that provides an efficient way to retrieve abstract trademark images. Critical to this technique is support for human-like shape perception. Other requirements are interest point techniques and shape descriptors based on local features.

The system framework is divided into four main processes:

1. The extraction of interest points.

2. The calculation of local features and matched points

3. The generation of meta-feature descriptors from local features.

4. The matching of a meta-feature vector representing the query against a database of existing meta-feature-vectors.

5. The selection of similar trademark images.

In the next chapter, we investigate techniques to extract interest points to be used in trademark image retrieval system.

Chapter 4

Performance Study of Interest Point Detectors

In this Chapter, we investigate and evaluate intensity-based interest point detectors. We have investigated many detectors and chosen four effective detectors with which to experiment. We study and explain each detector in section 4.1 and test the ability of four interest point detectors to detect features within basic shapes and also within 20 samples of trademark images in section 4.2. An evaluation of the chosen interest point detectors is performed to compare their relative performances in section 4.3. We draw our conclusion in section 4.4.

4.1 Interest point detectors

According to Biederman's suggestion that humans can recognize structured shapes by their distinctive parts, we are motivated by the idea of using interest points in trademark image retrieval (Details in chapter2), because interest points represent visually salient information and are robust to partial occlusion In other words, using interest points would appear to have the potential to support human-like perception of shape. As a consequence, researchers have used interest points for successful object recognition.

We can divide the interest point detectors into two main categories: those which are based on intensity-based methods and those which based on boundary-based methods. Intensity-based methods use every pixel in an image directly to detect and then calculate interest points. Grey-level intensities can directly provide object characteristics (Horn, 1975; Allezard and Jurie, 2000). Boundary-based methods extract shape boundaries within the input image and calculate interest points from image contours. If contours can be extracted accurately, the boundary-based method has the potential to reflect the real shape of an object more accurately than any intensity-based method (Mokhtarian and Suomela, 1998; Mokhtarian and Mackworth, 1992). However, accurate image segmentation

requires human guidance (Zhang, 2006). On the other hand, the intensity-based method does not require segmentation or contour detection, and can even extract a partially occluded object or connected objects (Schmid and Mohr, 1997, Allezard and Jurie, 2000).

In this research, we aim to reduce human intervention with the system to retrieve similar trademarks. For that reason, our system will use raw images as input.

The intensity-based detectors use an input image directly, i.e. they use the image intensity field to calculate interest points. This kind of detector includes the Harris corner detector, Chabat detector, SUSAN (Smallest Univalue Segment Assimilating Nucleus) corner detector, and the Wavelet-based detector. The Harris detector uses a local autocorrelation analysis method (Harris and Stephens, 1988) and is reported to be robust to noise, rotation, and lighting (Schmid et al., 2000). The Chabat detector uses an orientation analysis method (Chabat et al., 1999) and is invariant to noise and lighting variations (Zhou et al., 2002). The Susan detector uses a nonlinear filter analysis method (Smith and Brady, 1997) and is robust to noise and viewpoint changes (Cho et al., 2003). The Wavelet-based detector uses the wavelet transform analysis method (Sebe and Lew, 2001) and is claimed to be invariant to image rotation and image scale changes (Sebe and Lew, 2003).

We implemented Harris and Chabat detectors, and used the original software in (Smith and Brady, 1997; Sebe and Lew, 2001). All of the detectors we evaluate have been implemented in the C/C++ computer language. We give details of each detector in the sections that follow.

4.1.1 Harris or Plessey detector

The Harris detector is based on the local auto-correlation function (Harris, 1988). The fundamental idea is to measure the change in correlation of a window (with respect to its starting position) as it shifts along an image in order to find the shape of the local autocorrelation function; candidate interest points are extracted by measuring autocorrelation changes produced by the shifting window. The change produced by the shifting window is given by $E(x,y)$; E is approximately

the autocorrelation function.

$$E(x, y) = Ax^2 + 2Cxy + By^2 \qquad \text{(Equation 4.1)}$$

$$= (x, y) \, M(x, y)^T \qquad \text{(Equation 4.2)}$$

M is the 2 by 2 symmetric matrix below.

$$M = \begin{bmatrix} A & C \\ C & B \end{bmatrix} \qquad \text{(Equation 4.3)}$$

We can calculate A, *B*, and *C* using the following Equations.

$$A = X^2 \otimes w \qquad \text{(Equation 4.4)}$$

$$B = Y^2 \otimes w \qquad \text{(Equation 4.5)}$$

$$C = (XY) \otimes w \qquad \text{(Equation 4.6)}$$

$$w = w_{u,v} = exp -(u^2 + v^2)/2\sigma^2 \qquad \text{(Equation 4.7)}$$

where *w* is a smooth circular window produced by the Gaussian function,

u and *v* are window positions along the *x* and *y* axes respectively.

In this implementation, the method calculates the image first derivatives along the *x* and *y* axes (*X* and *Y*).

$$X = I(x,y) \otimes (-1, 0, 1) = \partial I/\partial x \qquad \text{(Equation 4.8)}$$

$$Y = I(x,y) \otimes (-1, 0, 1)^T = \partial I/\partial y \qquad \text{(Equation 4.9)}$$

where *X* is the image first derivative along the *x* axis,

Y is the image first derivative along the *y* axis,

I(x,y) is the pixel image,

$\otimes$ is the convolution operator.

Because the shifting window auto-correlation function of a corner point has a minimum value when centred on the corner, we can use *M* to detect corner points. If α and β are the eigenvalues of *M*, corner points will have high α and β

values. We use the corner response (R) to detect corner points.

$$Tr(M) \; = \; \alpha + \beta \qquad = A + B \qquad\qquad \text{(Equation 4.10)}$$

$$Det(M) = \alpha \; \beta \qquad = AB - C^2 \qquad\qquad \text{(Equation 4.11)}$$

$$R \; = \; Det(M) - k \, Tr(M)^2 \qquad\qquad \text{(Equation 4.12)}$$

Then the strength of corner (R) is calculated by

$$R = (AB - C^2) - (k \, (A+B)^2) \qquad\qquad \text{(Equation 4.13)}$$

In this research, we use $k = 0.04$ as suggested in Harris's article (Harris, 1988). Corner points are then selected using a threshold (the determined minimum of R).

4.1.2 Chabat detector

A Chabat detector is based on a single derivative scheme (Chabat et al., 1999). It can detect corner points and the orientation of each corner. The detector computes the corner location and the direction of edges that join the corner. This method is made less sensitive to noise by using a local anisotropy method.

We detect possible corners using an analogy to the power spectrum of a line in the Fourier domain; a line gives an exact intensity direction. Thus, we assume that a strong intensity direction is reflected as a cluster within the power spectrum and thereby indicates the presence of a line.

The orientation $\theta(x,y)$ and the strength $g(x,y)$ of the anisotropy of an intensity pattern in one direction can be approximated as follows:

$$\theta(x, y) = \frac{1}{2} \tan^{-1} \frac{\iint_{\Omega} 2 I_x I_y \, dxdy}{\iint_{\Omega} (I_x^2 - I_y^2) \, dxdy} + \frac{\pi}{2} \qquad \text{(Equation 4.14)}$$

where $\theta(x,y)$ is the direction of anisotropy.

$$g(x, y) = \frac{\left(\iint_{\Omega} (I_x^2 - I_y^2) \, dxdy\right)^2 + \left(\iint_{\Omega} 2 I_x I_y \, dxdy\right)^2}{\left(\iint_{\Omega} (I_x^2 + I_y^2) \, dxdy\right)^2} \qquad \text{(Equation 4.15)}$$

where $g(x,y)$ is a measure of anisotropy along one direction.

We calculate the orientation of a corner by integrating the image first derivatives along the x-axis and the y-axis in order to reduce noise sensitivity. A value of g(x,y) close to 1 there is a strong gradient direction in one orientation; if the value is equal to 0 there is no dominant gradient direction in that area.

Chabat (1999) claims that corners have two properties; corners and junctions exhibit strong intensity gradients; corners have more than one dominant gradient direction. Then, the cornerness is based on these properties; the corners are computed by an anisotropic method along several directions of the intensity derivative (cornerness). The cornerness ($c(x,y)$) is calculated by

$$c(x,y) = \psi(g(x,y))\ //\nabla I(x,y) //$$ (Equation 4.16)

where

$$\psi(t) = (1-t)^{m}$$ (Equation 4.17)

, m = ½ as suggested by Chabat.

$//\nabla I(x,y) //$ is a gradient magnitude that calculated by

$$//\nabla I(x,y) // = \sqrt{\frac{\partial I^{2}}{\partial x} + \frac{\partial I^{2}}{\partial y}}$$ (Equation 4.18)

where $I(x,y)$ is a pixel in the image.

A cluster of high value cornerness is considered to identify the presence of one corner and the point with highest cornerness is then an exact corner location. Corner points are then selected using a threshold which determines a minimum value for cornerness. However, some edge points are detected as corner points, if corner area is wide or the image is very noisy. These points are eliminated in the final process.

The corner's orientation is calculated from the orientation of the edges to which it connects. The function that specifies the membership of edges that contribute to the corner (C_j) is computed by

---

$$s_{c(j)}(x,y) = g(x,y) \, //\nabla I(x,y)// \cos^n \alpha \qquad \text{(Equation 4.19)}$$

Equation 4.19 consists of two parts: $g(x,y)//\nabla I(x,y)//$ measures the edge strength exhibited by the corner, $\cos^n(\alpha)$ measures the orientation difference between an edge pixel and the corner arm. Following Chabat's original article (Chabat et al., 1999), we use $n = 3$ to calculate $s_{c(j)}(x,y)$.

The value of $s_{c(j)}(x,y)$ is small if an edge pixel (M_i) is not aligned with the direction of the corner point (C_j) - shown in Figure 4.1.
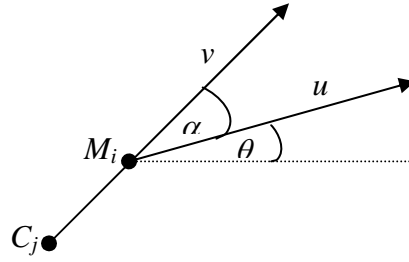
Figure 4.1 Direction of corner arm ($\overline{C_j M_i}$) and edge pixel (M_i)

The angle α is the angle of the orientation difference between an edge pixel (Mi) and corner arm ($\overline{C_j M_i}$).

$$\alpha = (u, v)$$

$$= \theta(x) - \tan^{-1}\left(\frac{C(y) - M(y)}{C(x) - M(x)}\right) \qquad \text{(Equation 4.20)}$$

Then, we calculate the corner confidence by the area under the $s_{c(j)}(x,y)$ histogram or the maximum of the $s_{c(j)}(x,y)$ histogram.

The histogram (H) is summed for all corner neighbourhood windows (Γ) with a given direction.

$$\forall \beta : \beta \in [0;2\pi): H(\beta) = \sum_{x \in \Gamma | \theta(x) = \beta} s_{c(j)}(x, y) \, \Phi_{c(j)}(x, y) \qquad \text{(Equation 4.21)}$$

$$\Phi_{c(j)}(x, y) = \Gamma(x, y) - \sqrt{(C(x) - M(x))^2 + (C(y) + M(y))^2} \quad \text{(Equation 4.22)}$$

$\Phi_{c(j)}(x,y)$ is a weighting function to decrease the value of pixels the further they are from the corner point centre. It is a positive function with a weight maximum

at C_j and zero when outside the window ($\tilde{\Gamma}$) We use Γ to be 10 pixels as in the original article (Chabat et al., 1999).

The number of peaks in the histogram shows the type of point. If it has two peaks, it is a corner. If it has three or more, it is a junction. To increase the robustness of the method with noise corrupted images, a corner with low corner confidence is eliminated. Finally, the number of corner points can be specified or a threshold set to select cornerness above a specific magnitude value.

4.1.3 SUSAN detector

SUSAN uses a non-linear filter to measure pixel brightness from a group of similar pixels in a local region (Smith and Brady, 1997). This non-linear filter uses a circular mask with a centre pixel as a nucleus. The mask area is also defined as the area of USAN (Univalue Segment Assimilating Nucleus). We can use USAN area to calculate the structure of the image.

Every pixel in an image is given its USAN value by using a small circular mask with that pixel as the nucleus. USAN has a maximum value when the nucleus is on a flat, non-varying area within the image. The USAN decreases to half of the maximum value when it is near a straight edge and it has its minimum value when it is on a corner. Therefore, the inverted USAN value can enhance edges and corners in images; this leads to an interpretation of SUSAN. SUSAN can suppress noise by a USAN function, if the noise is smaller then the USAN function.

The circular mask should be 37 pixels from the author's empirical study (Smith and Brady, 1997). The mask is placed over the pixel in the image to evaluate the pixel brightness. A USAN value is calculated by the equation below.

$$c(\vec{r}, \vec{r}_0) = e^{-\left(\frac{I(\vec{r}) - I(\vec{r}_0)}{t}\right)^6} \qquad \text{(Equation 4.23)}$$

where $\vec{r}$ is the pixel inside window area (circular mask),

$\vec{r}_0$ is the nucleus pixel (the centre point),

$I(\ )$ is the brightness of a pixel,

t is a brightness difference threshold to suppress noise,

c is the USAN value.

The total value of USAN area is calculated by

$$n(\vec{r}_0) = \sum_{\vec{r}} c(\vec{r}, \vec{r}_0)$$ (Equation 4.24)

where n is the number of pixels in the USAN area.

The USAN area (n) is compared to a geometric threshold (g); the threshold is fixed to $n_{max}/2$ where n_{max} is the maximum value of all USAN values; this threshold is used to suppress noise. USAN areas that have a USAN value less than half of the USAN maximum (n_{max}) are labelled as corners.

The geometric threshold (g) can be used to reveal the corner shape; a smaller value represents a sharper corner. Hence, the threshold can be fixed. The brightness difference threshold (t) is not sensitive to the structure of the corner, although it does control the number of detected corners; a smaller value gives more points. Smith claims that 25 is a suitable value for general real images and 7 for low contrast images (Smith and Brady, 1997). In addition, this threshold can be varied to specify a desired number of corners.

An initial edge response (R) is calculated to produce a corner strength image. The initial edge response equation is below.

$$R(\vec{r}_0) = \begin{cases} g - n(\vec{r}_0) & if \ n(\vec{r}_0) < g \\ 0 & otherwise \end{cases}$$ (Equation 4.25)

We can differentiate and thereby eliminate false positive corners from edges and noise by checking USAN's centroid and contiguity. The USAN centroid is computed as the distance from the nucleus to the centre of gravity. A short distance from the centre of gravity to the nucleus is defined as a false positive and eliminated. If a USAN area is smaller than the circular mask diameter, the centre of gravity ($\bar{\bar{r}}$) is calculated. The centre of gravity equation is as follows.

$$\bar{\bar{r}}(\vec{r}_0) = \frac{\sum_{\vec{r}} \vec{r} c(\vec{r}, \vec{r}_0)}{\sum_{\vec{r}} c(\vec{r}, \vec{r}_0)}$$ (Equation 4.26)

The contiguity of USAN is defined; a condition is that the lines running directly from the nucleus to the centre of gravity have to be a part of USAN.

Finally, non-maximum suppression is applied; the corners are selected by local maxima greater than zero from a window of 5 by 5 pixel regions (Smith and Brady, 1997).

4.1.4 Wavelet-based detector

Nicu Sebe and Micael Lew presented the Wavelet-based salient point detector in their article (Sebe and lew, 2001). This detector represents a trend to detecting evidence points at multiple resolutions by means of Wavelet transforms; a Wavelet transform is used to represent a number of discrete image scales. The aim is to investigate the change of information produced by the Wavelet representation at several scales. A Wavelet-based salient point detector can detect a point as a high variation of the Wavelet coefficients represented over several scales of the Wavelet transform.

At coarse resolution, a high absolute Wavelet coefficient implies that the wavelet support region contains a high global degree of variation, i.e. high image energy. Then, we can find a relatively salient point, or a point identifying a region of variation, by tracking the currently detected location at finer resolution within the next higher freqeuency Wavelet scale.

A Wavelet transform uses a scaling and translation function to calculate the frequency domain properties of the transform. A Wavelet coefficient ($W_{2^j} f$) is calculated by convolution of an image with a Wavelet function at multiple scales 1/2, 1/4, 1/8, …, 2^j; $j \in z$ and $j \leq -1$.

$$W_f = W_{2^j} f \quad \text{where} - J_{max} < j \leq -1. \qquad \text{(Equation 4.27)}$$

$$J_{max} = log_2 N \qquad \qquad \text{(Equation 4.28)}$$

where N is a size of image.

$$W_{2^j} f(n) = \sum_{k=-\infty}^{\infty} \left(g(k - 2n) \sum_{l=-\infty}^{\infty} h(l - 2k) A_{2^{j+2}} f(l) \right) \quad \text{(Equation 4.29)}$$

where $0 \leq n < 2^{j}N$,

g is the wavelet discrete filter,

h is the scaling discrete filter,

f(n) is an image,

A_s is the approximation result at each of the scales *s*.

We use an orthogonal Wavelet with compact support to be the "Mother" Wavelet function, following Sebe (Sebe, 2001); the advantage of this function is that it provides a non-redundant representation, and coefficients that can be traced over scale.

The set of coefficients, or child Wavelets, is given by the equation below.

$$(C(W_{2^{j}}^{d} f(x,y) = \left\{ W_{2^{j+1}} f(k,l) , 2x \leq k \leq 2x + 2p - 1, 2y \leq l \leq 2y + 2p - 1 \right\}$$

(Equation 4.30)

where p controls the shape of the Wavelet function,

$0 \leq x \leq 2^{j}N$,

$0 \leq y \leq 2^{j}N$,

N by *M* is the size of image,

$1 \leq d \leq 3$,

p=1 for Haar function,

f is an image,

d is the section number of the Wavelet coefficient.

The Wavelet coefficient at the scale 2^{j} ($W_{2^{j}} f$) is based on 2^{-j} image points; it shows information change at the scale 2^{j}. Furthermore, the maximum absolute value of the child coefficient set ($C(W_{2^{j}} f)$) reveals the most salient point; it corresponds to the maximum Wavelet coefficient at the scale 2^{j+1}.

We can extract a salient point by calculating a saliency value at all transform scales. The tracking process begins at highest scale and repeats to

extract the maximum coefficients recursively at lower scales until reaching scale 1/2. There are 2p final points. We then select points that have gradient values above a threshold. The saliency value of each detected point is calculated by the tracking process and comprises the sum of the absolute values of the coefficients. If the detected point has many salient values represented by the different child Wavelets found during scale tracking, the highest value detected serves as the saliency value for that point.

The saliency value is obtained by the equation below:

$$saliency = \sum_{k=1}^{-j} \left| c^k W_{2^j} f(n) \right| \qquad , 0 \leq n < 2^j N, -J_{max} \leq j \leq -1 \qquad \text{(Equation 4.31)}$$

Finally, we specify the threshold of the saliency value. Setting a high saliency value implies variation at the global level and a small saliency value implies variation at a local level. Therefore, the higher the threshold of the saliency value the fewer the number of salient points that results.

4.2 Development of interest point detectors

The interest point detectors extract potentially salient locations in an image that can be used in shape retrieval. Generally, all of the detectors specify the number of interest points, but some detectors require additional parameters. The Harris detector requires one parameter: the width of the Gaussian curve. A Chabat detector requires one parameter: the threshold for cornerness. The SUSAN detector and the Wavelet-based detector do not require additional parameters.

For testing, each detector is applied to the input image and outputs the interest points. The test applied for basic shape detection uses a rectangular figure to provide four corner points. The results are shown in Figure 4.2.

(a) (b) (c) (d)
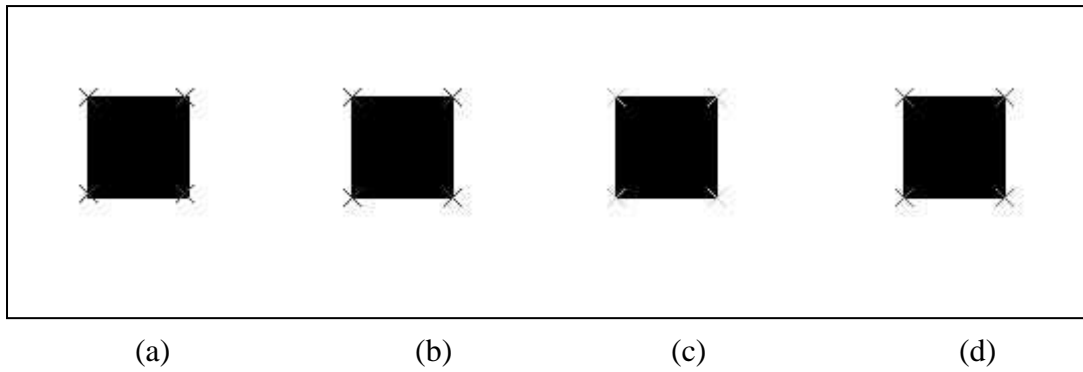
Figure 4.2 The results of (a) Harris detector, (b) Chabat detector, (c) SUSAN detector, and (d) Wavelet-based detector on rectangular figure.

The results show that every detector can detect correct corners in a basic shape such as a rectangle. However, the shapes found in real trademark images are more complex than this. Each detector tested could detect interest points within trademark images.

Consequently, the next testing phase employed sample trademark images. In this test, 50 interest points were extracted. The location of each detected interest point is depicted as a point in the test image, Figure 4.3.
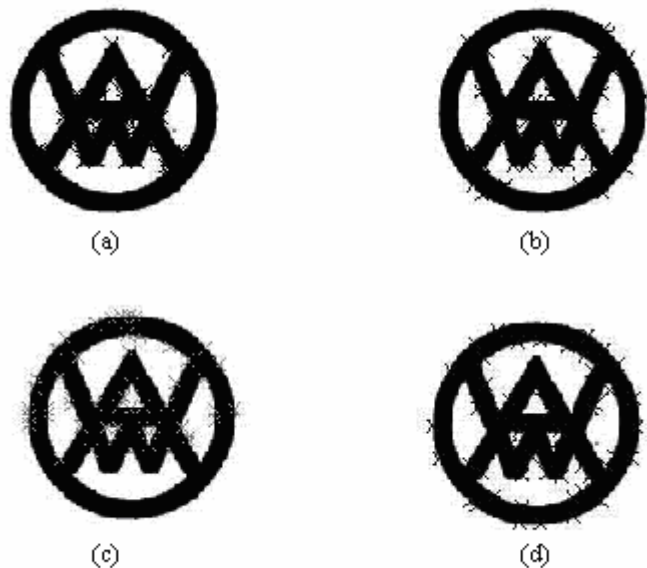
Figure 4.3 The results of (a) Harris detector, (b) Chabat detector, (c) SUSAN detector, and (d) Wavelet-based detector applied to a real trademark image.

However, the above experimental result also shows that each detector extracts different points, because of noise and their differencing underlying

computational approaches; therefore, these detectors have the potential to retrieve the different shapes that trademark components might comprise. However, we need to measure localization accuracy of the detectors; a highly accurate detector is suitable for use in shape retrieval (Schmid et al, 2000). The next section describes the experiment to measure the accuracy of the detectors.

We measured localization accuracy of the detectors using a series of 20 query images supplied by the UK Patent office obtained during the evaluation experiments of the ARTISAN system (Eakins et al., 1998). In the next section, the experiments designed to measure the accuracy of the interest point detectors are explained.

4.3 Experiments of interest point detector accuracy

The experiment aims to measure the accuracy of detectors to be used in trademark image retrieval. Repeatability is used to evaluate the accuracy of interest points (Schmid et al., 2000; Sebe and Lew, 2003). The main objective is to find the repeatability of each detector with different image scales and transformations (noise and rotation) (Schmid et al., 2000). A higher repeatability rate corresponds to a higher accuracy rate (Schmid et al., 2000). In addition, high accuracy detectors have the potential to be used in trademark image retrieval, because they are transformation invariant detectors.

The following sections describe the methodology to be used in the experiments, the measurement of detector repeatability and experimental results.

4.3.1 Methodology

The implementation of interest point detectors is programmed in C/C++ and executed on a personal computer. In each experiment, the number of interest points and the image test set are held constant. Each of the detectors is configured with the parameters specified in section 4.1.

The validation images are based on 20 trademark images obtained from the UK Patent office for user evaluation of ARTISAN. The images consist of multi-component and different abstract geometric shapes; the average number of components is seven.

Miller (1956) claimed that humans use seven, plus or minus two, chunks of short term memory to distinguish information where a chunk is any meaningful unit. The number seven then became the fundamental number for the limited capacity of short-term memory in all subsequent theories of memory. This leads to a number of seven points for each component. We use an average of seven components for each image (Eakins et al., 1996). The number of interest points is 50 (7x7 to the nearest 10).

In order to validate shape similarity measurement, we add scaled, rotated, and noise corrupted versions in our image test set. The image test set is based on 20 images that are transformed to provide two versions with differing degrees of noise added and three rotated versions; each set is scaled to three different sizes. The total number of images tested is 360. The details are explained in the next sections.

4.3.1.1 Image test set

There are three kinds of transformation used to generate images for the test set comprising size variation, noise addition, and image rotation. We found that at a small size (64x64 pixels), the image contains less information and it has many close points. Therefore, a suitable image size was considered to be more than 64x64 pixels. Generally, the size of the images is simplified to be a power of 2 (Petrou and Bosdogianni, 1999). The image sizes used are 128x128, 256x256, and 512x512 pixels. The image scaling transform applied is described in section 4.3.1.2.

Gaussian noise of either sigma = 10, or sigma = 30 is also applied to each image; these values are cited by Bovik as being typical in a real setting and to be difficult to remove by filtering (Bovik, 2000).

The test images are rotated by 15, 50 and 90 degrees since researchers reported that these values affect the changes in perceived orientation of an image (Goldstein, 1999; Elferink and Van Hof, 1988).

4.3.1.2 Image Scaling

The original trademark images are transformed by spatial scaling to produce test images of 128x128, 256x256, and 512x512 pixel dimensions. The scaled images

preserve the aspect ratio of the original images; this can be performed by filling to a square image. The filling process is shown in Figure 4.4.
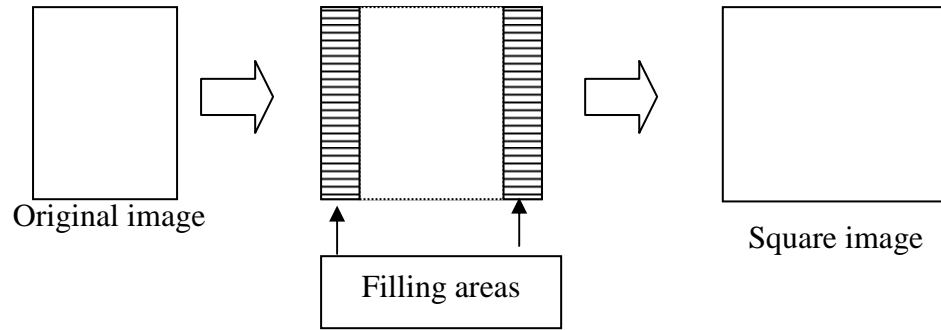
Figure 4.4 The process to fill an image to a square image.

Then the square original image is transformed by scaling as follows.

$$x' = x \times S_x \qquad \text{(Equation 4.32)}$$

$$y' = y \times S_y \qquad \text{(Equation 4.33)}$$

where x and y are the input pixel indices on the x axis and y axis respectively.

S_x and S_y are the scaling factors in the x and y dimensions and are given by

$$S_x = (n_x') / (n_x) \qquad \text{(Equation 4.34)}$$

$$S_y = (n_y') / (n_y) \qquad \text{(Equation 4.35)}$$

where n_x is the width of input image,

n_x' is the width of scaled image,

n_y is the height of input image,

n_y' is the height of scaled image.

Examples of scaled images are shown in Figure 4.5.
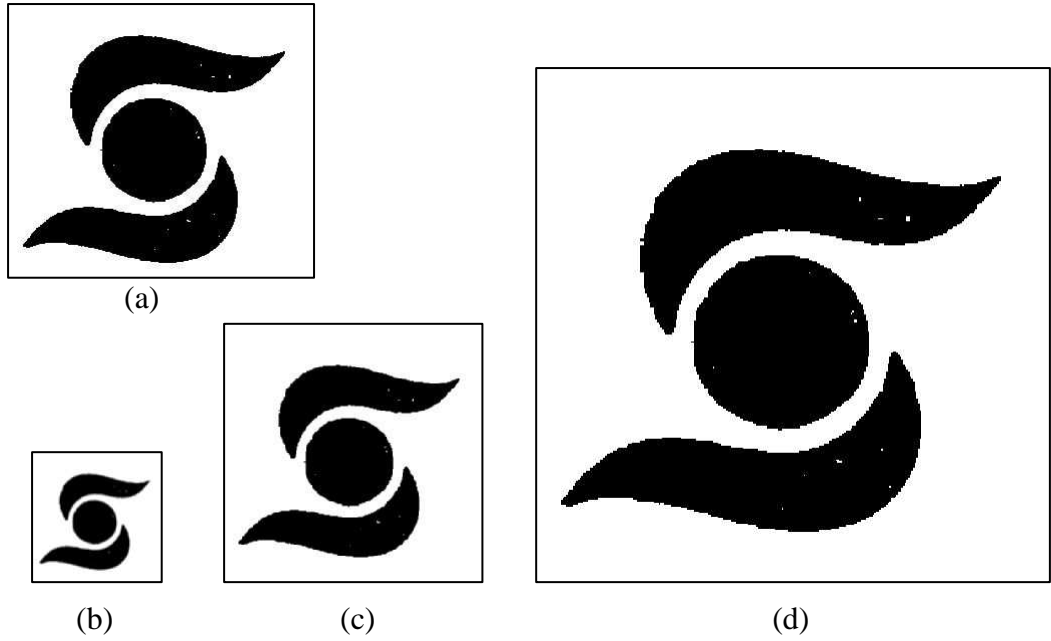
Figure 4.5 An original image in (a), scaled image size 128x128 in (b), scaled image size 256x256 in (c), scaled image size 512x512 in (d). All dimensions cited in pixels.

4.3.1.3 Image noise

Noise usually occurs during the image acquisition and transmission processes. It cannot be predicted accurately. However, it can be characterized by a probability distribution with a specific mean and standard deviation (Parker, 1997). Any transmitted image (*B*) is then modelled as the perfect image (*A*) plus noise (*N*).

$$B = A + N \qquad\qquad \text{(Equation 4.36)}$$

A and *N* are not related to each other. The noise N is a normal distribution with a zero mean and some standard deviation (Parker, 1997).

So each pixel in *B* is the sum of the image pixel value in *A* and a random, a Gaussian distribution noise value ($G_\sigma(x)$) with standard deviation as sigma (σ).

$$G_\sigma(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-x^2}{2\sigma^2}} \qquad\qquad \text{(Equation 4.37)}$$

Examples of noise corrupted images are shown in Figure 4.6.

a   b
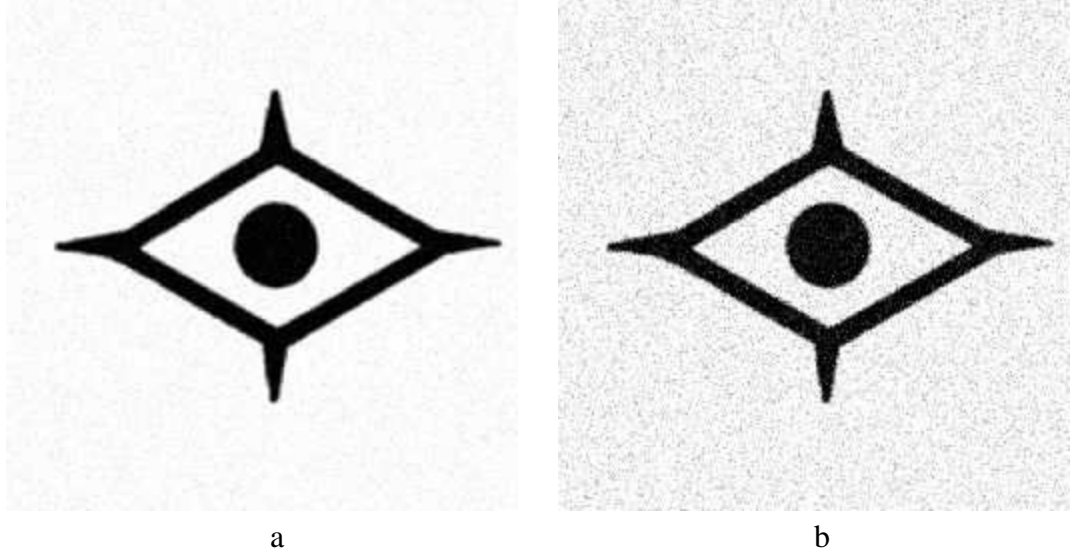
Figure 4.6 Gaussian noise corrupted images with sigma 10 in (a) and sigma 30 in (b)

4.3.1.4 Image rotation

Image rotation rolls images around a centre point. The transformed coordinates are given by

$$x' = x\cos\theta + y\sin\theta \qquad \text{(Equation 4.38)}$$

$$y' = -x\sin\theta + y\cos\theta \qquad \text{(Equation 4.39)}$$

where x and y are the input pixels of x axis and y axis respectively,

θ is the rotation angle between 0 and 360 degrees.

Examples of rotated images are shown in Figure 4.7.

(a)   (b)

(c)
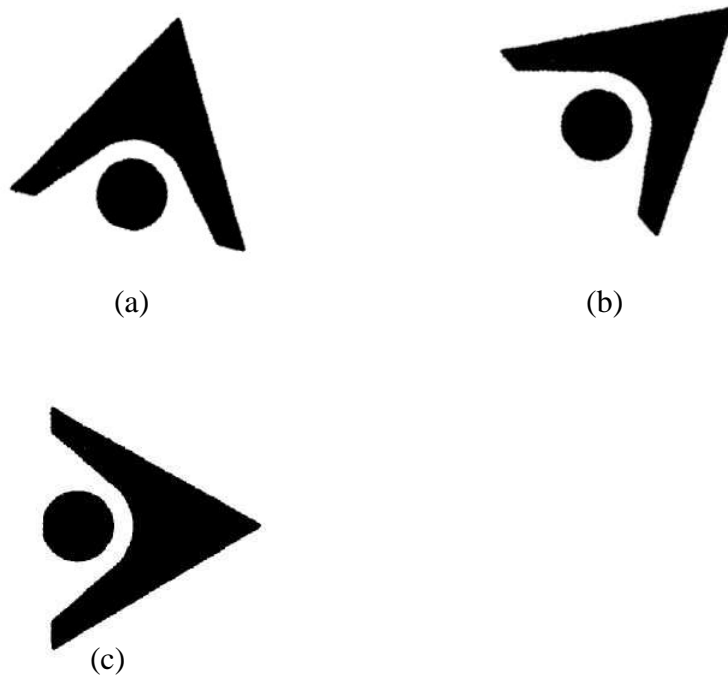
Figure 4.7 Rotated images with angle 15 degrees in (a) 50 degrees in (b) and 90 degrees in (c).

4.3.2 Repeatability

Repeatability measures the degree to which an algorithm detects the same features from variants of an original image (Schmid et al., 2000). Repeatability is one way to evaluate the accuracy of a detector; higher repeatability indicates a more reliable detector. This method can be used when matching a copy of an original image to a version of the same image that has undergone a known transformation. Therefore, a repeat point is reverse transformed back to its original location in order to compare its residual position error. Accordingly, this approach serves as an important method for measuring the localization accuracy of detectors.

Repeated points are not required to be at identical locations, as long as a point is in the neighbourhood of the original, it is regarded as repeated. This method uses a distance threshold to determine if a putuative repeat point is suffciently close to its original version to be accepted. Figure 4.8 shows the possible repeat point x from image I_1 on image I_2.
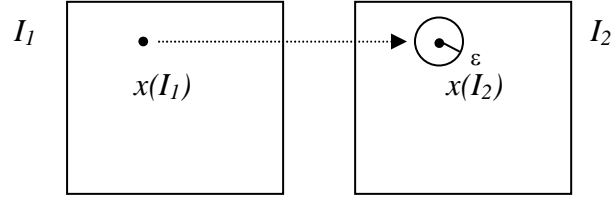
Figure 4.8 The possible repeat point x from image I_1 on image I_2

The neighbourhood size is ε pixels. Repeatability using neighbourhood matching is called pseudo repeatability. The pseudo repeatability rate ($r(I_i, I_j)$) of images I_i and I_j is calculated by:

$$r(I_i, I_j) = \sum_{n=0}^{\min(n_i, n_j)} R(x_n, I_i, I_j) / \min(n_i, n_j) * 100 \qquad \text{(Equation 4.40)}$$

$$R(x_i, I_1, I_2) = \begin{cases} 1, & if\ (\| x_i(I_1), x_i(I_2) \| < \varepsilon) \\ 0, & otherelse \end{cases} \qquad \text{(Equation 4.41)}$$

where x_i is a point in the image,

$\| x_i, x_j \|$ is the distance between points x_i and x_j.

In this research, the pseudo repeatability is in the range of 0 to 100.

4.3.3 Experimental procedure

In this section, we present the experimental procedure used evaluate the accuracy of interest point detectors applied to abstract trademark images.

The test set of images was based on 20 images and 9 examples of test images are shown in Table 4.1. The structure of tested images is shown in Figure 4.9. The total number of test images was 360. For example, one of the test-image set of size128x128 pixels is shown in Figure 4.10.

Table 4.1 Examples of testing images

Test image
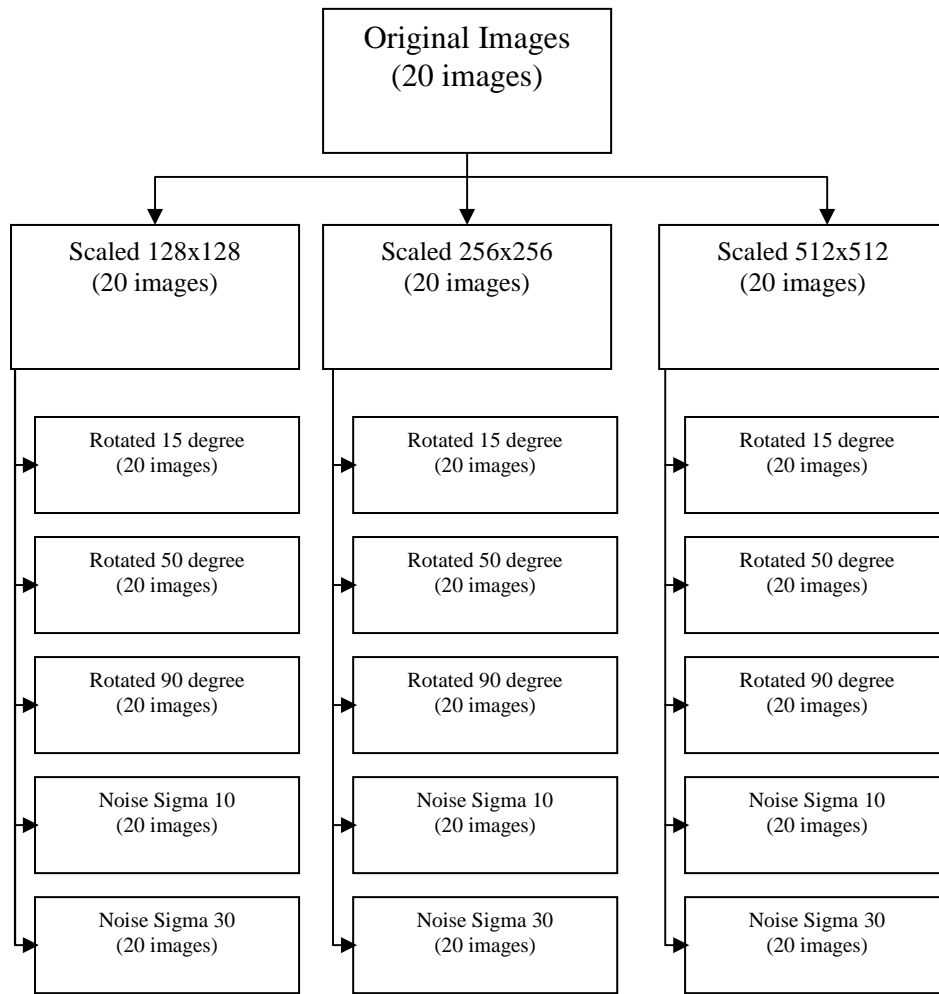
Figure 4.9 Summary of transformations to generate test images.

(a)

(b)

(c)

(d)

(e)

(f)
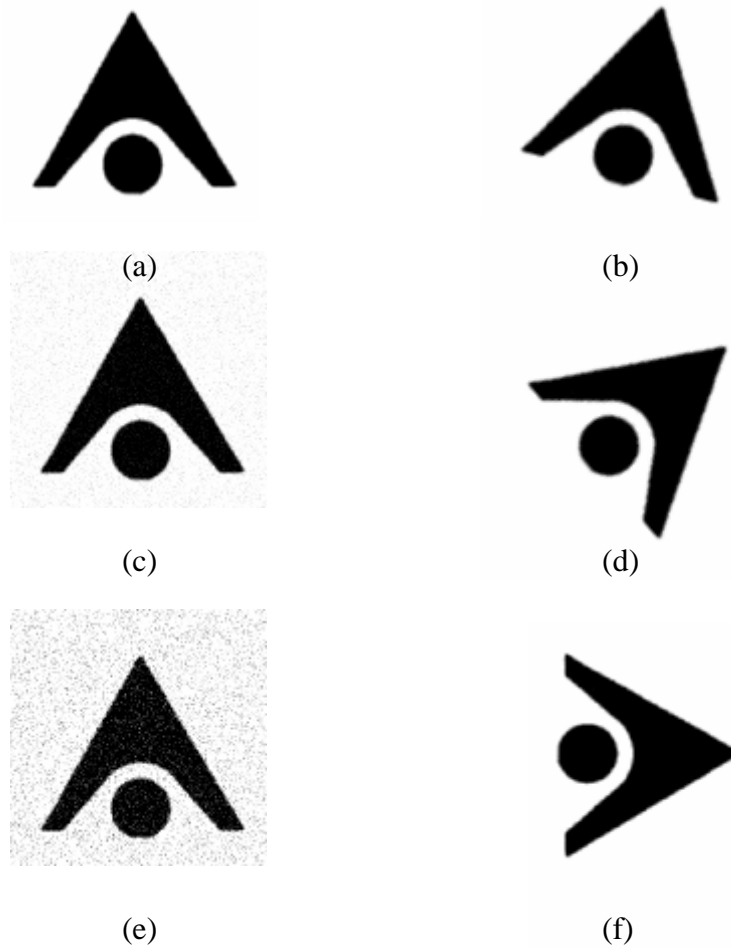
Figure 4.10 Image test set of image 1005896 with resolution 128x128pixels
when (a) original, (b) rotated 15 degrees, (c) noise with sigma 10,
(d) rotated 50 degrees, (e) noise with sigma 30, (f) rotated 90 degrees.

In section 4.1, we introduced interest point detectors: Harris, Chabat, SUSAN, and Wavelet-based detectors. Each detector extracted interest points and the repeatability score of these points was estimated for each test image. Many researchers use the repeatability score to compare the accuracy of detectors (Schmid et al, 2000; Sebe and Lew, 2003; Heidemann, 2004). To measure the repeatability, we had to consider particular image distortions comprising image rotation and image scaling. We then computed the repeatability score for each image from its interest points and averaged the repeatability scores over all tested images for each detector.

A summary of the experimental processes is shown in Figure 4.11. The

processes start by extracting interest points from the tested images. We divided the results into two categories from their input: reference and transformed points; the reference points were extracted from a reference image and the transformed points were extracted from a transformed image. Then, the repeatability of each image from each detector was calculated by computing the L2 norm difference between the positions of the reference and transformed points, as described in section 4.3.2. Finally, we summed and averaged all repeatability scores for each image and for each detector to determine the average repeatability score of each detector.

```
┌─────────────────────────────────────────────┐
│      Input images (Normal and transformed)   │
└─────────────────────────────────────────────┘
                     │ Extract interest points
                     ▼
┌─────────────────────────────────────────────┐
│          Perform Interest point detection     │
└─────────────────────────────────────────────┘
        │                              │
        ▼                              ▼
┌──────────────────────┐   ┌──────────────────────┐
│ Collect interest     │   │   Collect interest    │
│ points from          │   │ points from each      │
│ each reference image  │   │ transformed image     │
└──────────────────────┘   └──────────────────────┘
        │                              │
        ▼▼─────────────────────────────┘
┌─────────────────────────────────────────────┐
│          Calculate the repeatability          │
│        of each image (transformed image)      │
│              from each detector               │
└─────────────────────────────────────────────┘
                     │
                     ▼
┌─────────────────────────────────────────────┐
│  Sum and average the repeatability for each detector │
└─────────────────────────────────────────────┘
```
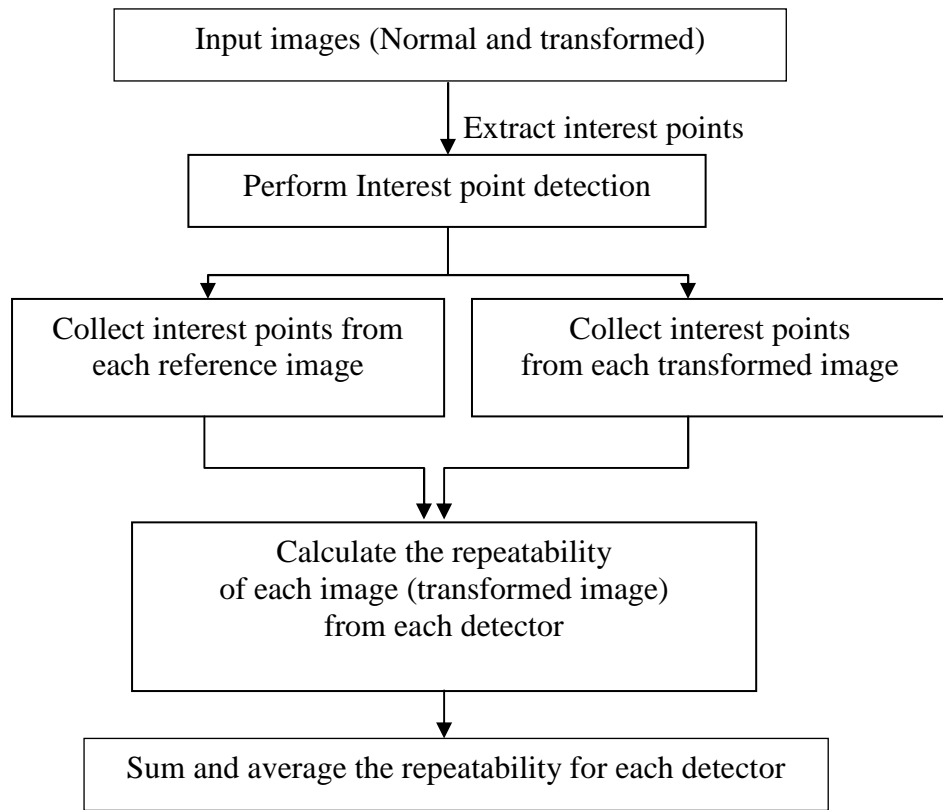
Figure 4.11 A summary of processes of experiments.

We summarise the experimental results in the next section.

4.3.4 Experimental results

A summary of results of all transformed image sizes of 128x128, 256x256 and 512x512 pixels are shown in Table 4.2; each tuple in the Table gives the average and standard deviation of the repeatability scores for each detector performed at the specified image size. The overall average for each detector is shown in Figure 4.12. The Harris, and Chabat detectors exhibited average repeatability scores

exceeding 50 %. Furthermore, these detectors exhibited better average repeatability than the SUSAN and Wavelet-based detectors, we believe because they provide gradient direction of corners; the Harris detector computes image first derivatives before smoothing by a Gaussian window; Chabat computes the anisotropy of the local intensity pattern and the intensity derivative. Hence, the Harris and Chabat detectors could reflect object boundary shapes in images and report better results than the others two detectors.

In addition, Harris and Chabat detectors were intensity-based detectors; they were degraded when the image artefacts present were increased by increasing the image size; they exhibited best result at an image size of 128x128 pixels because these reduced images contained least noise and artefacts.

Table 4.2 The overall repeatability of each detector.

	Repeatability (%)							
Detector Name	Image size						Average	
	128x128		256x256		512x512			
	Mean	SD.	Mean	SD.	Mean	SD.	Mean	SD.
Harris	84.62	13.72	74.62	16.19	63.04	14.60	74.09	17.29
Chabat	66.28	13.18	60.80	13.71	39.14	11.79	55.41	19.62
SUSAN	33.36	18.81	35.88	22.17	33.38	23.43	34.31	21.82
Wavelet-based	51.84	20.00	32.18	22.86	15.72	15.33	33.25	20.79
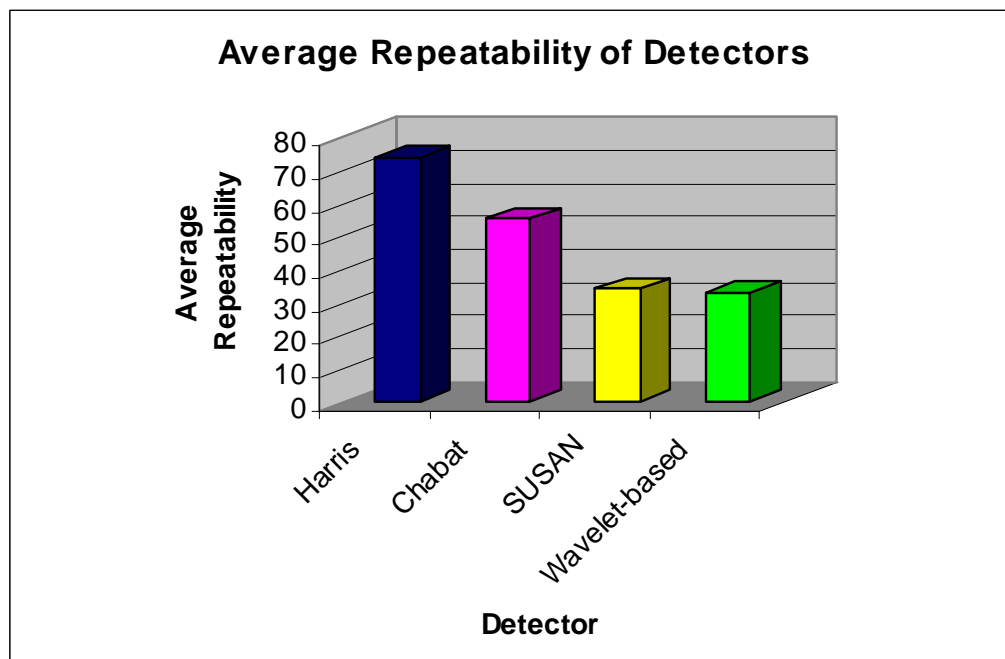
Figure 4.12 The overall average repeatability of each detector.

We investigated the result of experiments using different image sizes. The results of each transformation from image sizes of 128x128, 256x256 and 512x512 pixels are shown in Figures 4.13, 4.14 and 4.15 respectively.
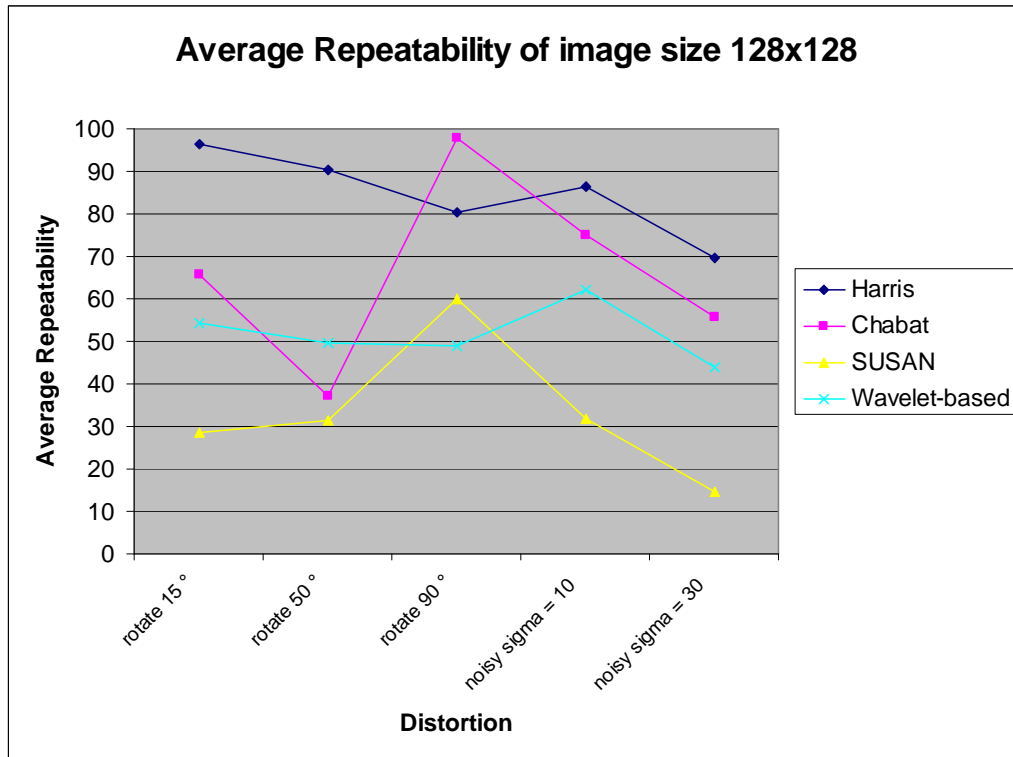
Figure 4.13 The average repeatability of image size 128x128 pixels.
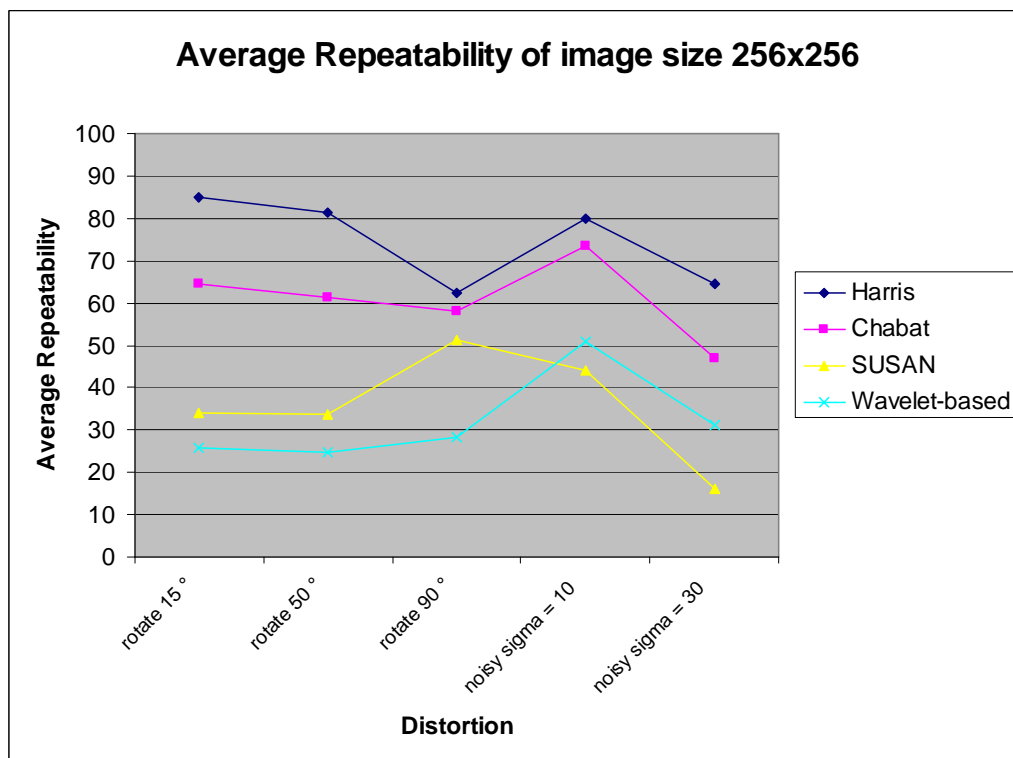
Figure 4.14 The average repeatability of image size 256x256 pixels.
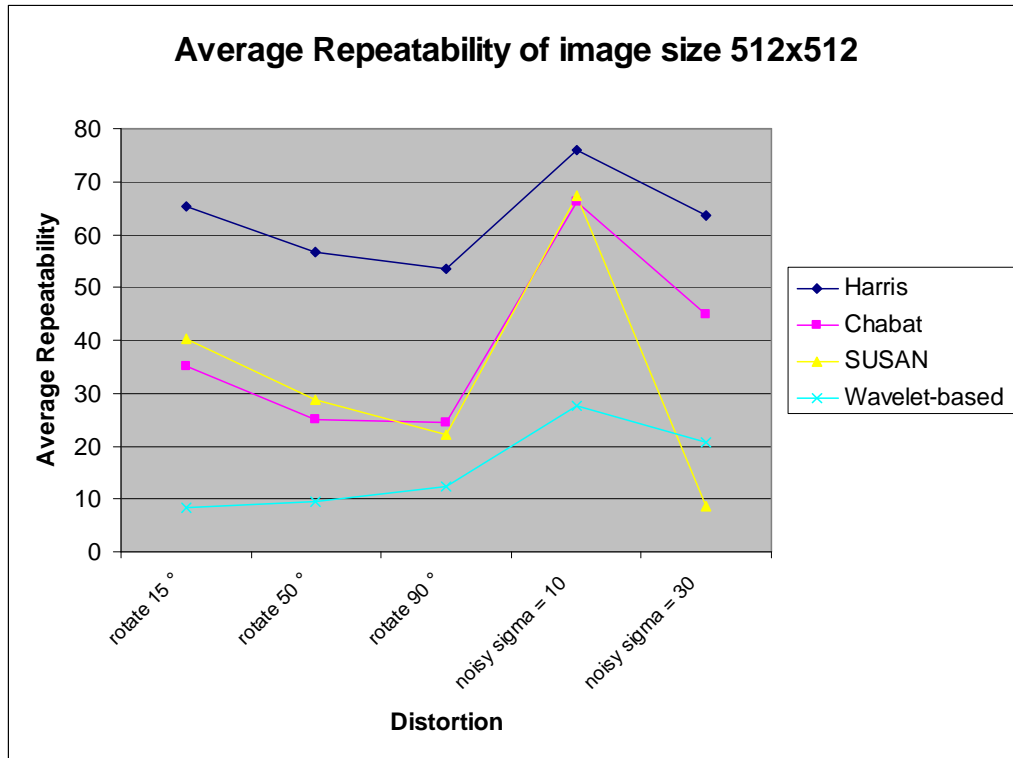
Figure 4.15 The average repeatability of image size 512x512 pixels.

From Figures 4.13, 4.14 and 4.19, the best result of all transformations in image size 128x128 pixels can be obtained, if we use the results from the Harris and Chabat detectors; the best result of all transformation in image size 256x256 pixels is from the Harris detector; the best result of all transformation in image size 512x512 pixels is also from the Harris detector.

4.3.5 Conclusion based on experimental work

The Harris detector demonstrated the best repeatability. In addition, the Chabat detector also produced good results with more than 50 % repeatability. Schmid (1997) claimed that a detector can be used for recognition of objects when it performs with a repeatability score of more than 50%. Therefore, the Harris and Chabat detectors have the potential to be used in trademark image retrieval applications.

4.4 Conclusion

Using interest points in trademark image retrieval is a new method. The results are sufficiently encouraging to justify finding the local features located at interest

points as described in the next chapter. In this chapter, we studied two tasks.

The first task was developing and testing interest point detectors. This task was to develop new software to implement the Harris and Chabat detectors and to evaluate the SUSAN and Wavelet-based detectors produced by the original authors. The validation methodology devised also tested the detectors with images containing basic shapes and also with real trademark images. The results show that every detector can detect corners correctly in a rectangular shape and extract a variety of different points in real trademark images.

The second task was to measure the repeatability of the interest point detectors. The Harris detector has the best repeatability and the Chabat detector also achieved good results with more than 50 % repeatability.

To sum up, we found that the Harris and Chabat interest point detectors have the potential to distinguish trademark images, because they have a high stability when retrieving interest points from transformed versions of the same image. According to our framework, we are interested in developing shape perception based on local features and interest points. The next stage of research explores how to use interest points in combination with local features (details in chapter 3). Therefore the next chapter will examine local feature extraction guided by the Harris corner detector as embedded within the SIFT algorithm.

Chapter 5

Performance Study of SIFT approach

In this chapter, we investigated and evaluated SIFT approach in order to measure the performance of local feature based on interest points. We briefly describe the SIFT approach in section 5.1. Then, we explain the implementation method in section 5.2, experimental procedure in section 5.3, experimental result in section 5.4, and conclude in section 5.5.

5.1 Introduction

In the previous chapter, we studied the performance of interest point detectors and found that interest points extracted by two gradient-based detectors were suitable for retrieving transformed trademark images. These results were sufficiently encouraging to justify continuing to next stage, i.e. to extract local features at interest point locations. From a literature review, the SIFT (Scale Invariant Feature Transform) descriptor appears to be the benchmark standard reliable feature representation to use in object recognition (Mikolajczyk, 2003; Lowe, 2004), and is applied in many image matching applications, e.g. in hand written word recognition (Rodriguez and Perronnin, 2008). In addition, the SIFT approach provides a robust feature detector that localises interest points, called SIFT *keypoints*, in scale and orientation to sample the SIFT feature, called a *keypoint descriptor*. Because, at this stage, we are interested in the performance of local features based on interest points, we selected the SIFT framework and keypoint descriptor for the experiments in this chapter. The Harris point detector is also employed by the SIFT algorithm, and this detector performed best on average of the point detectors evaluated in Chapter 4. SIFT therefore uses this same Harris point response function to select keypoints (detail in the next section) and extracts keypoint descriptor based on the gradient orientation histogram local to each interest point (Rodriguez and Perronnin, 2008),). The SIFT algorithm is explained in the following section.

5.2 SIFT

The SIFT algorithm was proposed by David Lowe (Lowe, 1999). This algorithm can extract interest points, keypoints, and generate a keypoint descriptor which is robustly invariant to general image transforms (rotation, translation and scale), and is also partially invariant to affine distortion, illumination change and noise (Lowe, 2004). The SIFT algorithm consists of four major stages comprising: scale-space extrema selection, keypoint localization, orientation assignment, and computation of keypoint descriptors.

In the first stage, potential interest points are extracted by finding local extrema of Difference-of-Gaussian filters at different scales. The Difference of Gaussians (*D(x, y, σ)*) function can be computed by subtracting two identical images each which have first been convolved with a Gaussian kernels with differing blur (*σ*) parameters. In a scale-space context, the kernel blur difference corresponds to the difference of two adjacent spatial scales separated by a constant multiplicative factor k:

$$D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma)$$ (Equation 5.1)

where

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$$ (Equation 5.2)

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma}$$ (Equation 5.3)

I(x, y) is the input image.

An example of computing the Difference-of-Gaussian convolution within a multi-resolution scale-space is shown in Figure 5.1.

Gausian   Difference of Gaussian

Scale (first Octave)

Scale (next Octave)

...
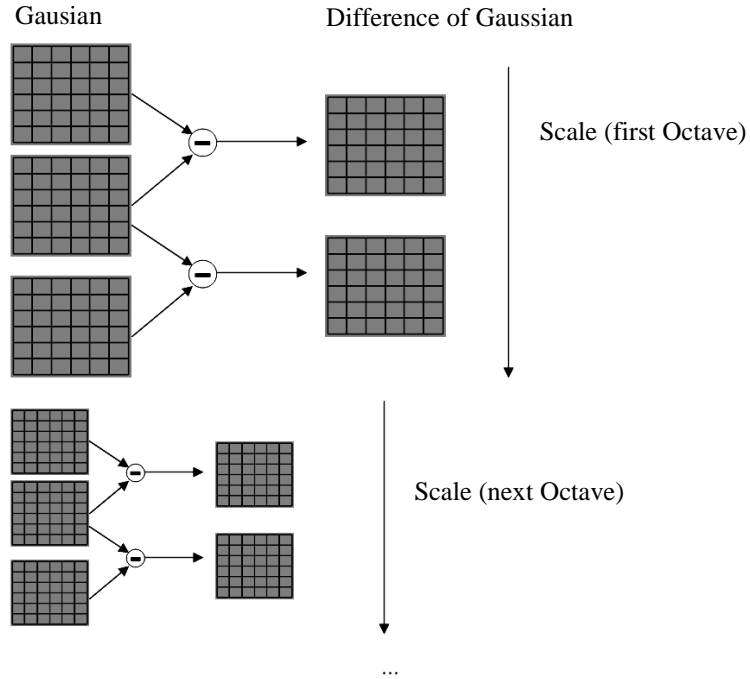
Figure 5.1 The computation of the Difference-of-Gaussian image pyramid.

The local extrema (minima and maxima) of $D(x, y, \sigma)$ in their own scale and one scale above and below are extracted as candidate points. The local extrema detection is shown in Figure 5.2.
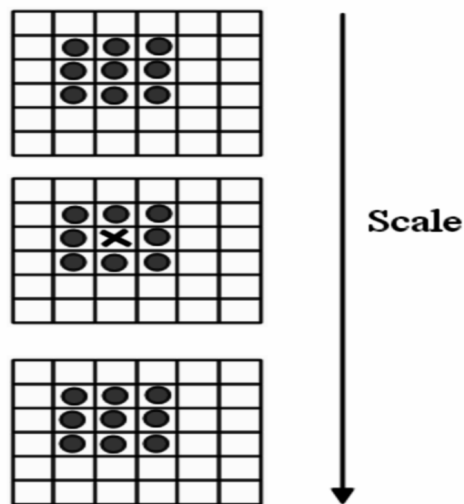
Scale

Figure 5.2 Local extrema detection: the pixel marked × is compared against its 26 neighbours in 3 by 3 regions at the current and adjacent scales (marked with circles).

In the second stage, each candidate point is localised to sub-pixel precision by interpolation. The candidate points that comprise low contrast or edge responses are eliminated to yield compact interest points that can be

localised uniquely in two dimensions.

For each candidate keypoint, the turning point of the local density function over a 3x3 system in x, y and scale is solved to define its sub-pixel/sub-scale position. Lowe's approach employs the quadratic Taylor expansion of the scale-space function $D(x, y, \sigma)$, given by:

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \qquad \text{(Equation 5.4)}$$

where D and its derivatives are computed at the candidate point and $(x = (x, y, \sigma))$ is the sub-pixel/sub-scale offset from this point.

The location of the extremum, $\hat{x}$, is calculated by taking the derivative of Equation 5.4 with respect to x and setting it to zero.

$$\hat{x} = \frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x} \qquad \text{(Equation 5.5)}$$

If the offset $\hat{x}$ is larger than 0.5 in any dimension, this means that the extremum lies closer to another candidate point. In this case, the candidate point is changed and the interpolation is performed at the new point. The final offset $\hat{x}$ is the interpolated estimate for the location of the extremum of the candidate point.

To discard candidate points with low contrast, the value at the extremum $D(\hat{x})$ is computed at the offset $\hat{x}$.

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x} \qquad \text{(Equation 5.6)}$$

If this absolute value is less than 0.03, the candidate point is rejected.

To discard candidate points which correspond to responses along edges, the ratio of principle curvatures (R) is calculated by means of the same approach used in the Harris detector (detailed in section 4.1.1).

$$R = \frac{Tr(H)^2}{Det(H)} \qquad \text{(Equation 5.7)}$$

where H is a 2 by 2 symmetric Hessian matrix described in section 4.1.1.

R for each candidate point is compared with the threshold (E).

$$E = \frac{(r+1)^2}{r}$$
(Equation 5.8)

where r =10 (Lowe, 2004).

Candidate points with R greater than E are discarded.

In the third stage, each interest point is assigned an orientation. The orientation is calculated from an orientation histogram of local gradients from the Gaussian smoothed image $L(x, y)$ at the scale of each interest point. For each image sample $L(x, y)$ at this scale, the gradient magnitude $m(x, y)$ and orientation $\theta(x, y)$ are computed using pixel differences:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2}$$ (Equation 5.9)

$$\theta(x, y) = \tan^{-1}((L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2)$$

(Equation 5.10)

The orientation histogram contains 36 bins, each representng10 degrees sampling precision over 360 degrees of orientation. Each point within a Gaussian-weighted circular window, of radius 1.5 times the scale of the keypoint, is added to the histogram and weighted by the gradient magnitude ($m(x, y)$). The highest peak in the histogram is selected and a keypoint generated for this orientation, and additional keypoints are created for any other local peaks whose size is within 80% of the highest peak.

For the fourth stage, SIFT keypoint descriptors are sampled by extracting a set of gradient magnitude weighted orientation histograms from a 16x16 pixel sampling patch centred on the keypoint location. The gradient magnitudes within the sampling patch are weighted by a centred Gaussian function having a sigma factor 1.5 times the scale of the interest point. This centre weighted sampling patch is then subdivided into a set of sixteen 4x4 pixel patches from which sixteen orientation histograms are computed from this 4x4pixel grid. Each histogram has

8 orientation bins over its corresponding 4x4 pixel support window and the resulting feature vectors are 128 elements in length, as depicted in Figure 5.3. These vectors are normalized to unit length to increase invariance to illumination changes.

Image gradients   Keypoint descriptor
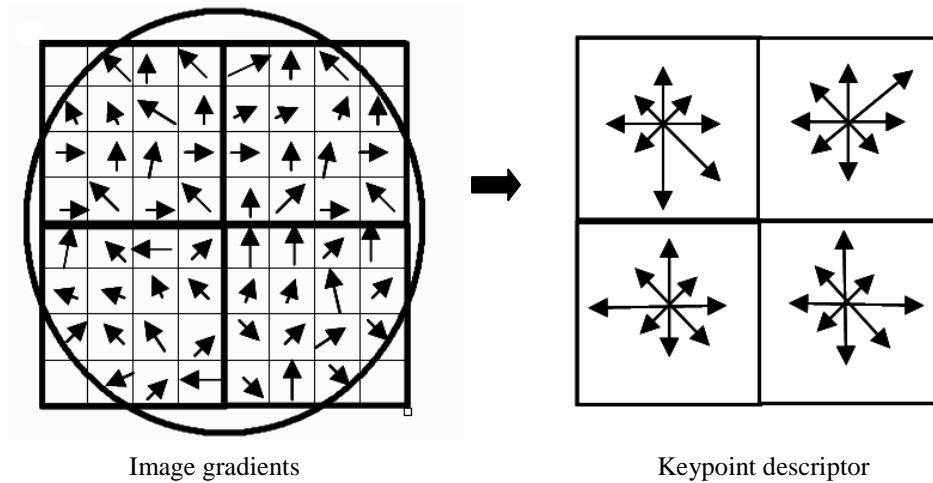
Figure 5.3 SIFT features comprise gradient magnitude weighted orientation histograms computed from a grid of sixteen 4x4pixel patches centered on each keypoint. The region is weighted by a circular Gaussian window indicated by the overlaid circle. Each orientation histogram is calculated from a 4x4 pixel support window and samples 8 orientation bins.

We use the SIFT algorithm as implemented by Lowe to extract interest points and local features, i.e. keypoints and keypoint descriptors. Then, we calculate the similarity between a query image and images in a database by matching the SIFT key point descriptors in the query image to those extracted from each image in the database images. The implementation methods are described in the next section.

5.3 Implementation Methods

The SIFT algorithm generates interest points or keypoints and keypoint descriptors from images that can be used in object recognition. Lowe also uses the Generalized Hough transform (GHT) to cluster matched points (Lowe, 2004). The GHT groups keypoint descriptors into clusters that match similar keypoints descriptors between two images and provides a high performance template matching scheme that supports matching between scaled, rotated, and partially occluded images (Lowe, 2004). Therefore, it appears that the SIFT algorithm in combination with the GHT could be used to retrieve trademark images in order to

extract local features and perform partial matching. The following experiment aims to retrieve transformed trademark images (normal, rotated and scaled images) using SIFT and GHT.

In this experiment, we propose to determine the match accuracy of the SIFT approach using scaled and rotated images. Correctly, matching images are identical to the query image but have been transformed to a different scale or rotation. We then investigate the performance of SIFT when attempting to retrieve modified trademark images.

5.3.1 Experimental procedure

In this section, we explain the procedure to set up experiments to evaluate SIFT for retrieving transformed images that contain scaled and rotated images. We show the database images in section 5.3.1.1, and describe the experimental processes in section 5.3.1.2.

5.3.1.1 Database images

In the experiment, the number of query images was 100 and the number of database images that were transformed from query images was 700 (400 by rotation and 300 by scaling).

The scaling adopted reduced the transformed images to half size and doubled size of the original images. The rotation parameters were 15 degrees, 50 degrees, and 90 degrees. Therefore, there were 300 scaled database images and the 400 rotated database images. The query images are shown in Figure 5.4 and the database images generated from the first query image are shown in Figure 5.5.
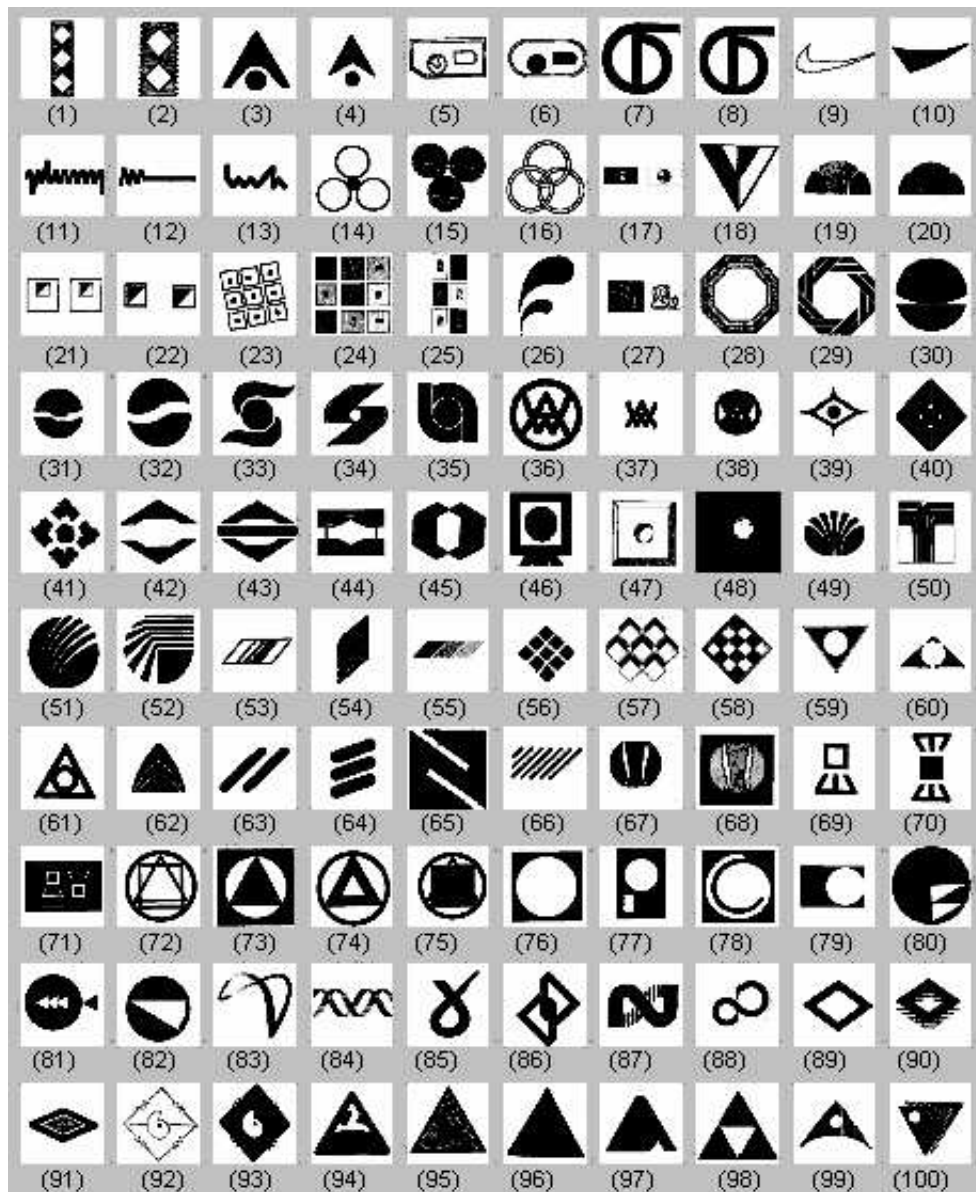
Figure 5.4 Query trademark images used during the experiments.
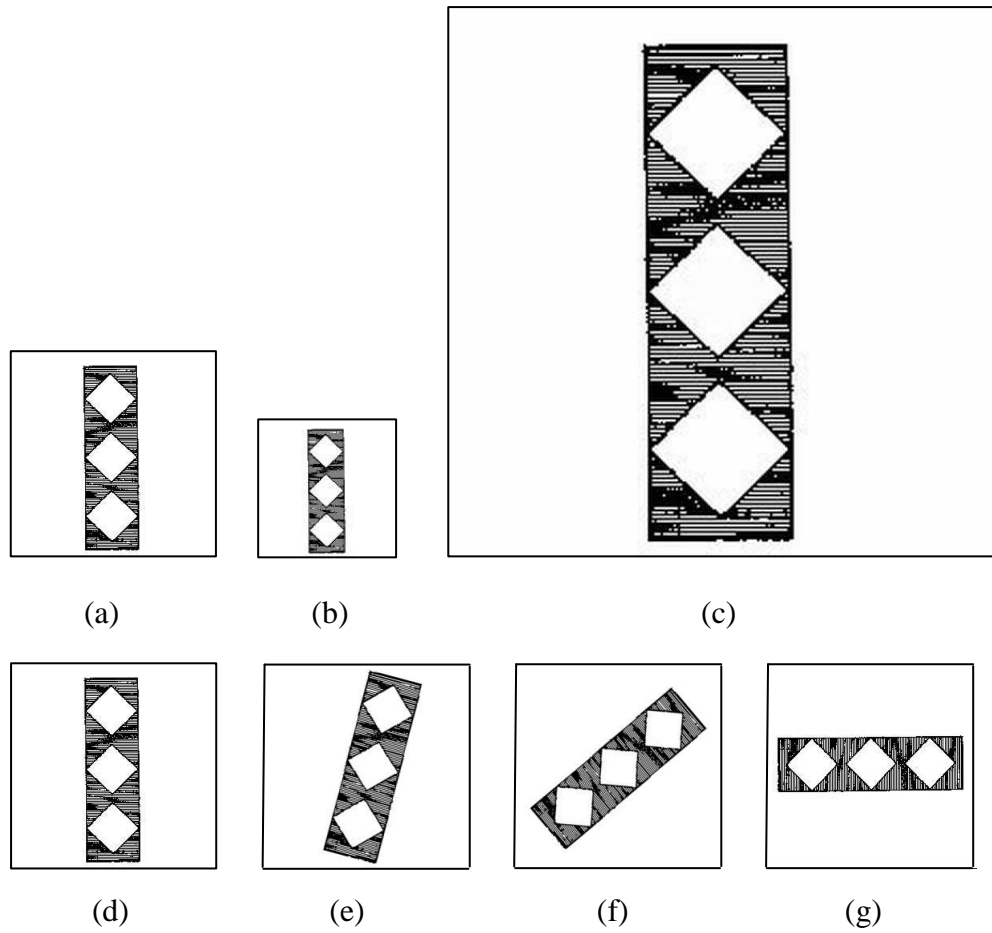
Figure 5.5 Examples of database images generated by transforming the first query image in Figure 5.4
in (a) - (c) are scaled images, and in (d) - (g) are rotated images.

We describe the experiment procedure in the next section.

5.3.1.2 Experimental processes

The overall process for the experiments is shown in Figure 5.6.
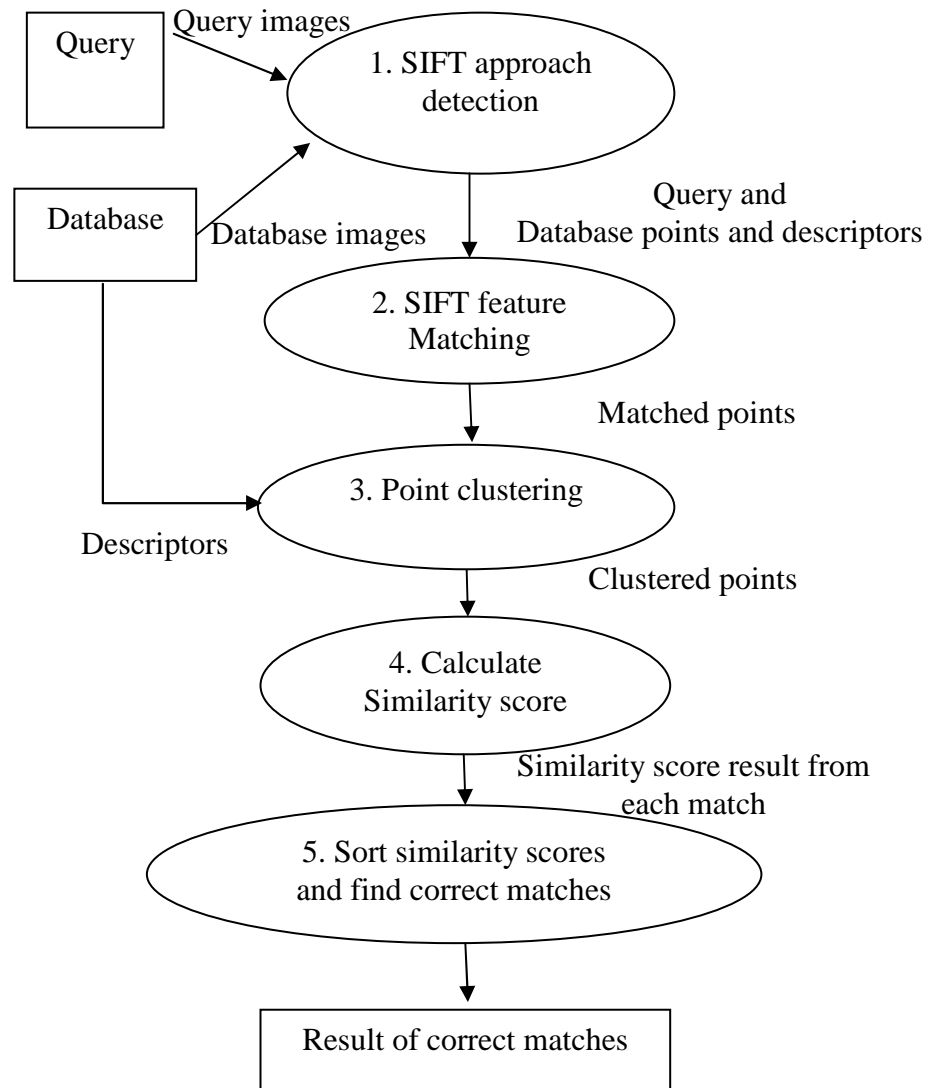
Figure 5.6 The overall process of experiments

First, interest points and descriptors were extracted from each image using SIFT. The software provided by David Lowe for extracting SIFT keypoints and keypoint descriptors is available at http://www.cs.ubc.ca/~lowe/keypoints/.

Second, each query image was matched with each database image. For robustness, we removed unmatched points that have a distance ratio greater than a preset rejection threshold. The distance ratio comprises the ratio of the nearest neighbour match score (the lowest distance) to the second nearest neighbour match score for each key point. The rejection threshold was set to 0.8 for all experiments following Lowe's suggestion that that this threshold can cull ~90 % of false matches (Lowe, 2004). The nearest neighbour (D_{lowest}) is defined in

Equation 5.11. $Ad_L(L_i)$ is a set of distances between each descriptor extracted from the query image and the closest match within all descriptors from a specific database image being matched. In order to compute each distance (match score), the Euclidean distance (D_L) of two descriptors, L_i and L_j, is calculated in Equation 5.12.

$$D_{lowest} = Min(Ad_L(L_i)) \qquad \text{(Equation 5.11)}$$

$$Ad_L(L_i) = \left\{ D_L(L_i, L_0), D_L(L_i, L_1), ..., D_L(L_i, L_n) \right\}$$

n is the number of all descriptors in the database image being matched.

$$D_L(L_i, L_j) = \sqrt{\left(L_i - L_j\right) * \left(L_i - L_j\right)} \qquad \text{(Equation 5.12)}$$

Third, the remaining key points were grouped by the Generalized Hough Transform (GHT) which gives clusters of matching points (Lowe, 2004). Suppose we have several points on arbitrary shape boundary as in Figure 5.7.
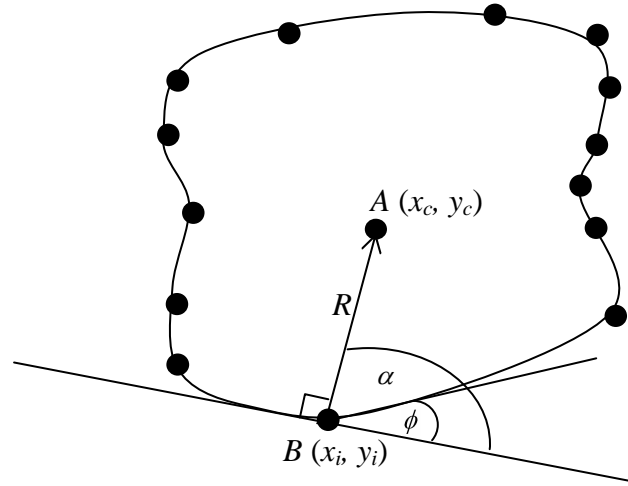
Figure 5.7 Geometry for GHT

Then, we define a point to calculate directional information and call it *A*, the centre point. For each interest point *B* on the boundary, the gradient direction (ϕ) is the orientation of the interest point given by SIFT (details in section 5.2), and the position of a reference point *r* is calculated by moving a distance *R* from the interest point in a direction α which is the angle of the line from each interest point to the reference point (see Figure 5.7). The *r* can vary and can be used in GHT for finding an arbitrary shape. A table to represent information to store shape data is called the R-table (Ballard, 1981).

The R-table is easily constructed by examining the boundary points of the shape. Table 5.1 shows the form of the R-table. The reference point (x_c, y_c) is selected as centre of all points.

We calculate the R-table using the algorithm in Figure 5.8.

Table 5.1 R-table format

Orientation of point	Set of radii and orientations where $l = (r, \alpha)$
ϕ_1	$l_{11}, l_{12}, l_{13}, \ldots, l_{1n1}$
ϕ_2	$l_{21}, l_{21}, l_{23}, \ldots, l_{2n2}$
…	…
ϕ_m	$l_{m1}, l_{m2}, l_{m3}, \ldots, l_{mnm}$

R-table construction
For all interest points (x_i, y_i) in the image
Assign the orientation of the interest point (ϕ) given by SIFT
Compute $r(x_r, y_r)$ and α
$\quad$ α is the angle of the line from each keypoint to the reference point.
$\quad$ $x_r = (x_c - x_i) / \cos \alpha$ $\qquad$ (Equation 5.21)
$\quad$ $y_r = (y_c - y_i) / \sin \alpha$ $\qquad$ (Equation 5.22)
Add an entry of (r, α) to the row indexed by ϕ

Figure 5.8 Overview of R-table construction

The GHT algorithm is derived from Ballard's original description of the GHT based on edge matching (Ballard, 1981) and is summarized in Figure 5.9.

GHT matching

Step 0: Create an R-table for each database image to be compared as described in Figure 5.8.

Step 1: Initialise the Accumulator array (*Acc*) of possible reference points to zero. The scale(*S*) and rotation (Ω) parameters are required to render the GHT invariant to scale and orientation. In this research, we use a range of scales of 0.5, 1 and 2, and rotations in the range from 0 to 360 degrees 30 degree increments, due to the large error bounds that are stated in Lowe's article (Lowe, 2004).

$Acc(x_{min}: x_{max}, y_{min}:y_{max}, s_{min}:s_{max}, \Omega_{min}: \Omega_{max})$

Step 2: For each keypoint in the query image:

Perform voting for location, orientation, and scale as follows:

Step 2.1: calculate possible reference point (x_c, y_c) for each pair of r_i and α_i in R-table

$x_c = x_i - r_i * S_i * cos(\alpha_i - \Omega_i)$ (Equation 5.15)

$y_c = x_i - r_i * S_i * sin(\alpha_i - \Omega_i)$ (Equation 5.16)

Step 2.2: Increment the accumulator array score in the four closest positions of each possible reference point by 0.25 times of the maximum model dimension (n_x and n_y) to avoid the size assignment problem as suggested by Lowe (Lowe, 2004). All possible reference points are (x_{c1}, y_{c1}), (x_{c1}, y_{c2}), (x_{c1}, y_{c2}), and (x_{c2}, y_{c2}).

$x_{c1} = x_c - x_c \, mod \, (0.25 * n_x)$ (Equation 5.17)

$x_{c2} = x_{c1} + (0.25 * n_x)$ (Equation 5.18)

$y_{c1} = y_c - y_c \, mod \, (0.25 * n_y)$ (Equation 5.19)

$y_{c2} = y_{c1} + (0.25 * n_y)$ (Equation 5.20)

Then the accumulator of each possible reference point is increased.

$Acc(x_{ci}, y_{ci}, s, \Omega) = Acc(x_{ci}, y_{ci}, s, \Omega) + 1$

Figure 5.9 Overview of GHT algorithm

A match is discarded if the number of points accumulated in the Hough Transform bin containing the greatest number of matching keypoints is less than three. At least three points are required to calculate the affine parameters relating the query image and the matched database image (Lowe, 2004).

Fourth, a similarity score for the points was calculated by taking the average of the lowest distance (D_{lowest}) from Equation 5.11 over all matching keypoints.

Finally, the match results for the query and database images were sorted by similarity scores and checked for correct matches. In the experiment, we only count a correct match if the retrieval images are the same as the query image in first three images in case of scaling and in first four images in case of rotation. The correct matches were then summarised for overall correct match numbers. We summarise the results of the above experiments in the next section.

5.4 Experimental results

The correct match results average for the experiment is shown in Table 5.2.

Table 5.2 The correct match results for the scaling and rotated database images in sub experiment 1 and 2

Image condition	Mean	Standard deviation
Scaling	98	7.92
Rotation	79.75	24.42

The mean number of correct matches under scaling was about 98 percent with a standard deviation of 7.92 percent. The mean number of correct matches under rotation was approximately 80 percent with a standard deviation of 24.42 percent. The table shows that SIFT performed better at matching scaled than rotated images.

This result would appear to indicate that SIFT is indeed promising as a method for retrieving both identical and altered versions of query images from an image database.

5.5 Conclusion

The SIFT algorithm combined with the GHT gave a good result on scaled and rotated images, we believe because SIFT keypoint descriptors possess high discriminability and are able to classify local parts of images. The GHT serves as a global comparison mechanism based on comparing local parts. In other words, it can achieve a partial match between the sub-components of compared trademark images. However, it does not take into account human decisions and preferences when creating and matching local features. Therefore, a further mechanism is required to consider the similarity of points that group into sub-components that also captures human perceptual judgement.

Appearance-based features are used to recognise objects (Hornegger et al., 2000). A further investigation will consider which appearance features are appropriate to provide perceptual grouping and a process to input user judgment when retrieving similar trademark images. We explain the details of these further investigations in the next chapter.

Chapter 6

Point-Based Grouping of Local Features

In the previous chapter, we found that the SIFT approach has the ability to identify local parts of images but only provides a single global similarity interpretation of detected local features that match to features extracted from similar database images. Furthermore, this matching process is invariant only to 2D affine transformations between such compared images. In this chapter, we investigate the formulation of meta-features which characterise how matching keypoint descriptor locations are distributed in order to give the system the ability to measure a number of "appearance properties" of the compared images. In the following sections, we explain how a perceptual grouping process can be applied to local features to implement meta-features (section 6.1), the experimental framework for meta-feature selection is presented in section 6.2, experimental results in section 6.3, appearance properties of meta-features in section 6.4, and conclusions are drawn in section 6.5.

6.1 Perceptual grouping by means of local features

We are motivated by the Gestalt laws of organization (details in chapter 2) to develop a perceptual grouping mechanism to characterize the global spatial arrangement of matching local features extracted from compared images. In order to achieve such visual grouping competencies, i.e. the ability to characterize the meta-structure of an image, we are guided by the Gestalt grouping laws to imitate aspects of perceptual grouping exhibited by human beings. As shown in the previous chapter, SIFT features are capable of retrieving transformed trademark images. However, an additional mechanism is required to provide perceptual grouping that describes local features appropriately in terms of their gross structural configuration and thereby enables image comparisons in terms of higher semantics. A meta-feature is an intermediate level of representation that is

calculated from the relations among the image components (Moreno et al., 2002). In this research, we are interested in three Gestalt properties (proximity, similarity, and simplicity) that can be directly measured by appearance meta-features. The above three Gestalt properties, and a set of proposed meta-features used to measure the Gestalt properties, are described below.

6.1.1. Global Similarity meta-Features

Similarity is used to group similar parts of an image; similarity can be measured in terms of the size, orientation, pattern appearance, or shape of an image part. These extended similarity meta-features require comparison between descriptors of image parts to measure this property (Jacobs, 2000).

We propose to calculate feature scale and orientation using SIFT in order to measure global size and global orientation similarity between sets of SIFT keypoint descriptors. We calculate the summation (total), mean, median, and RMS (Root-Mean-Square) of the differences in scale and orientation between sets of matched keypoints, extracted from query and compared database images respectively. Hence, we can measure the total global size and orientation difference between compared image features based on computing: summation, average and dominant global size and orientation differences using the mean and median statistics respectively; and also using the magnitude of scale and orientation variation differences by taking the RMS of the matching feature differences (Manikandan and Rajamani, 2008). An example of a global orientation difference between compared images is show in Figure 6.1. The arrows annotating the images in Figure 6 indicate the location (arrow start point) and scale of extracted keypoint descriptors, while the direction of the arrows indicate the canonical orientation directions for keypoint descriptors (details in section 5.2 of chapter 5). The global orientation difference meta-features calculated using sum, mean, median, and RMS of matching local feature differences of Figure 6.1 are summarised in Table 6.1.

<center>(a)   (b)</center>
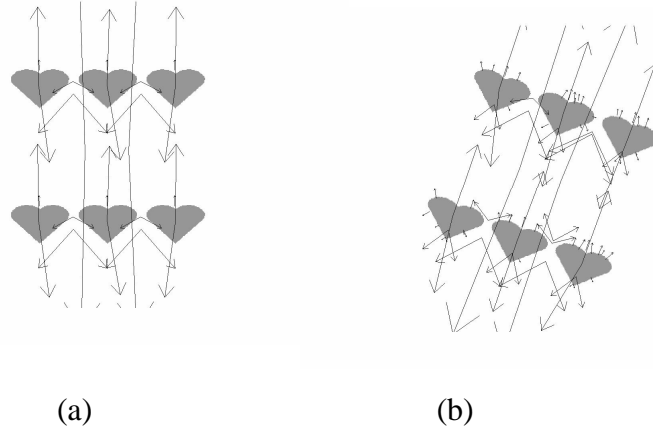
Figure 6.1 images of six of hearts with keypoints and their orientations (a) with l0 degrees rotation and (b) 20 degrees global rotation

Table 6.1 the global orientation difference meta-features of Figure 6.1

Image	Sum	Mean	Median	RMS
(a) and (a)	0.00000	0.00000	0.00000	0.00000
(a) and (b)	0.06715	0.05903	0.06064	0.05933

In addition, we investigate measures to implement a similarity score from SIFT matching, based on: the number of matching points, the average of multi-peak GHT (Generalized Hough Transform) scores, self similarity, and statistical moments of keypoint (x, y) positions.

An image similarity score is obtained by means of the SIFT algorithm as explained in section 5.3.1.2 of Chapter 5. The number of correctly matched points is determined by the GHT accumulator bin with the maximum number of points (details in section 5.3.1.2), since the matched keypoints in this bin exhibit the dominant common transformation between the compared images. Accordingly, the GHT is capable of identifying shapes by detecting peaks in its accumulator that correspond to feature groups with common transformations between features extracted from a query image and a set of database exemplars (Ballard, 1981). Therefore, the presence of multiple peaks in the GHT accumulator indicates matching feature sub-groups, each with differing common transformations.

Following human perceptual grouping principles, we are required to select only the significant parts of the image for similarity comparison (details in

sections 2.8 and 2.9 of chapter 2). Many researchers suggest that only the two or three dominant sub-parts which characterise an object are required to recognize that object (Biederman, 1987; Kirkpatrick, 2001). Hence, we only consider three maxima of the GHT accumulator to account for the dominant (in terms of numbers of matching keypoints) three matching sub-groups. The average of the multi-peak GHT scores (MGHT) is calculated by taking the mean score of the first three maxima of the GHT accumulator as follows:

$$MGHT = \frac{\sum_{i=1}^{npeak} Score(i)}{npeak}$$ (Equation 6.1)

where $npeak = 3$.

$$Score(i) = \frac{\sum_{j=1}^{ni} D_{lowest}(j)}{ni}$$ (Equation 6.2)

where ni is the number of matched points in the ith row of the accumulator. D_{lowest} is calculated by Equation 5.11 (see section 5.3.1.2 for details).

We are able to measure self *similarity* within the image by computing the self similarity of the top 10 keypoints (ranked by matching score). We select 10 as the significant maximum number of self similar keypoints in accordance with the number of items that can be held in short-term visual memory by humans (Miller, 1956). Each of the 10 most highly ranked keypoints will match to similar keypoints in the remainder of the keypoints extracted from a particular image and counts their relative frequency in a self similarity histogram. Our *self similarity* measure is computed by taking the mean of all points in the self similarity histogram and is summarised in Figure 6.2.

Self similarity algorithm
Step 1: Compute the nearest neighbour (D_{lowest}) of each interest point and
add it to an array of distances for each point. The D_{lowest} is described in section 5.3 of Chapter 5.
Step 2: Sort the array of distances into ascending order and select the top 10 points.
Step 3: For each selected point.
Step 3.1: Calculate the distance between each selected point and non-selected points (D_i)
$$D_i = \sqrt{(L_i - L_j)*(L_i - L_j)} \qquad \text{(Equation 6.3)}$$
where L_i and L_j are SIFT descriptors.
Step 3.2: If the distance is less than 0.2, according to (Lowe, 2004), then increase the self similarity number of this point by one.
Step 3.3: Save the self similarity number of this point in the self similarity histogram and repeat from Step 3.1 until perform all selected points have been considered.
Step 4: calculate average of all self similarity histogram numbers

Figure 6.2 The self similarity algorithm

Image moments can be used describe the 2D spatial configuration of raw grey levels or the positions of extracted image features and have been successfully used to measure similarity in object identification and pattern recognition (Mukundan & Ramakrishnan, 1998; Hu 1962). Moments of order zero up to three correspond to gross level image descriptions (including the ellipse characterising the measured spatial pattern distribution) while higher order moments hold more detailed data (such as asymmetries and skew in the observed spatial configuration) and are more sensitive to noise (Mukundan & Ramakrishnan, 1998). Image moments have the potential to represent shapes and geometric parts of images (Mukundan & Ramakrishnan, 1998).

As mentioned, moments could be calculated from different types of information extracted from an image, for example closed contours or a set of points (Tahri and Chaumette, 2005). Point-based image moments have been reported to yield high discrimination and good robustness when used to identify the same set of points in compared images (Tahri and Chaumette, 2005).

We would like to measure the similarity of two trademark images by computing the moments of the keypoint locations extracted from each of the compared trademarks. These moments are not shift invariant and in the case of trademark images, it means that the shape comparison is anchored with respect to the image frame. The moments are calculated by:

$$m_{pq} = \sum_{i=1}^{N} x_i^{\,p} y_i^{\,q} \qquad \text{(Equation 6.4)}$$

where x and y are the location of each matched keypoint in the image,

N is the number of the matched keypoints,

p and q are the orders of the computed moment.

In this research, we computed up to third order moments in order to analyse orthogonal transformations (Mukundan & Ramakrishnan, 1998). Paquet et al. (2000) state that the low-order moments could represent the most prominent aspects of the spatial configuration of keypoint features characterising an object. In addition, the moments can provide similarity information. *m00* represents the total mass of image points (Prokop et al., 1992). *m10* and *m01* are used to compute the centre of mass of image points (Prokop et al., 1992). *m02*, *m11*, and *m20* are used to calculate the moments of inertia (Prokop et al., 1992, i.e. the distribution ellipse major and minor axes. The proposed meta-features comprise *m00*, *m01*, *m02*, *m03*, *m10*, *m11*, *m20*, *m22*, *m30,* and *m33*.

6.1.2 Global Proximity meta-features

Proximity is used to group the connected area or nearest neighbours of components in an image; proximity can be inverse distance, touch, overlap or some combination of these. The proximity similarity meta-features also require comparison between descriptors of image parts to measure this property (Jacobs, 2000)

We propose to use SD (standard deviation) to compute the distance-error between sets of matched keypoints. Standard deviation is widely used measure of statistical dispersion (Manikandan, 2008). This meta-feature measures the residual spatial mismatch (error) between query and database images once these datasets have been aligned via an affine transformation. In other words, this measure computes the non-linear spatial differences between sets of matching image features following translation, rotation, scale, and sheer alignment. The algorithm for calculating the SD of distance-error between pairs of matched points is shown in Figure 6.3.

<div style="border:1px solid">

SD of error-distance between a pair of matched keypoint sets

Step 1: Extract matched points from the maximum of GHT accumulator.

Step 2: Calculate Affine parameters of matched points.

The solution is suggested by Lowe (Lowe, 2004).

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \qquad \text{(Equation 6.5)}$$

where (u,v) is a keypoint extracted from a database image,

(x, y) is a keypoint from a query image,

m1, m2, m3, and *m4* are affine parameters,

t_x and t_y are the translation parameters.

Then, we require at least 3 points to calculate the affine parameters:

$$\begin{bmatrix} x_1 & y_1 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_1 & y_1 & 0 & 1 \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m1 \\ m2 \\ m3 \\ m4 \\ tx \\ ty \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \vdots \end{bmatrix} \qquad \text{(Equation 6.6)}$$

Step 3: calculate affine transformation to map query keypoints to the database

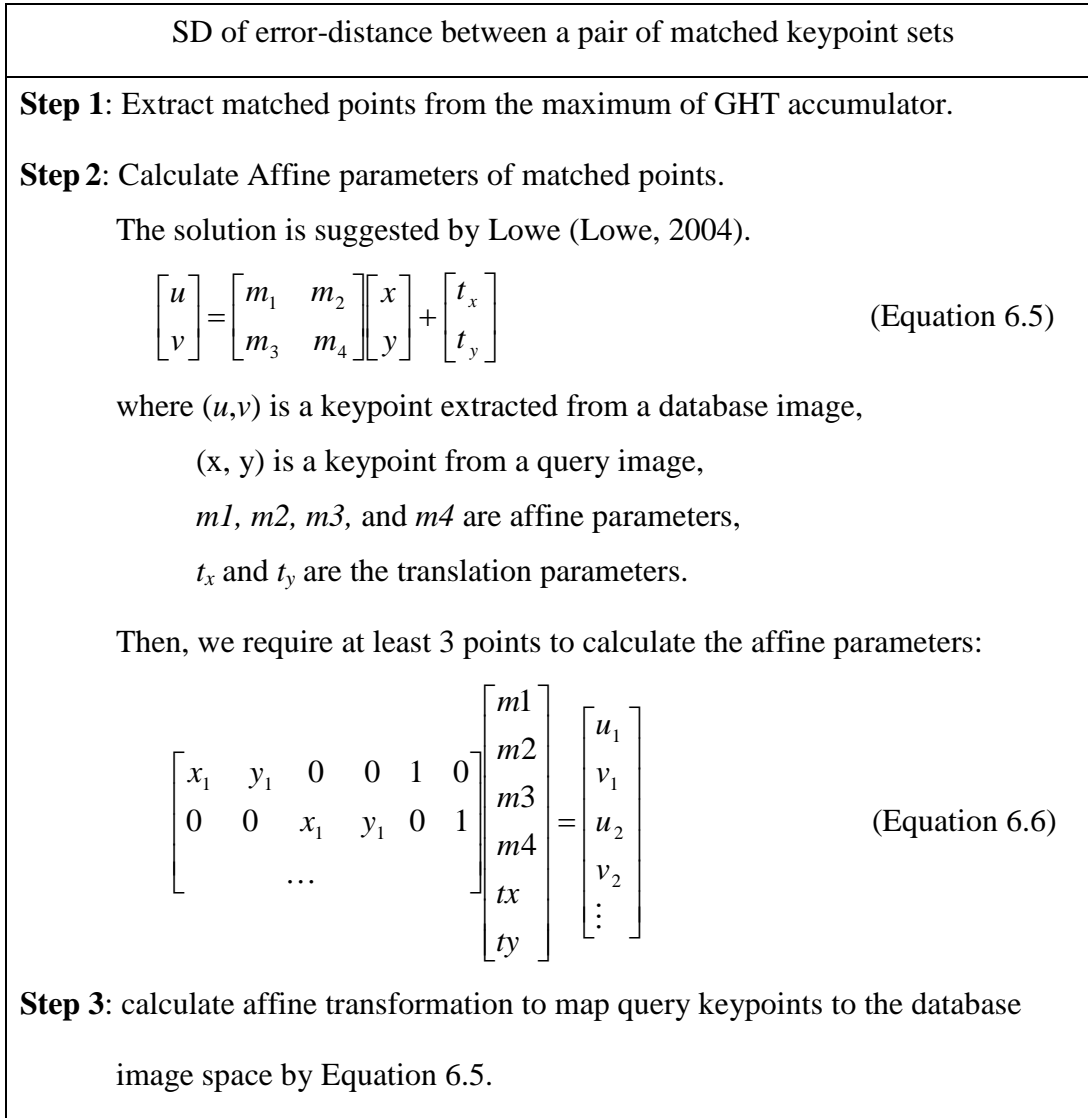image space by Equation 6.5.

</div>

Figure 6.3 The algorithm to calculate SD of the spatial distance error between a pair of matched and registered keypoint sets

SD of error-distance between a pair of matched keypoint sets (continued)
Step 4: find the distance error between each pair of matched database keypoints and affine transforms query keypoint using Equation 6.3 from Figure 6.2.
Step 5: calculate mean and standard deviation of all error distances in Step 4.
Step 6: find the inliner keypoint error distance. The error distances are selected if the distance is not greater than mean plus 2 times the SD from Step 5 (Jung and Lacroix, 2001).
Step 7: calculate the standard deviation of all distances in Step 6.

Figure 6.3 The algorithm to calculate SD of the spatial distance error between a pair of matched and registered keypoint sets (continue)

In addition, we propose to utilise the ratio of matched keypoints to total keypoints in the query image, and the ratio of matched points to total keypoints in the database image serves to measure overlap proximity, i.e. to quantify by how much (in terms of keypoints) do the compared images overlap. The ratio of matched keypoints to total keypoints for query image is calculated by the number of points in the maximum GHT accumulator divided by the number of keypoints in the query image. The ratio of matched points to total keypoints in each database image is calculated by the number of keypoints in the maximum GHT accumulator divided by the number of interest points in the database image. The ratio of matched keypoints and total keypoints (PMT) is calculated by:

$$PMT = \frac{n}{nq} \qquad \text{(Equation 6.7)}$$

where n is the number of matched points in the maximum accumulator,

nq is the total keypoints of the query image

The examples of the ratio of matched keypoints to total keypoints for the query image is shown in Figure 6.4 and summarised in Table 6.2.
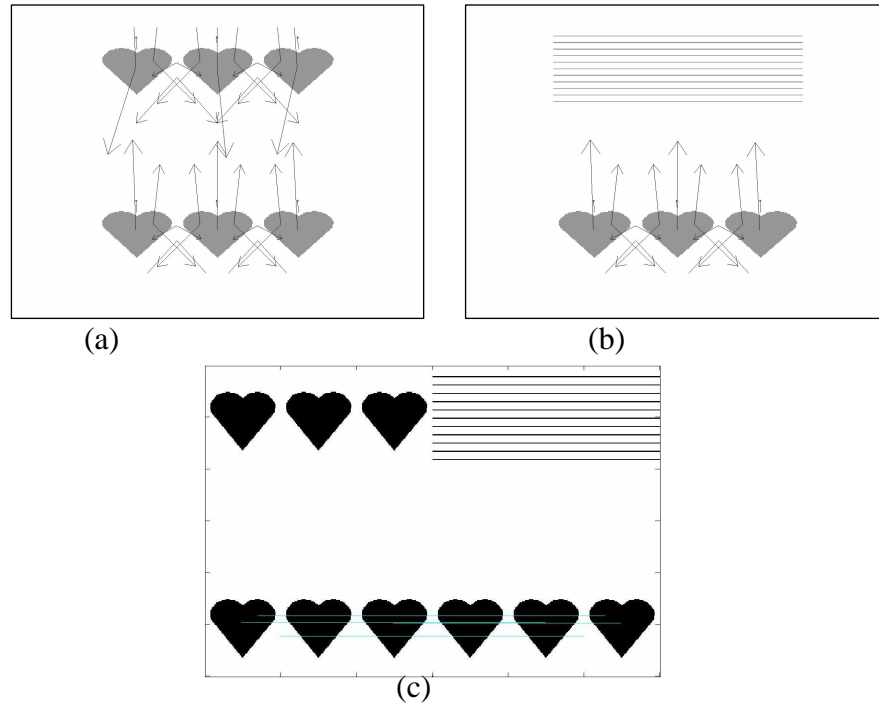
Figure 6.4 images of 6 hearts with keypoints and their repetition (a) 6 hearts (b) 3 hearts, and (c) matching result

Table 6.2 PMT meta-feature for the query image in Figure 6.4

Total keypoints	Number of matched points	PMT meta-feature
47	15	3.13

6.1.3 Global Simplicity meta-features

Simplicity is used to group the relative parts into simple components; simplicity of form can be represented by symmetry, regularity, or smoothness. Once more, to compute simplicity, the spatial configuration of the locations of keypoints matched between compared images are used to measure this property.

We propose to extract the vertical and horizontal symmetries exhibited within sets of matched keypoints. The vertical and horizontal symmetries are computed using the median distance from the centre of all matched keypoints in the major and minor (orthogonal) reflection axes of the matched keypoint spatial distribution axes respectively. Hence, we compute in essence an asymmetry score, as our measure will return a score of zero for a perfectly symmetrically configured keypoints and the score will increase in value as asymmetry is

introduced into this configuration. The algorithm to calculate the vertical and horizontal symmetries of sets of matching keypoints is shown in Figure 6.5.

Vertical and horizontal symmetries of matching keypoint sets

Step 1: Calculate axis of rotation by:

$$Shift\_axis = 0.5 * tan^{-1}((2*m11) / (m20 - m02))$$ (Equation 6.8)

where *m11*, *m20*, and *m02* are moments from Equation 6.4

Step 2: Rotate matching points by - *Shift_axis.*

ry= x* sin(- Shift_axis) + y*cos(- Shift_axis) (Equation 6.9)

rx= x*cos(- Shift_axis) - y*sin(- Shift_axis) (Equation 6.10)

where (*x*, *y*) is the matched point.

Step 3: Calculate the centre of rotated points (*cx* and *cy*)

as the mean of all rotated points.

$$cx = \frac{\sum_{i=1}^{n} rx}{n}$$ (Equation 6.11)

$$cy = \frac{\sum_{i=1}^{n} ry}{n}$$ (Equation 6.12)

where *n* is the number of matching points

Figure 6.5 The algorithm to calculate vertical and horizontal symmetries of matching sets of keypoints

Vertical and horizontal symmetries of matched sets of keypoints (continued)

Step 4: Calculate the error distance for each pair of keypoints (D_{err}) in each axis.

> **Step 4.1:** separate rotated points into two sets
>
> > If $ri > ci$
> >
> > > Add this point to set1 (*pset1*)
> >
> > Else
> >
> > > Add this point to set2 (*pset2*)
> >
> > End
> >
> > Where *ri* is a location of a rotated keypoint in each axis and
> >
> > > *ci* is a location of centre of a set of rotated keypoints in
> > >
> > > each axis
>
> **Step 4.2:** calculate the error distance (D_{err}) of each point in the two sets
>
> $$D_{err}(i) = \min(D_L(pset1(i), \forall pset2(i)))$$ (Equation 6.13)
>
> where
>
> $$D_L(p_1(x_1, y_1), p_2(x_2, y_2)) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$
>
> (Equation 6.14)

Step 5: Calculate the median of error distances of each axis (D_{err}). The horizontal symmetry is the median of the y axis and the vertical symmetry is the median of the x axis.

Figure 6.5 The algorithm to calculate vertical and horizontal symmetries of matching sets of keypoints (continued)

An example of the vertical symmetry score computed on the keypoints matched between an unmodified heart shape and an asymmetric heart shape is shown in Figure 6.6 and summarised in Table 6.3.

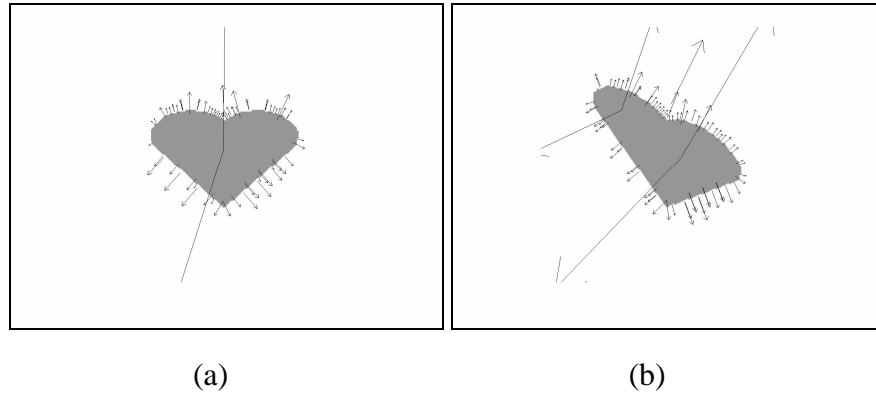<center>(a)   (b)</center>

Figure 6.6 images of a heart shape with interest points and their orientations overlaid for (a) 0 degrees of asymmetry and (b) 50 degrees of asymmetry

Table 6.3 the vertical symmetry of matched points Figure 6.6

Image	Vertical symmetry of matched keypoint sets
(a) and (a)	0.06560
(a) and (b)	0.24932

In this chapter, we plan to find appropriate meta-features to the system with a perceptual grouping mechanism. In order to carry this out, we design a set of validation experiments to test the proposed meta-features and we explain the experimental framework in the next section.

6.2 Experimental Framework

An important criterion by which to validate the proposed meta-features is which meta-features retain most of the variation present in the underlying feature data. Using appropriate meta-features could help to identify shapes (Aguirre et al., 2007). The appropriate meta-features are determined by removing redundant meta-features. In addition, employing only efficient (non-redundant) meta-features could increase the overall system performance. Therefore, we investigate the proposed meta-features in order to get remove redundant meta-features or to reduce dimensionality of the meta-features (Bashir et al., 2004). We present the implementation methods below.

In the experiment, we calculated eigenvalues and eigenvectors of the

---

proposed meta-features using Principal Component Analysis (PCA) (Smith, 2002). We select the meta-features with high eigenvalues as the number of principal components of major significance and further analyse this set of meta-features using factor analysis (Field, 2000; Aguirre et al., 2007). Next, we group all meta-features into significant meta-feature sets as determined by their factor loadings. Then, we can assign meta-feature sets comprising candidate major meta-features.

In the next section, we describe the experiment procedure.

6.3 Experimental procedure

In this section, we explain the experimental procedure to extract and evaluate the meta-features of major significance. We present the database images in section 6.3.1, summarise our proposed meta-features in section 6.3.2., and describe the experimental processes in section 6.3.3.

6.3.1. Image data

Trademark images provide an important basis for shape retrieval testing (Eakins, 1998; Jain and Vailaya, 1996) (details in chapter 2). Furthermore, Gestalt principles are overtly used in trademark design (Arntson, 2006). For these reasons, we predict that trademark images will serve as an appropriate dataset when testing the Gestalt properties of the meta-features. The system used 33 query images and 100 database images to evaluate the proposed meta-features, including part of the database used in the evaluation of the Artisan system (Eakins et al, 1998). All the trademarks are binarised images of 256x256 pixels. The 33 queries are shown in Figure 6.7 and the database images are shown in Figure 6.8.
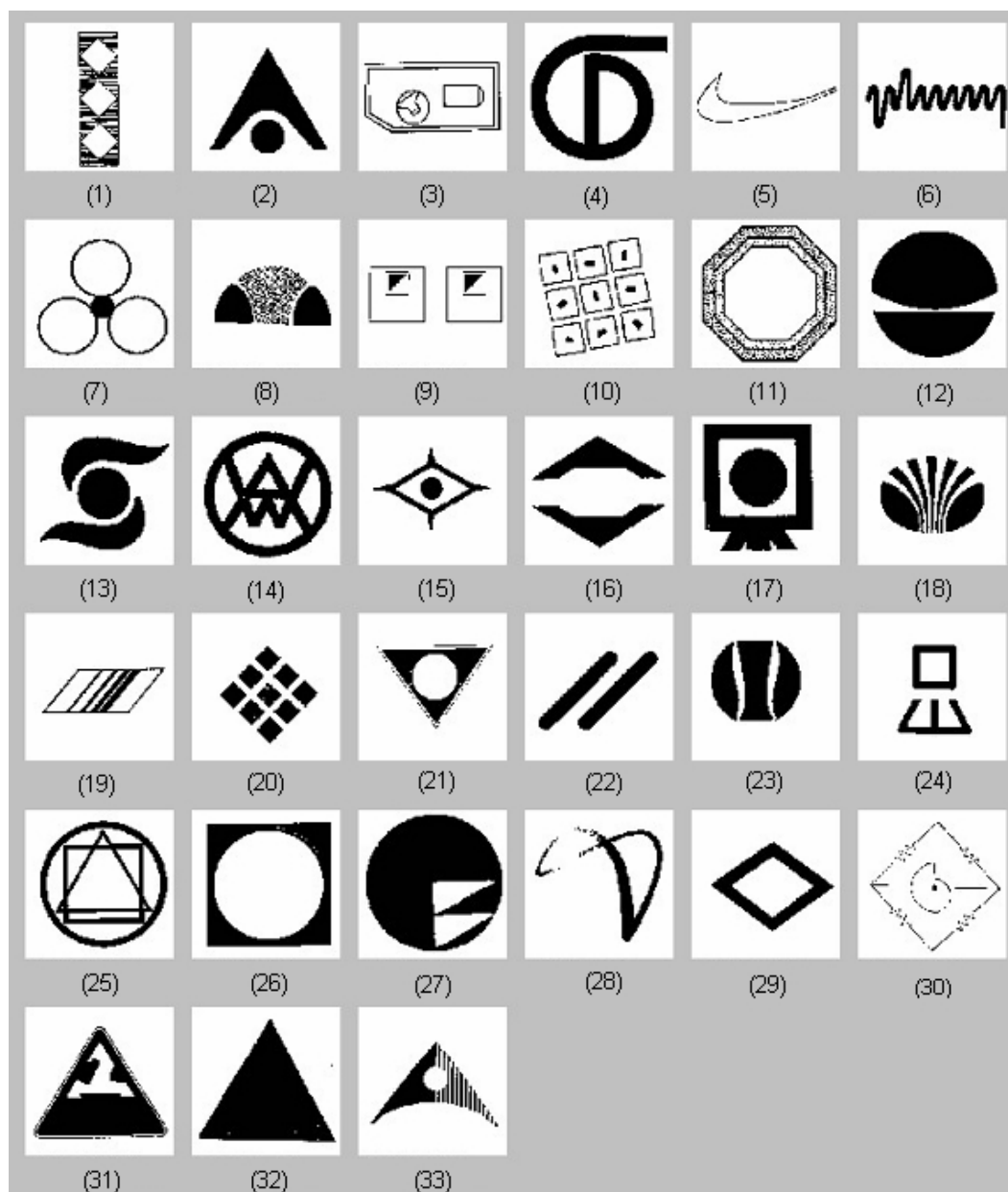
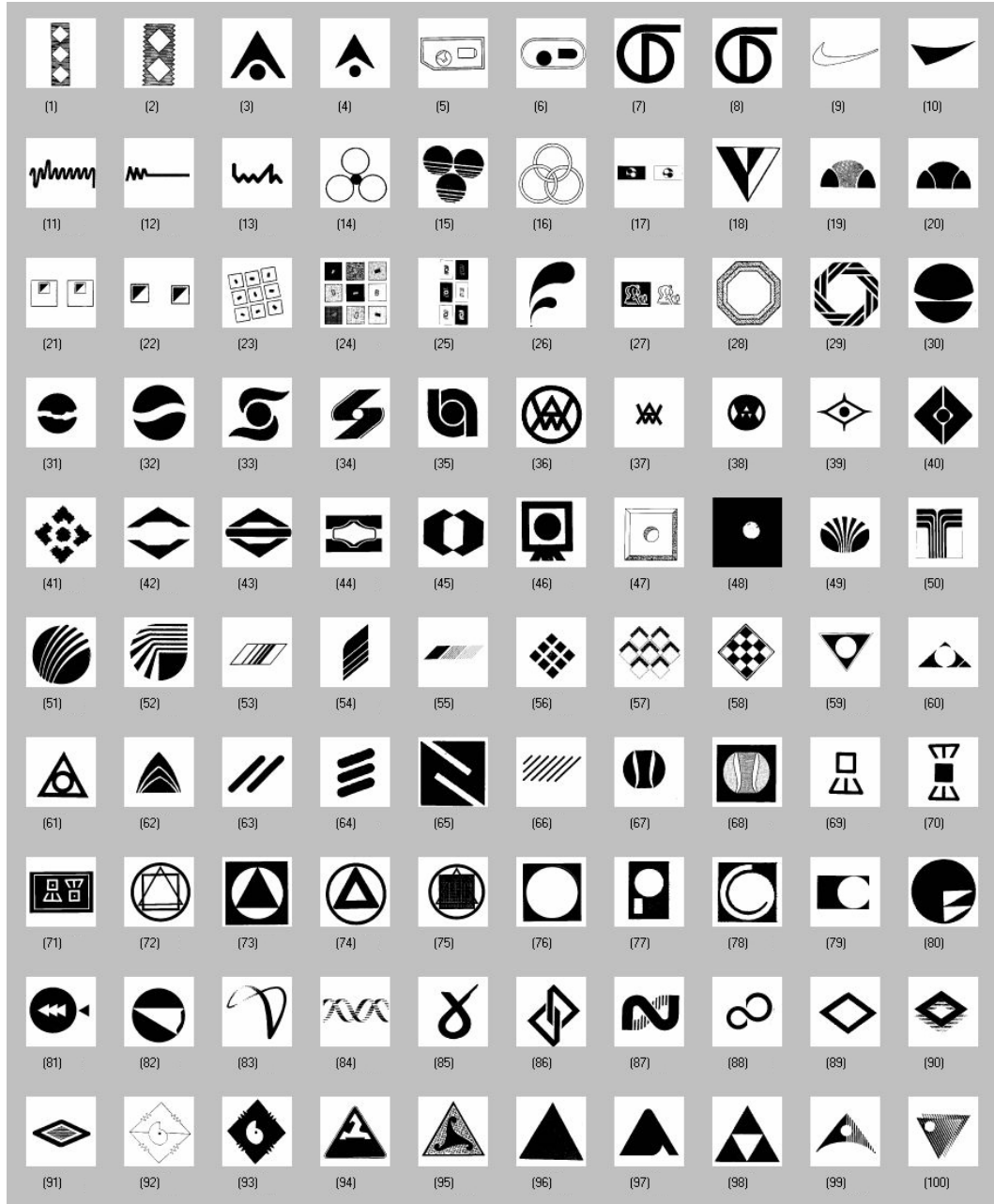**Figure 6.7 Query images used in the experiments**

Figure 6.8 Trademark database images used in the experiments

6.3.2 Proposed meta-features

We formulated a set of 27 meta-features, summarised in section 6.1 and shown in Table 6.4, from which we wish to find a subset of candidate meta-features in the first experiment.

Table 6.4 Meta-features evaluated in the experiment.

Number	Name
	Proposed Meta-Features
1	Summation of scale differences between pairs of matched keypoints
2	Summation of orientation differences between of pairs of matched keypoints
3	Moment ($m00$) from matched keypoints
4	Moment ($m01$) from matched keypoints
5	Moment ($m02$) from matched keypoints
6	Moment ($m03$) from matched keypoints
7	Moment ($m10$) from matched keypoints
8	Moment ($m11$) from matched keypoints
9	Moment ($m20$) from matched keypoints
10	Moment ($m22$) from matched keypoints
11	Moment ($m30$) from matched keypoints
12	Moment ($m33$) from matched keypoints
13	SD of residual spatial mismatch (error) between query and test images post alignment via an affine transformation
14	Similarity score of matched keypoints
15	The total number of matched keypoints
16	Mean of scale differences between pairs of matched keypoint sets
17	Median of scale differences between pairs of matched keypoints
18	RMS of scale differences between of pairs of matched keypoints
19	Mean of orientation differences between pairs of matched keypoints
20	Median of orientation differences between pairs of matched keypoints
21	RMS of orientation differences between pairs of matched keypoints
22	Percentage of matched keypoints/ total keypoints of query image
23	Percentage of matched keypoints/ total keypoints of database image
24	Self similarity
25	Horizontal symmetry
26	Vertical symmetry
27	The average of multi-peak GHT scores

The above meta-features are described in further detail in Appendix A. We explain the experimental processes in the next section.

6.3.3 Experimental processes

In this experiment, the system generated sets of 27 meta-features from 3300 data sets (33 query images with 100 database images). Thereafter, the eigenvectors and eigenvalues of the meta-features were computed by PCA (Machado and Marinho, 2003). We sort the eigenvectors by their eigenvalues and calculate the cumulative eigenvalues. Then, we select those candidate meta-features that have a significant cumulative eigenvalue.

Next, we select each meta-feature subset by their factor loadings. We summarise the processes of the experiment in Figure 6.9.

The experimental protocol
Step 1: Compute27 meta-features from each query image and the database images (details in section 6.1).
Step 2: Compute the eigenvectors and eigenvalues of the meta-features by PCA
Step 3: Calculate the cumulative percentage of the eigenvalues.
Step 4: Select the number of major meta-features from the cumulative percentage of the eigenvalues.
Step 5: Group the meta-features to major meta-feature subsets by their factor loadings.
Step 6: Select the candidate major meta-feature set.

Figure 6.9 The experimental protocol

The experimental results are summarized in the next section.

6.4 Experimental results

In the experiment, we would like to find the number of meta-features to use in the system. We use PCA to assist our decision. The eigenvalues can be used to measure the number of meta-features to retain. We calculate eigenvalues and their cumulative representation by PCA as shown in Table 6.5 in order to determine the number of viable meta-features.

Table 6.5 Eigenvalues of proposed meta-features and their cumulative percentage

Meta-feature no.	Eigenvalues	Cumulative percentage
1	1.23E+01	45.40681
2	5.17E+00	64.54631
3	2.54E+00	73.94227
4	1.74E+00	80.38586
5	1.48E+00	85.85359
6	1.08E+00	89.84219
7	9.67E-01	93.42416
8	7.56E-01	96.22479
9	4.58E-01	97.92181
10	2.62E-01	98.89199
11	1.19E-01	99.33422
12	8.66E-02	99.6548
13	5.13E-02	99.84479
14	1.09E-02	99.88498
15	8.98E-03	99.91823
16	8.34E-03	99.94913
17	5.99E-03	99.97132
18	2.05E-03	99.97891
19	1.89E-03	99.98591
20	1.60E-03	99.99183
21	8.63E-04	99.99503
22	7.18E-04	99.99769
23	5.74E-04	99.99981
24	4.31E-05	99.99997
25	4.85E-06	99.99999
26	1.40E-06	100
27	6.97E-07	100

From the cumulative percentages, we select the number of significant meta-features to be 17, because the cumulative percentage achieved at this meta-feature's eigenvalue ranking is almost 100 percent and the cumulative percentage

of the next eigenvalue contributes less than 0.01%. Therefore, we use 17 meta-feature subsets for further factor analysis. We rank each meta-feature subset by its factor loading and show an example of the top three factor loadings for the 17 meta-feature subsets in Table 6.6.

Table 6.6 Top three highest eigenvectors of 17 meta-feature subsets

Meta-feature order subset	1		2		3	
	Meta-feature no.	Factor loading	Meta-feature no.	Factor loading	Meta-feature no.	Factor loading
1	8	0.2851	7	0.285	4	0.2849
2	21	0.3896	19	0.387	20	0.3856
3	17	0.5092	16	0.5087	18	0.5063
4	23	0.5722	22	0.5052	25	0.3941
5	26	0.5769	25	0.5611	22	0.3691
6	13	0.8475	24	0.4223	2	0.2524
7	24	0.8673	13	0.4034	25	0.1694
8	27	0.6017	14	0.535	23	0.2582
9	25	0.7042	26	0.6994	24	0.0917
10	2	0.8458	13	0.3137	20	0.2561
11	23	0.6974	22	0.697	26	0.1036
12	14	0.7251	27	0.6714	23	0.0793
13	12	0.5399	15	0.4364	1	0.4358
14	6	0.5641	11	0.5054	5	0.4056
15	17	0.5745	18	0.5123	20	0.4704
16	21	0.5533	20	0.5354	17	0.449
17	15	0.4705	12	0.4382	7	0.3984

We found that some meta-feature subsets had similar values, for example, meta-feature subset 1 had nearly the same value of the first and second factor loadings corresponding to 0.2851 and 0.285 respectively. Since this result does not reveal candidate major meta-features unambiguously, we rotated the eigenvectors by the Varimax method (Field, 2000) with a 17 meta-feature subset.

The Varimax rotation moves the eigenvectors to maximize the variance within each eigenvector. This procedure has the effect of reducing the number of eigenvectors with small eignevalues and increasing the number of eigenvectors with large eigenvalues (Field, 2000). An example of the results produced by rotating eigenvectors is shown in Table 6.7.

Table 6.7 Top three results by rotating eigenvectors of 17 meta-feature subsets

Meta-feature order	1		2		3	
Component	Meta-feature no.	Factor loading	Meta-feature no.	Factor loading	Meta-feature no.	Factor loading
1	7	0.5063	3	0.4616	9	0.3939
2	21	0.8727	19	0.4875	16	0.0183
3	18	0.872	16	0.4843	15	0.0451
4	23	0.9998	15	0.0114	1	0.0112
5	26	1	11	0.0045	6	0.0025
6	13	1	19	0.0035	21	0.0026
7	24	1	8	0.001	6	0.001
8	27	1	15	0.0037	1	0.0035
9	25	1	8	0.0015	3	0.0013
10	2	0.9997	19	0.0161	21	0.0118
11	22	0.9997	11	0.0127	6	0.0094
12	14	0.9999	15	0.0102	1	0.0101
13	12	0.7245	10	0.4246	11	0.4164
14	6	0.6002	5	0.4684	11	0.4615
15	17	0.9301	16	0.3195	18	0.1736
16	20	0.9271	19	0.33	21	0.1742
17	15	0.7291	1	0.6729	4	0.0738

By rotating eigenvectors, we can group 27 meta-features to 17 meta-feature subsets by selecting high factor loadings in each meta-feature group. The resulting 17 meta-feature subsets is shown in Table 6.8.

Table 6.8 the candidate major meta-features for a 17 meta-feature groups

Meta-feature subset	Candidate major meta-feature(s)
1	3, 4, 7, 8 and 9
2	19 and 21
3	16 and 18
4	23
5	26
6	13
7	24
8	27
9	25
10	2
11	22
12	14
13	10 and 12
14	5, 6 and 11
15	17
16	20
17	1 and 15

Meta-feature subsets 1, 13, and 14 all comprise moments. The meta-feature subset 1 comprises meta-features 3, 4, 7, 8, and 9 and corresponds to *m00*, *m01*, *m11*, and *m20* respectively. Because high order moments tend to be unstable and are sensitive to noise (Kotoulas and Andreadis, 2005; Sluzek, 2005), Chaumette (2004) suggest using the lowest order moments possible. For that reason, the meta-feature 3 (*m00*) is selected to represent the spatial moments meta-features. The same decision criteria are used to select representative meta-features in subsets 13 and 14 because they all consist of image moments. Meta-feature subset 13 consists of meta-features 10 and 12 which comprise *m22 m33* respectively. Therefore, meta-feature 10 (*m22*) is chosen to represent meta-feature subset 13. Meta-feature subset 14 consists of meta-features 5, 6, and 11 which comprise m02, *m03*, and *m30* respectively. Therefore, the meta-feature 5 (*m02*) is selected to represent meta-feature subset 14.

Meta-feature set 2 consists of meta-features 19 and 21 which code mean orientation differences and RMS orientation differences between sets of matched keypoints respectively. Because we are interested in the dominant orientation difference between matched keypoints, meta-feature 19 is selected to represent this meta-feature subset.

Meta-feature set 3 consists of meta-features 16 and 18 comprising the mean scale differences and the RMS scale differences between matched keypoints. Because we are interested in the dominant degree of scale difference, meta-feature 16 is selected to represent this meta-feature subset.

Meta-feature set 17 consists of meta-features 1 and 15 that represent the total sum of scale differences and the total number of match points respectively. Because there are already meta-features representing scale, meta-feature 1 is taken to be representative for this meta-feature subset.

The remaining meta-feature set consist of only a single meta-feature, meta-feature set 4, meta-feature 23 (the ratio of matched points to total keypoints) and this meta-feature is used to represent its meta-feature subset accordingly.

As a result of the above anaylsis we were able to select a subset of 17 meta-features that we believed would best support computation of the following three Gestalt properties: similarity, proximity, and simplicity. In addition, the system also computes meta-features such as symmetry, self similarity, and moments to group similar image characteristics by their appearance properties. We investigate the ability of these meta-features to characterise appearance properties in next section.

6.5 Appearance properties of meta-features

A number of experiments were devised that generated appearance differences between compared images. We then applied our set of 17 meta-features to each of the sets of matched keypoints extracted from the compared images for each appearance property investigated. Details of each meta-feature subset used to quantify each appearance property are described below.

Global rotation: meta-features 1, 10, and 11 measure this appearance property because they capture the difference in orientation of features extracted

from two compared images.

Global pattern similarity: meta-features 2, 6, and 7 measure this appearance property because meta-feature 2 calculates area difference, meta-feature 6 calculates similarity between matched keypoint sets, and meta-feature 7 calculates the number of keypoints that match between compared images. Generally, the above meta-features measure pattern similarity between compared images.

Global pattern overlap: meta-features 12 and 13 measure this appearance property because these meta-features calculate the ratio of the matched points to total points for compared images.

Spread of matched keypoints: meta-features 3 and 4 measure this appearance property because they measure the spatial distribution of the matched points.

Structural configuration between matched keypoint sets, we used meta-feature 5 to measure this appearance property because it measures the distortion of two images from their standard deviation of matched points error residuals.

Scale difference between matched keypoint sets: meta-features 8 and 9 measure this appearance property because they measure the global scale difference of keypoints matched between compared images.

Self similarity: meta-feature 14 measures this appearance property because it measures self similarity directly in each compared images.

Symmetry: meta-features 15 and 16 measure this appearance property in the form of horizontal and vertical symmetries.

Component similarity, meta-feature 17 measures this appearance property by finding the average of multi-peak GHT scores, corresponding to major sub-component matches within the compared images.

The above feature grouping structures the 17 meta-features into 9 appearance properties comprising: global rotation, global pattern similarity, global pattern overlap, spread of matched keypoints, structural configuration of matched keypoints, scale difference between matched keypoints, self similarity, symmetry, and sub-component similarity. A summary of each meta-feature and

its appearance property is shown in Table 6.9.

Table 6.9 The meta-features to measure appearance properties in the system

Meta-feature	Appearance property
1. Summation of orientation differences of a pair of matched keypoint sets	Global rotation
2. Moment (m00) from matched keypoints	Global similarity
3. Moment (m02) from matched keypoints	Spread of the match points
4. Moment (m22) from matched keypoints	Spread of the match points
5. SD of error-distance for a pair of matched keypoints	Structural configuration of the match points
6. Similarity score for matched keypoints	Global similarity
7. Total number of matched keypoints	Global similarity
8. Mean of scale differences between a pair of matched keypoint sets	Scale difference of matched keypoint sets
9. Median of scale differences between a pair of matched keypoint sets	Scale different of matched keypoint sets
10. Mean of orientation differences for a pair of matched keypoint sets	Global rotation
11. Median of orientation difference for a pair of matched point sets	Global rotation
12. Percentage of matched points/ total keypoints in query	Global pattern overlap
13. Percentage of matched points/ total keypoints in the model	Global pattern overlap
14. Self similarity	Self similarity
15. Horizontal symmetry	Symmetry
16. Vertical symmetry	Symmetry
17. The average of multi-peak GHT scores	Sub-component similarity

In addition, the 9 appearance properties are related to the following 3 Gestalt properties: similarity, proximity, and simplicity. Similarity groups image parts that have similar configurations together. Proximity groups image parts that are close to each other together. Simplicity groups the same image sub-structures together. The relationship between Gestalt properties and appearance properties is

explained in the following and shown in Table 6.10.

First, we applied similarity to group similar parts in an image. Similarity can be size, orientation, pattern, shape, or value, as measured by the following appearance properties:

- **Global rotation**
- **Global pattern similarity**
- **Scale difference between matched keypoint sets**
- **Self similarity**
- **Sub-component similarity**

From the above appearance properties, we can measure multiple types of similarity.

Second, we employed proximity to group the connected area or closeness components in each image. Proximity can be nearness, touch, overlap, or combine, as measured by the following appearance properties:

- **Global pattern overlap**
- S**pread of matched keypoints**
- **Structural configuration between matched keypoint sets**

From the above appearance properties, we can measure component density and proximity.

Third, we exploited simplicity to group multiple sub-parts into components defined by simplicity of form. Simplicity can be represented by symmetry, regularity, or smoothness, as measured by the following appearance property:

- **Symmetry**

Therefore, we measure symmetry simplicity.

We summarise the mapping between appearance properties and Gestalt properties in Table 6.10.

We explain the experimental framework for validating perceptual grouping in the system according to appearance properties and Gestalt properties in Appendix B.

Table 6.10 The relationship between Gestalt properties and appearance properties

Gestalt number	Gestalt property	Gestalt property	Meta-feature(s)
1	similarity	Global rotation	1, 10 and 11
1	similarity	Global pattern similarity	2, 6 and 7
1	similarity	Scale difference between matched keypoint sets	8 and 9
1	similarity	Self similarity	14
1	similarity	Sub-component similarity	17
2	simplicity	Symmetry	15 and 16
3	proximity	Global pattern overlap	12 and 13
3	proximity	Spread of matched keypoints	3 and 4
3	proximity	Structural configuration between matched keypoint sets	5

6.6 Discussion and Conclusion

We have proposed meta-features based on interpreting the spatial configuration of matching keypoints to provide a perceptual grouping mechanism within our trademark retrieval system. In this chapter, we selected the number of dominant meta-features by means of PCA. The result suggests that only 17 meta-features are significant out of the 27 meta-features we computed originally, because we implemented measures for several very similar visual appearance characteristics which results in their outputs being correlated when exposed to the same visual stimuli (details in section 6.1). Then, we grouped 27 meta-features into 17 meta-feature subsets, using FA to prune the redundant meta-features, and selected these for our subsequent investigations.

The 17 major meta-features are then grouped using correlation analysis to characterise 9 appearance properties comprising: global rotation, global pattern similarity, global pattern overlap, spatial spread of the matched keypoints, structural configuration of matched keypoints, scale difference of matched keypoints, self similarity, symmetry, and sub-component similarity.

Our meta-features are broadly related to three of the Gestalt properties comprising similarity, proximity, and simplicity. Therefore, the above meta-features provide the representation to endow our system with the ability to determine different aspects of similarity between compared images, such as how the components are rotated or spread out along an axis or indeed if the compared images contain conspicuous symmetry in the configuration of their components.

Our meta-features have been designed to provide significant information from low-level features to assist retrieval performance (Eklund and Goebel, 2006). Obvious limitations of our meta-features are that they are much less sophisticated than those found in human vision and also that our meta-features only perform global grouping, or interpretation, of local keypoints. While human vision appears to be capable of hierarchical grouping in scale over the visual field, we can to a degree justify our simple global grouping approach based on the following observation: trademark images are an example of an image class where attention over the whole image is important since every pattern group present in the trademark is likely to be significant. Therefore, a simple global interpretation of the appearance properties of the trademark may be sufficient to achieve improved database search performance over keypoint similarity alone as offered by the standard SIFT algorithm. Human judgment plays an important role in the specification of semantic content in creative images (Enser et al., 2003) and this judgement is related to the expression of high-level concepts (Liu et al., 2007).

Live trademark retrieval systems in patent offices are used by humans to judge similar trademark images. However, few researchers have used user feedback to retrieve similar trademark images and as a consequence many systems are unable to reflect user consideration. Therefore, we propose a system that applies relevance feedback to retrieve trademark images based on classifying relevant images by means of our dominant meta-features. The key concept is that the system can learn which, if any, of the meta-features is diagnostic of the desried trademark images, through learning the type of images which are being labelled as relevant by the user. When a new database search is initiated following relevance feedback, those images meeting the learned feedback criteria will be ranked highly for presenttaion to the user, as described in the next chapter.

Chapter 7

Quantifying High-level Concepts from Point-based Grouping of Local Features

In Chapter 6, we proposed a set of meta-features as a way to imitate perceptual grouping and selected 17 meta-features to use in the system. In this chapter, we investigate the meta-features according to high-level concepts. Our approach to deriving high-level concepts from meta-features is explained. In the following sections, we motivate our adoption of high-level concepts, propose the high-level concepts approach, investigate validation of this approach, and finally draw conclusions.

7.1 Introduction

Gestalt grouping principles have been exploited in many applications, such as map reading, graph drawing, and homepage design (Paay & Kjeldskov, 2007; Hsiao & Chou, 2006). Perceptual grouping offers a stable basis for recognizing shapes, symbols, and domain objects (Saund and Mahoney, 2004). Trademark image retrieval systems that use perceptual grouping have been reported in the literature to having achieved good retrieval performance (details in chapter 2). Regarding the Gestalt laws of perceptual grouping, shape is very important in human visual judgement. Hence, perceptual grouping plays an important role in trademark image retrieval. Non-accidental properties give discrimination power to human shape similarity judgement (Gibson et al., 2007; Biederman, 2007). Both of these principles have motivated us to propose meta-features that capture the global configuration, i.e. grouping, of local features (details in chapters 4, 5, and 6).

Human judgment uses high-level concepts to measure image similarity (Liu et al., 2007, Zhou and Huang, 2000). However, the features utilised within CBIR systems are mainly low-level features (Liu et al., 2007). Therefore, a technique to derive high-level concepts from low-level features is required (Liu et

al., 2007, Zhou and Huang, 2000). Since we are interested in obtaining high-level concepts without keywords, we propose to base our system on machine learning and relevance feedback to derive high-level concepts (details in chapter 3).

In the next section, we explain our approach to deriving high-level concepts in our system.

7.2 Utilisation of High-level concepts by grouping local keypoint features

Relevance feedback is a special technique employed in on-line information retrieval systems. Relevance feedback utilizes user feedback that scores whether prior retrieval results are relevant, or irrelevant, in order to inform (i.e. optimize) a new database search cycle. There are three types of feedback: explicit feedback, implicit feedback, and blind feedback (Hopfgartner and Jose, 2007; Jordan, 2005).

When the user directly indicates relevance judgments to a database retrieval system this is termed **explicit feedback**. The user may indicate by two values (binary relevance) or multiple values (graded relevance) (Kekalainen, 2005). Binary relevance feedback indicates either a relevant or irrelevant retrieval result. Graded relevance feedback is typically quantised on a scale such as not relevant, a little relevant, relevant, or very relevant.

Implicit feedback comprises indirect feedback from the user that can be inferred from user conduct, for example, eye movement for viewing, or viewing time, page browsing, or scrolling actions (Hopfgartner and Jose, 2007; Buscher et al., 2008).

Blind, or pseudo relevant, feedback does not require user feedback. It simply assumes that the first k documents (top-k) in a ranked result set are relevant.

Normalized discounted cumulative gain is a performance metric which became popular ~2005 to measure the usefulness of ranking algorithms based on explicit relevance feedback. Other measures include precision at the *kth* item returned from a query and also the mean average precision (Agichtein et al.,

2008). Normalized precision and recall are used to measure performance of trademark image retrieval systems (Eakins et al., 1999; Jiang et al., 2006) and therefore this measurement appears to be well suited to our context the research reported here.

In our system, we focus on explicit feedback: each image in a list of top-k images, returned in response to an input query image, is labelled by the user as relevant or irrelevant. At any point during a query session, the user can select to continue or stop the cycle of marking relevant images and retrieving results.

In this research, we approach relevance feedback as a two-class (relevant/irrelevant) classification problem, as first suggested by van Rijsbergen (van Rijsbergen, 1979). When a query is initiated, the query image is automatically compared to all of the images in the database and the top-k images are then classified (i.e. manually labelled) by the user as relevant or irrelevant. We then apply the learning processes to this training data and the system returns a set of retrieved images to the user for the next cycle of database search.

Generally, for each cycle, the user selects a small set of images to train the system and the system then uses meta-features derived from these selected relevant images for learning. Normally, the number of training examples is small, comprising less than 20 images per cycle of interaction, due to the user's limited patience and willingness to cooperate (Zhou and Huang, 2003). For such small sample sizes, some standard learning algorithms, such as the support vector machine (SVM), are not stable enough to give viable classification performance and require more training samples from the user (Zhou and Huang, 2003).

We employ a non-parametric classifier, ID3 (Iterative Dichotomiser 3) developed by Ross Quinlan in 1983, to endow our retrieval system with a machine learning and decision making mechanism. ID3 builds a decision tree from training data. In this research, we prepare a feature vector (comprising meta-features described in Chapter 6 and Appendix B) from relevance feedback to train ID3. Decision trees are then used for the purpose of decision making by information gain when ranking subsequent database matches.

ID3 is appropriate in the context of this work since it is both robust and

also available in source code form. Since ID3 generates human-readable decision trees, the classifiers it generates are often comparatively easy to understand and therefore easy to analyse and debug. Similarly, the complexity of the classifiers generated can be readily estimated from the number of decisions required to achieve a classification, indicating the likely utility of the classification meta-features employed.

There is an issue in decision tree learning when training data contains noise or when the number of training data is too small to represent at sample of the desired result. In both cases, ID3 can generate trees that overfit the training data and thereby decrease the system performance. To solve overfitting, Mitchel reported that rule post-pruning is a practical approach to solve the overfiting problem (Mitchel, 1997). This technique checks decision tree growth to give best performance. When the performance begins to decrease, the tree needs to be pruned. By comparing tree classification performance of the previous training iteration and the current training iteration, we can measure when tree performance is converging. The tree growth is terminated if the performance of ID3 decreases. The tree is converted to an equivalent set of rules with one rule for each path from the root to a leaf node. In pruning, some rules are removed to increase performance.

We have used our retrieval system in conjunction with relevance feedback and ID3 to evaluate the efficiency of our set of meta-features. The overall process to retrieve similar images according to high-level concepts by grouping keypoint locations into meta-features is shown in Figure 7.1 and can be described as follows: We extract key points and descriptors from each image by means of the SIFT algorithm as described in section 5.3.1.2 of Chapter 5 and then calculate the matching scores of each image as described in section 5.3.1.2 of Chapter 5. We next extract local image features, i.e. keypoints, using the SIFT and consistent keypoints locations by means of the GHT accumulator as described in section 6.1 and then calculate meta-feature matching scores by computing the dot product between candidate significant meta-features extracted from the query image and each database image as explained in Figure 7.2.
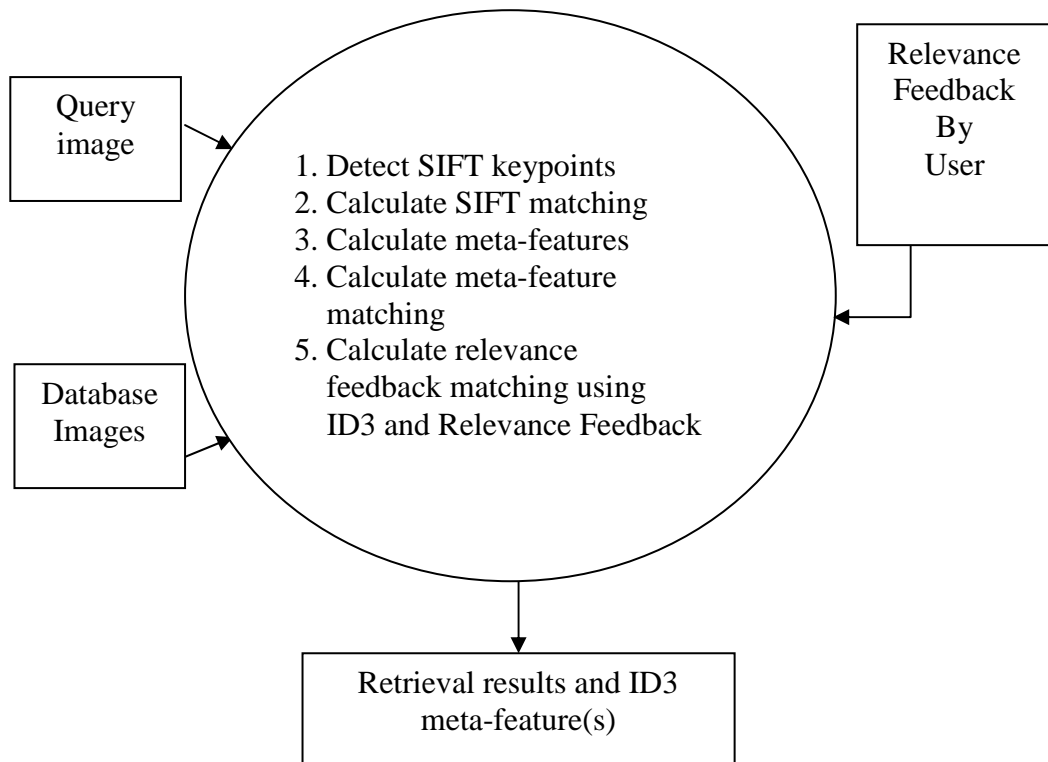
Figure 7.1 The overall process of the experiment

We can now compute the similarity score by ID3 and also the meta-feature vector. We start by sorting the results from meta-feature matching. After that, we input relevance feedback for each image in each display page (the top-k is 9) until the first similar image is found, the query itself is indicated to relevant image. Relevance feedback is achieved by indicating images similar to the query, and all other images as non-relevant (Giacinto and Roli, 2005). An example of user feedback is shown in Figure 7.3. The training meta-feature set is created by extracting meta-feature vectors from relevant and non-relevant images. An example of a training meta-feature set resulting from user feedback in Figure 7.3 is shown in Table 7.1. The relevance feedback is divided into 2 classes, where 0 codes for dissimilar and 1 codes for similar images respectively.

Meta-feature matching
Step 1: Normalize the meta-feature vector extracted from each of the database images and the query meta-feature vector to F_{Norm}. $\quad F_{Norm} = F \, / \, |F| \qquad\qquad\qquad\qquad\qquad$ (Equation 6.15) $\quad |F| = \sqrt{F_1^2 + F_2^2 + ... + F_n^2}$ Step 2: Matching query meta-feature vector (Q) with the normalized meta-feature vector (F_{Norm}) by taking the dot product between Q and F_{Norm}, i.e. cosine distance $D_{cos}(Q, F_{Norm})$. $\quad S_{feature\_vectors} = D_{cos}(Q, F_{Norm}) = Q'. F_{Norm} \qquad$ (Equation 6.16) $\qquad\qquad = (Q_1 \times F_{norm1}) + (Q_2 \times F_{norm2}) + ... + (Q_n \times F_{norm\,n})$

Figure 7.2 Overview of meta-feature matching algorithm
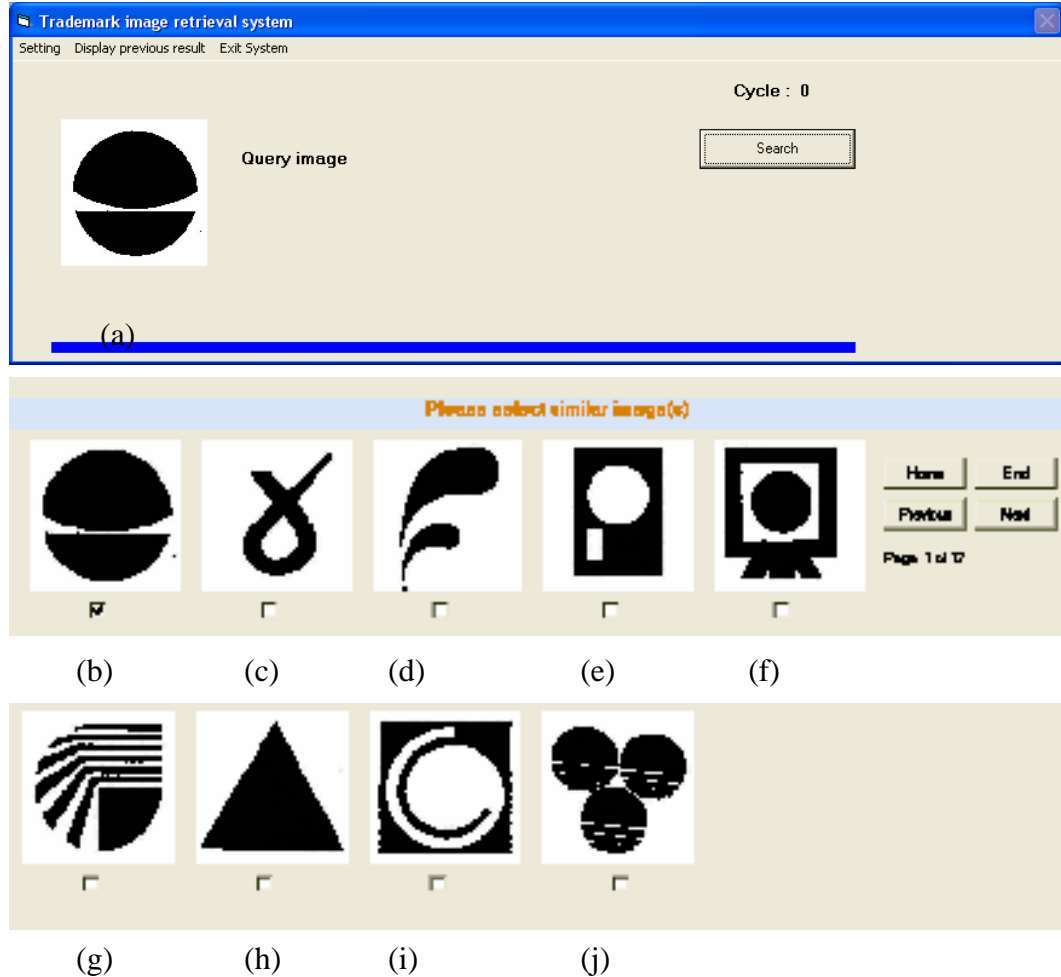
Figure 7.3 Input of relevance feedback from a user where (a) is the query image, (b) is the relevant image, and (c) – (j) are non-relevant images.

Table 7.1 Example of training meta-feature set from the relevant and irrelevant images in Figure 7.3

Image Feature	a	b	…	j
1	0.00000	0.25750	…	0.04299
2	0.69231	0.07692	…	0.15385
…	…	…	…	…
17	0.94585	0.26568	…	0.00000
Relevance feedback	1	1	…	0

The system then generates an ID3 tree by splitting nodes which have high information gain in terms of class examples. The algorithm for ID3 tree building is shown in Figure 7.4. From the ID3 tree, we create decision rules in the form of if-then-else clauses. An example of decision rules is shown in Figure 7.5. Next, we classify the relevance of each database image (*Class, Class* is 0 if ID3 classifies the database image as being non-relevant otherwise, *Class* is 1 if the database image is classified as being relevant), for example, *Class* in Figure 7.5 is 0 if the leaf node of the decision path is No, otherwise *Class* is 1 if the leaf node is Yes. In order to be able to rank images that are classified as being relevant, similarity scores (S_{ID3}) are computed (Equation 6.17) for each cycle of querying with relevance feedback.

$$S_{ID3} = (S_{node1} + S_{node2} + S_{nodek})/n \qquad \text{(Equation \quad 7.1)}$$

where *n* is the number of nodes in decision path in ID3 that evaluates the relevant class (Yes), for instance, *n* is 2 in Figure 6.14, and S_{nodei} is the node score that is calculated from the meta-features extracted from the database images for each node.

$$S_{nodei} = \sqrt{((F_{nodei} - Fs_i)/Fs_i)^2} \qquad \text{(Equation 7.2)}$$

where F_{nodei} is a meta-feature threshold value of the database image in each node.

Fs_i is the threshold value of each node from the meta-feature value of S_v in Equation 7.3 while in the process of splitting nodes.

ID3 tree building

Input training meta-feature set (*S*) with relevance feedback from user selection of displayed images.

Step 1: If all meta-feature values are the same then

Return a leaf node with the result from relevance feedback

Else

Find meta-feature (*F*) with highest information gain

Gain(S, A) the information gain of meta-feature set *S* on each meta-feature *A* is defined as

$$Gain(S, A) = Entropy(S) - \sum_{v=1}^{N} ((|S_v|/|S|) * Entropy(S_v))$$

Where (Equation 7.3)

N is the number of all possible values of attribute *A*,

S_v is the subset of *S* for which attribute *A* has value *v*,

$/S_v/$ is the number of elements in S_v,

$/S/$ is the number of elements in *S*.

$$Entropy(S) = \sum_{i=1}^{L} - p(class_i) \log_2 p(class_i)$$

(Equation 7.4)

$$= \sum_{i=1}^{L} - (|S_{class_i}|/|S|) \log_2 (|S_{clas_{si}}|/|S|)$$

where

p(class_i) is the proportion of *S* belonging to class *i*,

$/S_{class_i}/$ is the number of elements in *S* belonging to class *i*,

$/S/$ is the number of elements in *S*,

L is the number of class categories in *S*.

End

Figure 7.4 Overview of the ID3 tree building algorithm

ID3 tree building (continued)
Step 2: Assign the best meta-feature (F) in step 1 as a decision node
Step 3: For each value of F create a branch by partitioning the training set S into subsets $S1$, $S2$, ..., S_N according to the values of F.
Step 4: Follow each branch whereby the value of the branch is present If the meta-feature perfectly classifies the training sets then The process stops and outputs a decision tree. Else Perform step 1 recursively to each of the sets Si from step 3. End

Figure 7.4 Overview of the ID3 tree building algorithm (continued)

(a)   (b)
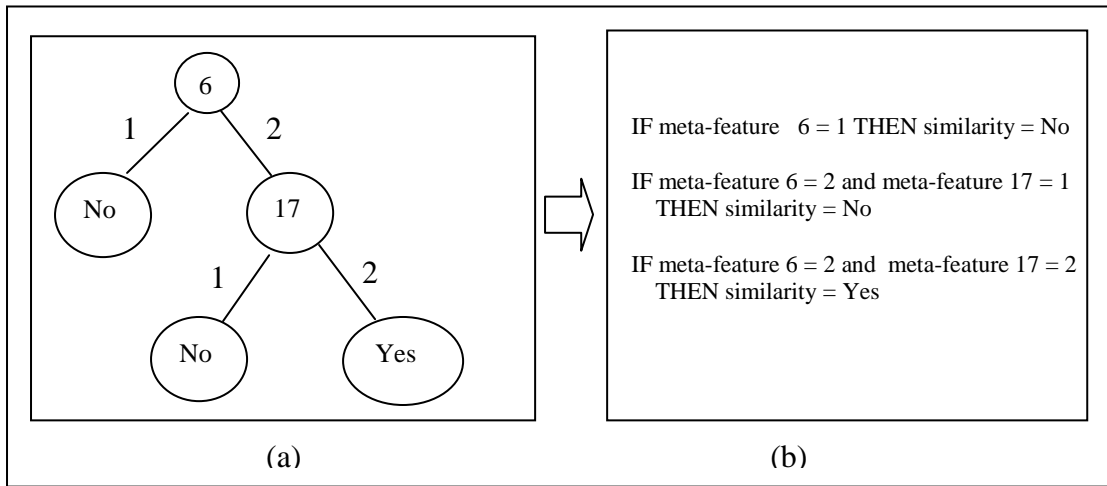
Figure 7.5 Example of decision rules in (b) that are generated from the ID3 tree in (a)

We now investigate high-level concepts of appearance properties in the system, and the benefit of using relevance feedback (i.e. feedback from the user to indicate relevant and non-relevant images) in order to make real decisions based on the high-level notion in the next section.

7.3 Investigation of the high-level concepts

We are interested in to what degree by using relevance feedback and ID3 our system is able to extract feedback meta-feature vectors that encapsulate Gestalt properties in order to retrieve similar trademark images. In other words, we now investigate how the retrieval results are influenced by classifying the relevance of

database image meta-features using ID3 trained by relevance feedback. We utilize the implementation described below.

Three experiments were devised in order to investigate the implementation of three Gestalt properties comprising: similarity, proximity, and simplicity validated by a specific test image set. Each set is designed to invoke only individual Gestalt properties in isolation, although this in fact is not always possible as mentioned in Chapter 6 and will be explained in more detail below. We select meta-features to represent each appearance property based on using PCA and Factor Analysis, and further analysis of each appearance property (Details in sections 6.2 to 6.5 of Chapter 6). The Gestalt properties and corresponding appearance properties investigated in each experiment are listed in the Table 7.2, and the meta-features are listed in Table 6.10.

Table 7.2 Gestalt properties investigated and proposed meta-features

Experiment number	Gestalt property	Appearance property	Proposed meta-features
1	Similarity	Global rotation	1, 10, and 11
5	Proximity	Structural configuration of the matched points	5
8	Simplicity	Symmetry	15

We investigated the effect of relevance feedback by measuring retrieval performance and examining the feedback meta-feature vectors generated when the system incorporates ID3 trained by relevance feedback. In this experiment "model" (ideal) feedback was provided to the system, and ID3 then made a binary decision regarding the relevance of each image in the test set. We then observed the retrieval result, i.e. which images of the test set were deemed to be relevant, and which meta-feature was selected to classify the appearance property present in the image training set. The effect of relevance feedback was determined by comparing the results of the system using ID3 provided by relevance feedback with the system results using meta-feature matching without relevance feedback.

We describe the above experimental procedure in detail in the following section.

7.4 Experimental procedure

In this section, we explain the experimental procedure for investigating the potential for high-level concepts to benefit the system retrieval performance. The objective of this set of three experiments is, given a set of meta-features that describe visual appearance properties, to determine which meta-features will be selected by relevance feedback based learning to discriminate the degree of each appearance property when presented to the system. We show the set of test images employed in section 7.4.1, describe experimental processes in section 7.4.2, and explain the result in section 7.4.3.

7.4.1 Test images

Each experiment used a different set of test images, in which each test image set changes the degree of each appearance property. Nominal "relevance feedback" is provided that labels a subset of images in the test set as being relevant and the remaining disjoint subset of images as being non-relevant. In all of these experiments, the first three images (including the query image) have been deemed by the above relevance feedback process to be similar, for example, the images (a) to (c) in Figure 7.6 have been defined to be similar images, and the remaining images defined to be dissimilar when compared with the query image in (a). The appearance threshold used to define similarity or dissimilarity for each subset of images in each test set is essentially *arbitrary*, since the objective is to determine if the system is able to group similar and dissimilar properties as defined by a user, i.e. arbitrarily.

Many researchers suggest that a small number of similar images should be used to support user convenience and cooperation (Zhou et al., 2006; Manning et al, 2008). Accordingly, the total number of images in each set is 9 to fit in one display area. The test sets are shown in Figures 7.6 to 7.8 for each experiment.

Experiment 1: We evaluate similarity appearance discrimination based on applying global rotation to the test image set.

The test set in this experiment is arranged by the degree of global rotation of the test images. The degrees of global rotation are 0, 10, 20, 30, 40, 50, 60, 70,

and 80 degree respectively. The test set in experiment 1 is shown in Figure 7.6.
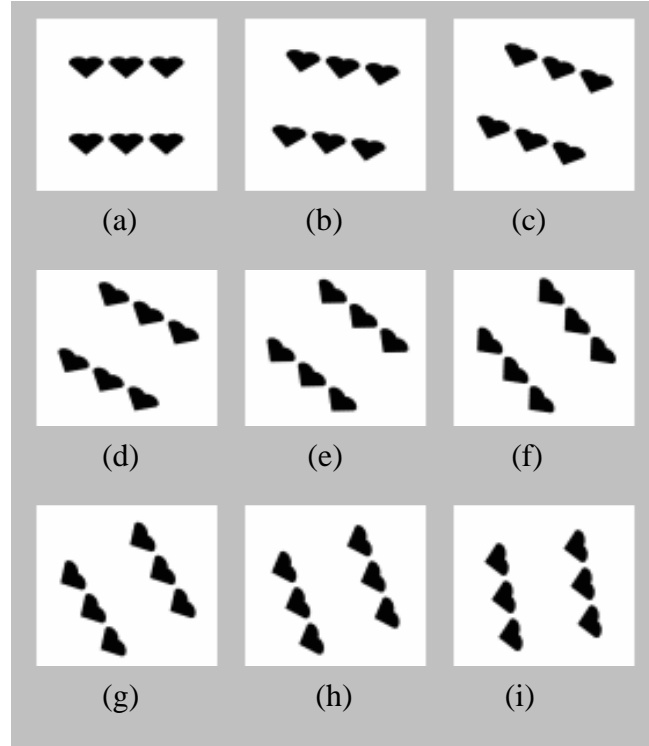
Figure 7.6 Test images in experiment 1 (Global rotation).
The relevant images are defined to be (a) – (c), the non-relevant images are defined to be (d)
– (i), and the query image is (a).

Experiment 2: We aim to evaluate appearance discrimination by varying proximity similarity and quantifying the structural configuration of matched keypoints.

The test set in experiment 2 is arranged by progressively varying the distance between two components in test images. The separation distances of the two components are 86.5, 76.5, 66.5, 56.5, 46.5, 36.5, 26.5, 16.5, and 6.5 percent respectively. The testing set in experiment 2 is shown in Figure 7.7.
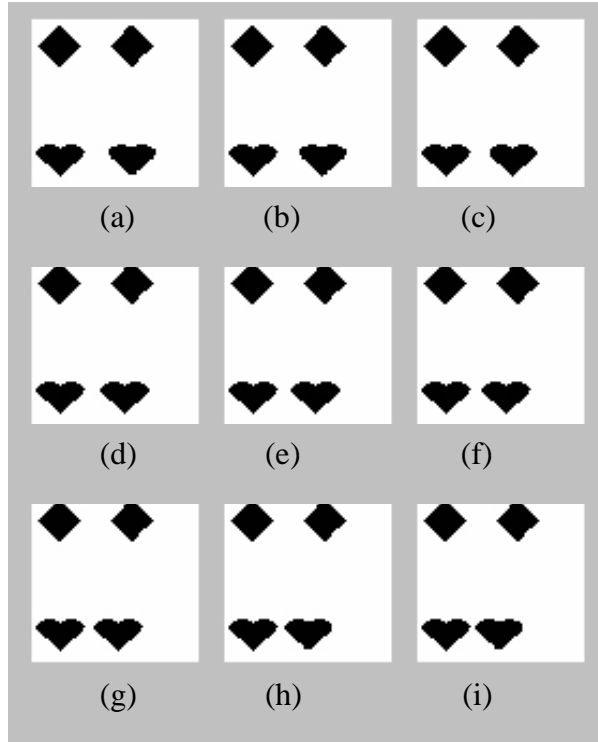
Figure 7.7 Test images in experiment 5 (Structural configuration of the matched points). The relevant images are defined to be (a) – (c), the non-relevant images are defined to be d) - (i), and the query image is (a).

Experiment 3: We aim to evaluate the simplicity property for appearance discrimination using the symmetry of matched keypoint positions.

The test set in experiment 3 varies the degree of global asymmetry for each of the test images, ordered by 0, 10, 20, 30, 40, 50, 60, 70, and 80 percent. The test set in experiment 3 is shown in Figures 7.8.
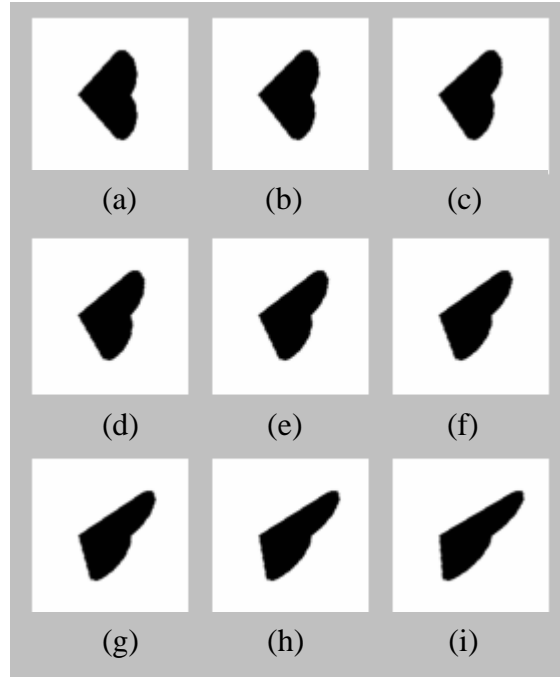
Figure 7.8 Test images in experiment 8 (Horizontal symmetry).
The relevant images are defined to be (a) – (c), the non-relevant images are defined to be (d) – (i), and the query image is (a).

In the next section, the experimental procedures for all three experiments will be described.

7.4.2 Experimental procedures

In each experiment, we use a different image set to investigate a single appearance property. We evaluate each appearance property with the system using the meta-feature matching with and without relevance feedback and ID3.

When relevance feedback is not used, the system bases its query decisions using meta-feature matching alone. A query and the test set images are input to the system, which extracts SIFT features from each image. The system matches and groups keypoints extracted from the query image and each test image using the GHT, which is explained in section 6.1 of chapter 6. The system calculates the 17 meta-features (details in chapter 6) to construct the meta-feature vector and then compute meta-feature matches. The overall process is shown in Figure 7.9.
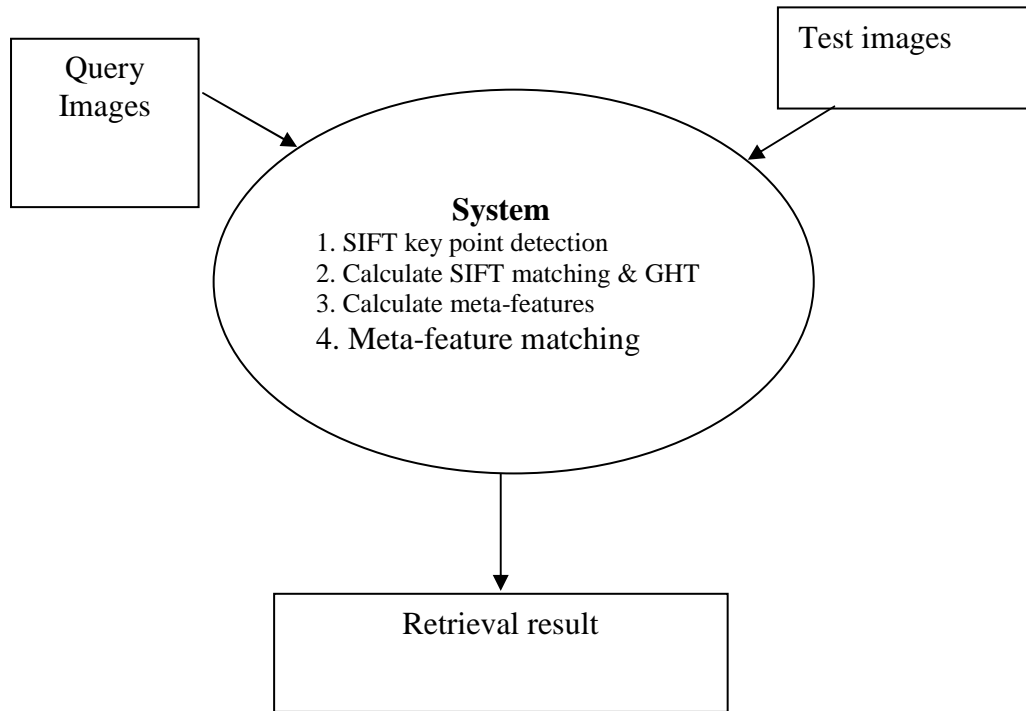
Figure 7.9 Processes in the system using meta-feature matching alone

When incorporating RF, the system extracts SIFT key point descriptors from each image, and then matches the query image and each test image using the GHT, as described in section 5.3.1.2. The 17 local meta-features are then calculated and an initial retrieval result by matching the meta-feature vector of the query to the meta-feature vectors of the test images. Then the system calculates a similarity score using relevance feedback and ID3 (details in section 7.2). An overview of the second sub-experiment is shown in Figure 7.1.

The processes can be summarised as:

1. Extract the keypoints and descriptors from each image using SIFT.

2. Calculate the SIFT matching score, as described in section 5.3.1.2.

3. Calculate the meta-feature vector of the keypoints for each image, as explained in section 6.1.

4. Calculate the meta-feature matching scores, as explained in section 7.2.

5. Compute the similarity score using ID3 and the meta-feature vectors, as described in section 7.2.

The system returns the retrieval result and feedback meta-feature vector

for each experiment. Thereafter, the retrieval result obtained by meta-feature matching only and ID3 with relevance feedback are compared to determine the utility of relevance feedback classification.

In addition, we evaluate the utility of the relevance classifiers generated by user feedback from the complexity of the trees induced byID3, in terms of tree depth and number of branches. Trees with fewer nodes indicate a simpler decision space and hence are more desirable than trees with greater numbers of nodes (Mitchell, 1997), indicating a complex decision space which is likely due to overfitting to the training data. Therefore, following the principle of Occam's Razor, the lower the complexity of the trees generated by ID3, the simpler the decision space potentially resulting in better performance of the relevance classifier.

7.5 Experimental results

In each of the experiments, the same class of (similar or dissimilar) images is retrieved through user feedback. Each experiment generates only one node for each ID3 tree, for example, the tree produced by ID3 for experiment 1 is shown in Figure 7.10.
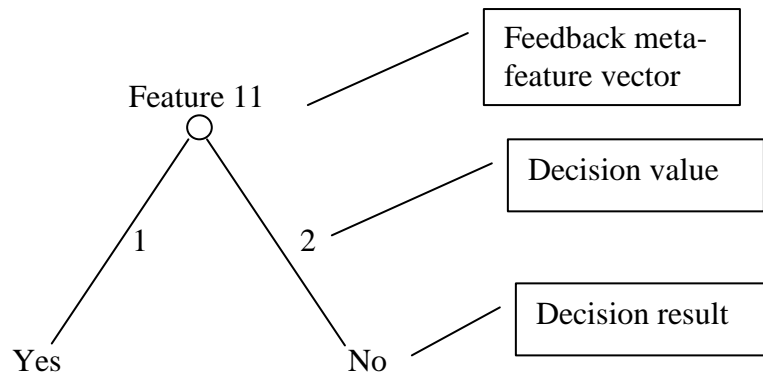
Figure 7.10 ID3 tree generated in Experiment 1. The root node represents the selected meta-feature, the branches are the decision values from ID3, and the leaves illustrate the decision result for each path.

A single ID3 node is generated in every experiment, so each appearance property decision is based on a single feedback meta-feature.

We summarize the feedback meta-feature vectors selected by ID3 and compare the feedback meta-feature vector with proposed meta-features for each Gestalt property in Table 7.3.

Table 7.3 Proposed meta-feature(s), and meta-feature selected by ID3 and Relevance Feedback in each experiment

Experiment number	Gestalt property	Proposed meta-feature(s) (Feature no)	RF meta-feature (Feature no)
1	Similarity	1, 10, 11	11
2	Proximity	5	5
3	Simplicity	15	15

According to Table 7.3, the set of feedback meta-feature vectors are contained in the set of proposed meta-features. Therefore, suitable meta-features in each experiment have been selected by ID3 to support decisions based on high-level visual appearance properties. This result suggests that if there are several meta-relations (keypoint configurations) operating simultaneously in an image, a decision tree could be generated by ID3 from several corresponding meta-features, or any single dominant meta-feature, where a single visual appearance property dominates. We investigate the stability of meta-feature selection by varying the number of similar images, and the result is summarized in Table 7.4.

Table 7.4 The feedback meta-feature in each experiment with different numbers of similar images

The number of similar images	Feedback meta-feature		
	Experiment 1	Experiment 2	Experiment 3
2	10	8	15
3	11	5	15

In experiment 3, the feedback meta-feature is the same for different numbers of similar images. The feedback meta-feature selected from experiment 1 and 2 varies when the number of similar images changes.

The feedback meta-feature results in experiment 1 code the same appearance property: similarity. This result suggests that the meta-features coding appearance property concepts can provide the system with the means to select the more dominant meta-feature in the same appearance property group.

The feedback meta-feature selected in experiment 2 code two different appearance properties: similarity and proximity. The test images were generated to vary proximity (structural configuration of the matched keypoints), and comprised similar components. Therefore, it is difficult to decouple the similarity and proximity properties.

Table 7.5 Example result s using Relevance Feedback, ID3, and meta-feature matching

Image	relevance feedback	ID3		Meta-feature matching	
		Result	Similarity value	Result	Similarity value
	Similar	Correct	1	Correct	3.443
	Similar	Correct	0.790	Incorrect	1.433
	Similar	Correct	0.586	Incorrect	1.705
	Dissimilar	Correct	0.394	Incorrect	1.810
	Dissimilar	Correct	0.156	Correct	0.990
	Dissimilar	Correct	0	Correct	0.995
	Dissimilar	Correct	-0.226	Incorrect	2.263
	Dissimilar	Correct	-0.413	Correct	1.128
	Dissimilar	Correct	-0.678	Correct	0.955

The retrieval results from the system using relevance feedback and ID3 were consistent with user feedback in every experiment, but the results using meta-feature matching were inconsistent in many experiments. This is because RF imposes user intentions that simple meta-feature matching (vector dot product comparisons between meta-features) has no means of deducing. ID3 is also able to select appropriate meta-features rather than using all the meta-features when matching, and therefore the system using relevance feedback and ID3 has the potential to achieve better results. An example of retrieval results obtained in experiment 1 is shown in Table 7.5. The system using relevance feedback and

ID3 returns correct results for all images, but when meta-feature matching alone we observe certain images being incorrectly retrieved as similar.

7.6 Conclusion

In response to the gaps in the research literature identified in Chapter 3, this research proposes the use of meta-features in conjunction with relevance feedback as a way to imitate aspects of human perceptual grouping. There is little reported work on utilizing perceptual grouping to retrieve similar images, and none reporting the calculation of meta-features from SIFT keypoints to measure Gestalt-based properties.

We evaluated the ability of the system to make trademark image similarity judgements using a single visual property in each experiment we conducted. Our results show that it was indeed possible to quantify basic visual appearance properties and select suitable meta-feature for measuring visual appearance properties based on user supplied relevance feedback.

However, due to the difficulty of decoupling visual appearance properties in the test images, more investigation is required to determine if our proposed approach is viable when more than one visual appearance property is active within compared images.

In conclusion, from the initial results reported in this chapter, it would appear that by endowing the system with relevance feedback training it becomes possible to construct a classifier that can base relevance decisions on suitable appearance properties by automatic selection of appropriate meta-features. Thereby we are able to retrieve similar images by categorising the meta-feature vector extracted from database images using a decision tree learned from training examples given by relevance feedback. This approach affords flexibility in the ability of the system to retrieve database images that are similar to a query image based in part on a user defined notion of similarity.

In the next chapter, we evaluate the system with real trademark images containing several visual appearance properties, in order to measure the system's retrieval performance when operating on real data.

Chapter 8

System Evaluation Experiments

In the previous chapter, high-level concepts captured by employing Gestalt-Based Perceptual Grouping (GBPG) of visual appearance properties were investigated within our trademark image retrieval system. The results showed that by incorporating high-level concepts wihtin our system it becomes possible to quantify each Gestalt property in order to retrieve similar images. In this chapter, we evaluate the system with a database of real trademark images, where each individual image may contain several Gestalt properties. The results illustrate the performance of the system when retrieving similar images that contain several Gestalt properties. We compare 3 methods for retrieving similar trademark images: In the first, we use SIFT matching which returns similar images by matching SIFT keypoint features. In the second method, we utilize several meta-features (meta-feature matching) to take into account the GBPG properties of images compared and retrieved by the system but without utilising relevance feedback to support the user definition of high-level concepts. In the third method, we extend the second system to employ relevance feedback (RF) in conjunction with machine learning to provide relevance feedback-based decisions by which similar images are selected (and ranked) from the database. We describe the experimental framework in section 8.1, implementation methods in section 8.2, experimental procedures in section 8.3, discuss the results in section 8.4, and summarise our findings in section 8.5.

8.1 Experimental Framework

We would like to evaluate the retrieval performance of our system supporting high-level concepts using relevance feedback and decision tree-based relevance classification (Relevance Feedback matching) and to compare the performance of this system configuration with system configurations that do not incorporate

---

relevance feedback (meta feature matching alone, and SIFT matching alone). This section presents the hypotheses, objectives, and research questions for this evaluation. Our hypotheses are:

It is possible to build an effective retrieval system for searching databases of trademark images by adopting image matching based on representing multiple GBPG properties. Relevance feedback based decision-making can select appropriate features based on GBPG properties, and can thereby increase the retrieval effectiveness of the system.

In our experiments, we compare the retrieval effectiveness of the system when using SIFT matching, feature matching, and RF matching.

8.2 Implementation Methods

In the first method, the system employs SIFT matching, and the system generates SIFT features. In the second method, the system utilizes meta-feature matching as the basis for extracting GBPG properties (details in chapter 6 and appendix B). In the third method, the system uses with RF matching and is supplied with user labelled relevant images (relevance feedback) during each database query cycle, and then selects GBPG properties (meta-features selected by ID3). This method attempts to represent high-level concepts in order to capture perceptual grouping by relevance feedback based on meta-features selected by ID3 decision trees.

The details of each of the above methods are explained in section 8.3.3. The system results using RF matching are compared with the other two methods by employing precision and recall measures. The results are shown in section 8.4.1. The advantages of using RF matching in trademark image retrieval is analysed in terms of the system retrieval effectiveness in section 8.4.3.

8.3 Experimental procedure

The experiments will measure system performance when retrieving similar images that contain multiple GBPG properties. We show the database images in section 8.3.1, list the meta-features in section 8.3.2, describe experimental processes in section 8.3.3, and explain the measurement method in section 8.3.4.

8.3.1 Image data

We evaluated our system using a database comprising 200 abstract images taken from the UK Trademarks Registry, and a set of 12 query images with known "correct" search results for this database. All the trademarks are binary images of 256x256 pixels. The image set comes from the same set used in the evaluation of the Artisan system (Eakins et al, 1997). The set of relevant images for each query was selected by UK trademark examiners and the query itself is contained in the relevant images (Giacinto and Roli, 2008). The 12 query images are shown in Figure 8.1, and the 200 database images in Figure 8.2.
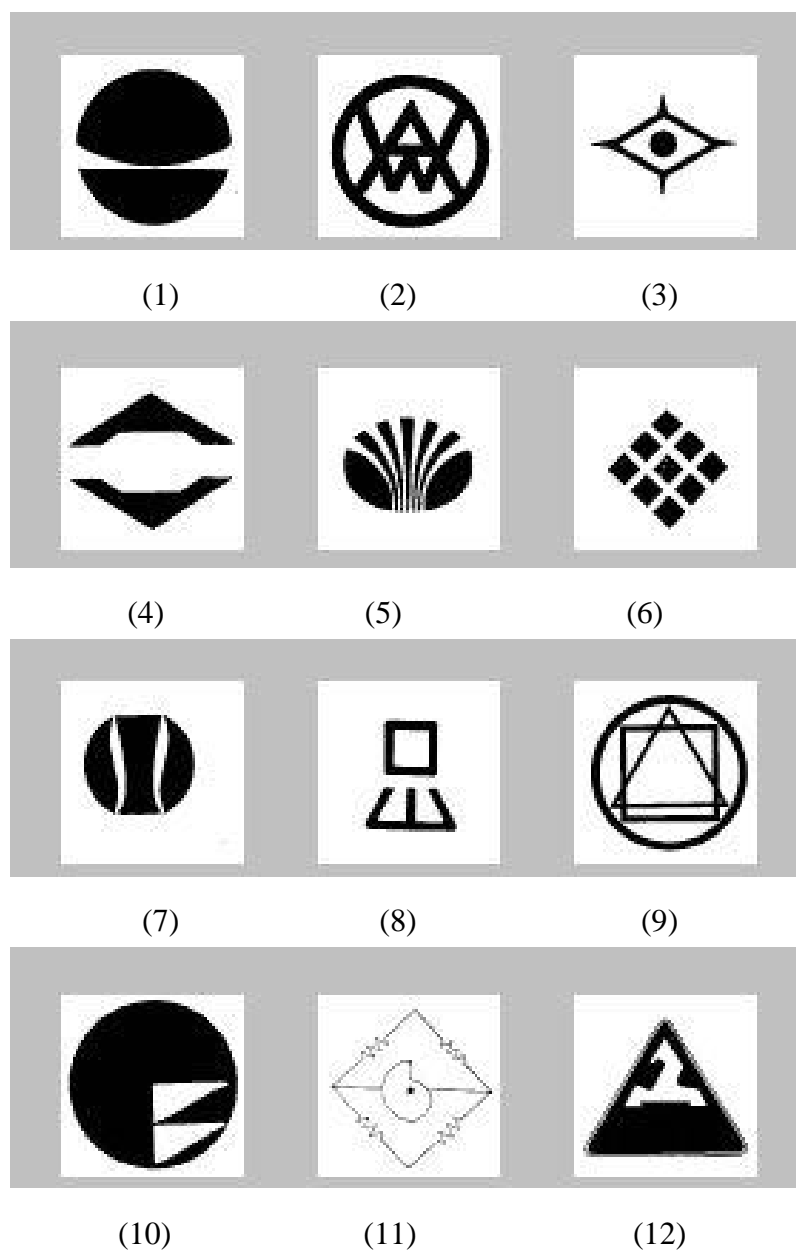
Figure 8.1 Query images used during the experiments
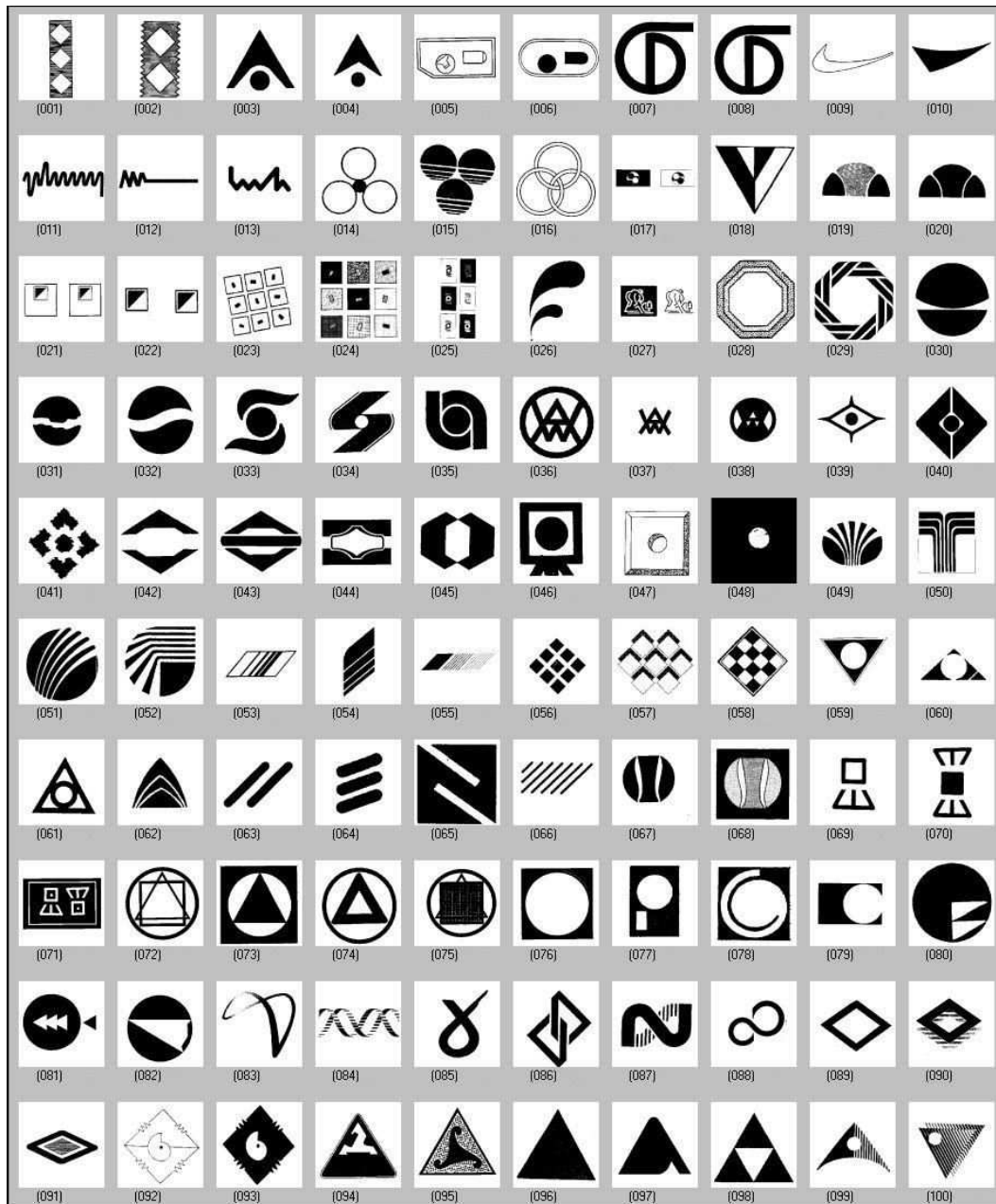
Figure 8.2 (a) Trademark image database used during the experiments
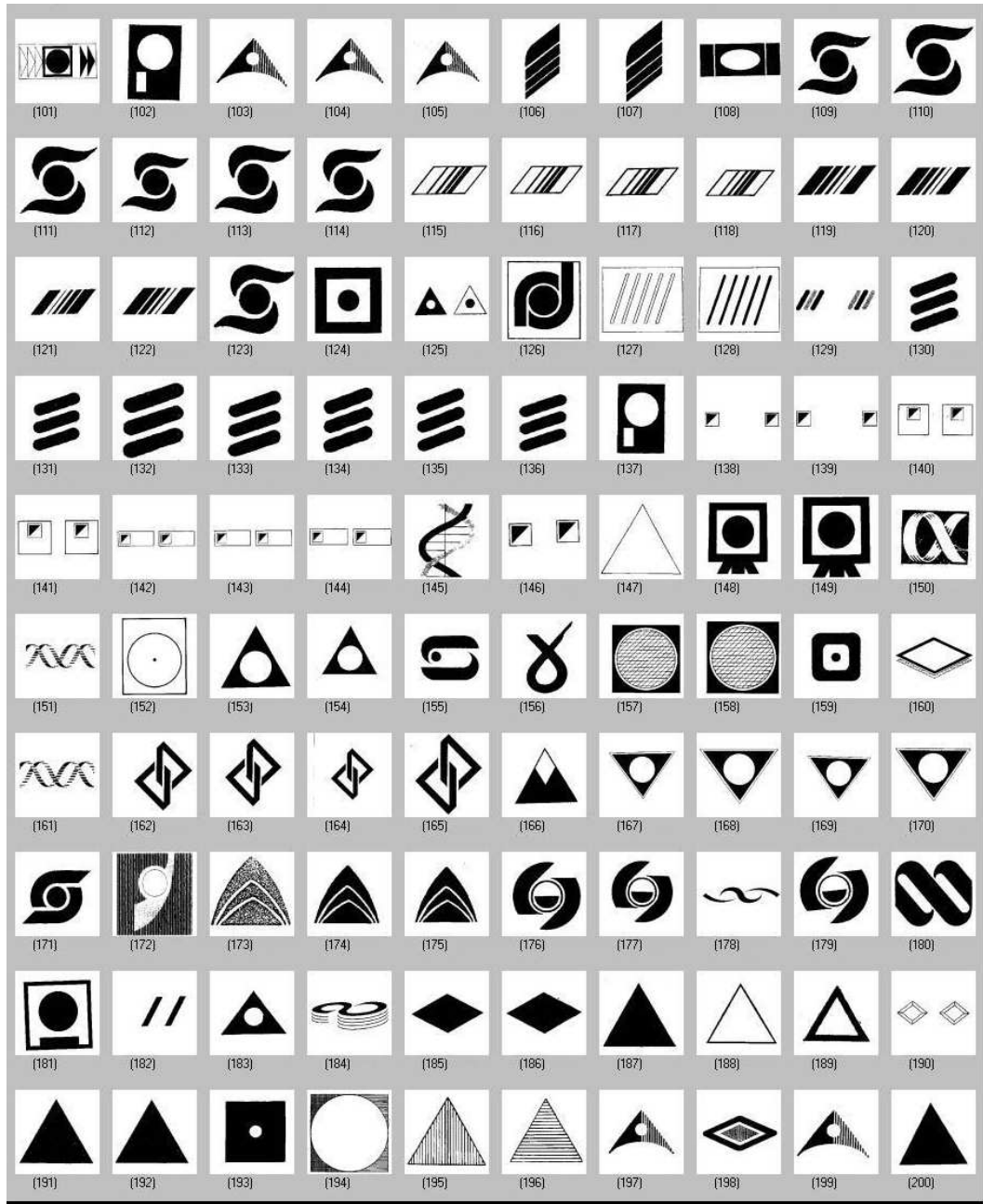
Figure 8.2 (b) Trademark image database used during the experiments (continued)

8.3.2 Meta-features

17 meta-features are utilized, which were described in chapter 6 and are listed here in Table 8.1. They are used in the meta-feature matching and RF matching methods for trademark image retrieval.

Table 8.1 17 meta-features extracted from each image

NO	Description of meta-feature
1	Summation of orientation differences of a pair of matched keypoint sets; this feature measures orientation similarity.
2	Moment (m00) from matched keypoints; this feature measures pattern similarity.
3	Moment (m02) from matched keypoints; this feature measures nearness proximity.
4	Moment (m22) from matched keypoints; this feature measures nearness proximity.
5	SD of error-distance for a pair of matched keypoints; this feature measures nearness proximity.
6	Similarity score for matched keypoints; this feature measures pattern similarity.
7	Total number of match keypoints; this feature measures pattern similarity.
8	Mean of scale differences between a pair of matched keypoint sets; this feature measures size similarity.
9	Median of scale differences between a pair of matched keypoint sets; this feature measures size similarity.
10	Mean of orientation differences for a pair of matched keypoint sets; this feature measures orientation similarity.
11	Median of orientation difference for a pair of matched point sets; this feature measures orientation similarity.
12	Percentage of matched points/ total keypoints in query; this feature measures overlap proximity.
13	Percentage of matched points/ total keypoints in the model; this feature measures overlap proximity.
14	Self similarity; this feature measures value similarity.
15	Horizontal symmetry; this feature measures symmetry simplicity.
16	Vertical symmetry; this feature measures symmetry simplicity.
17	The average of multi-peak GHT scores; this feature measures multiple values similarity.

8.3.3 Experimental Processes

We measure the effectiveness of the system when using RF matching, meta-

feature matching, and SIFT matching. The system implementation details of these methods are explained in the following sections.

8.3.3.1 SIFT matching

Matching using SIFT keypoint descriptors is shown in Figure 8.3.
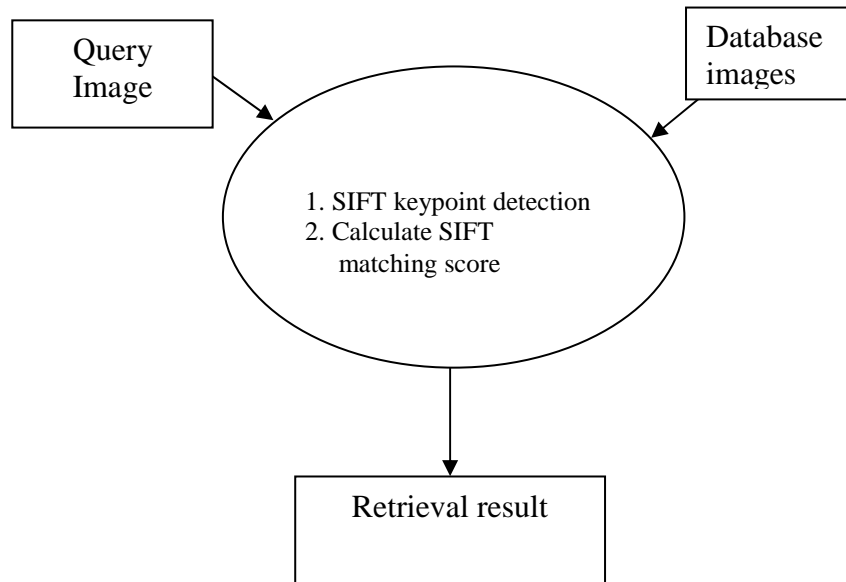
Figure 8.3 SIFT matching

We extract keypoints and descriptors from each image by means of the SIFT detector, and then calculate the SIFT matching score as described in section 5.3.1.2.

8.3.3.2 Meta-feature matching
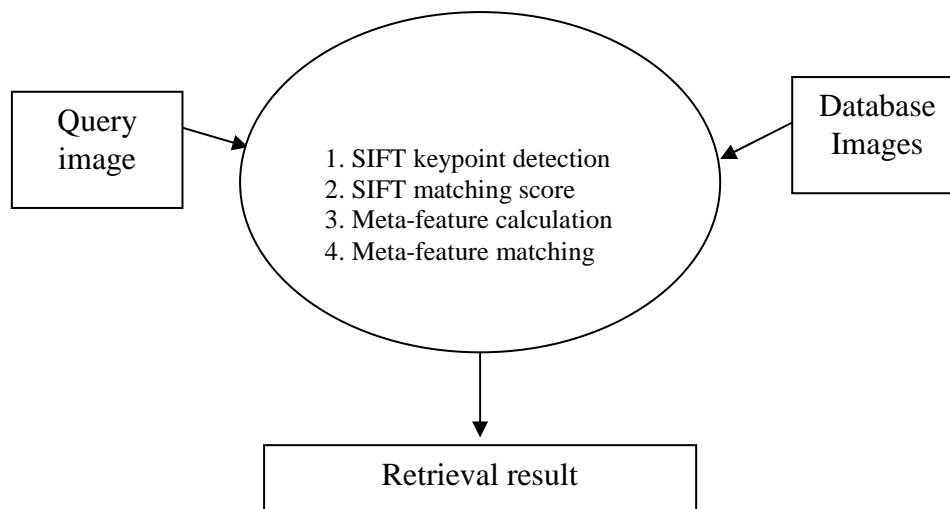
Figure 8.4 Meta-feature matching

Matching using a meta-feature vector is shown in Figure 8.4 We first extract keypoints and keypoint descriptors from each image by means of the SIFT detector, then calculate the SIFT matching score as described in section 5.3.1.2. The vector of meta-feature scores representing the high-level image appearance properties for each compared database image is constructed as explained in section 6.1, and finally the meta-feature matching scores are obtained as explained in section 7.2. In all experiments conducted during this validation, we in effect use nearest neighbours matching to match SIFT keypoint descriptors by setting Lowe's log-likelihood cut-off ratio criterion for matching to 1.0 (the ratio between the match score of the best matching descriptor and the next best matching descriptor). This modification has the effect of considering all descriptors when constructing matching pairs, as opposed to only matches between highly distinctive descriptors. We justify this approach on the grounds that it greatly improves the number of SIFT keypoint descriptor matches we obtain and it also appears to improve the overall performance of the system, compared to using the standard log-likelihood ratio of 0.8. We believe that use of nearest neighbour matching is appropriate here as all high-confidence keypoint descriptor matches are likely to be significant in trademark images since such images contain geometric figures which are likely to be self similar that would otherwise be rejected by log-likelihood matching. Therefore, due to an economy of form usually present within trademark images, all good keypoint matches are potentially important and should be recorded.

8.3.3.3 Relevance Feedback (RF) matching

Matching using RF matching is shown in Figure 8.5. We extract keypoints and keypoint descriptors from each image by means of a SIFT detector, and then calculate the SIFT matching score as described in section 5.3.1.2. The meta-feature vector is constructed as explained in section 6.1, and used to obtain meta-feature matching scores, as described in section 7.2. Finally, the RF matching score is computed using ID3 decision trees, as described in section 7.2. To generate relevance feedback automatically for each query image, a set of relevant images is provided by program code implementing an "ideal observer" (Xu et al., 2009; Farag and Wahab, 2003). This ideal observer supplies the appropriate "ground truth" set for each query image as illustrated in section 8.3.1 where a

Patent office examiner has identified images deemed to be relevant.
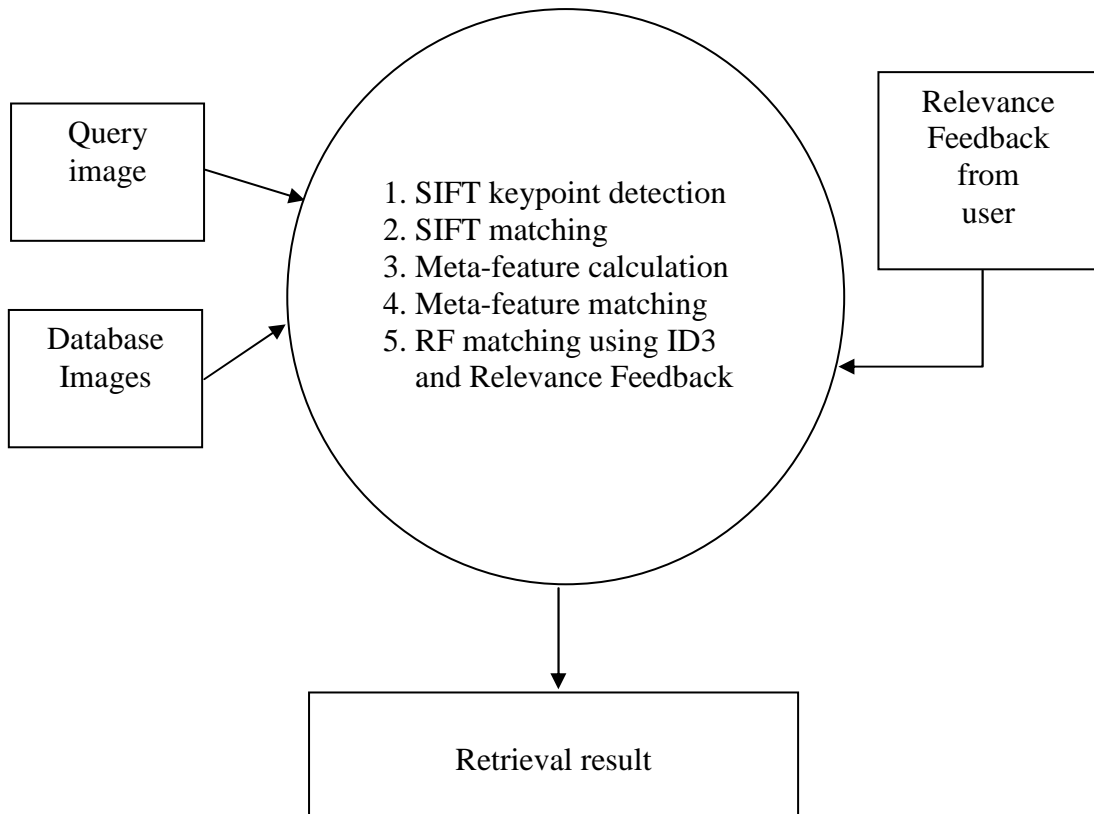
Figure 8.5 Relevance Feedback matching

8.3.4 Measurement method

The system effectiveness is measured by precision and recall, following Van Rijsbergen's definition that effectiveness strictly measures the system's ability to retrieve relevant documents that satisfy the user. Precision and recall can be used to measure information retrieval performance in this context (van Rijsbergen, 1979; Makhoul et al, 1999).

8.3.4.1 Precision and Recall

Precision (*P*) and recall (*R*) can be defined as (van Rijsbergen, 1979):

$$P = \frac{|A \cap B|}{|B|}$$
(Equation 8.1)

$$R = \frac{|A \cap B|}{|A|}$$
(Equation 8.2)

where | . | is the counting measure,

A is the number of relevant images,

B is the number of images that the system can retrieve.

The contingency table for calculating precision and recall from *A* and *B* is shown in Table 8.2.

Table 8.2 The contingency table for calculating precision and recall

	Relevant	Non-relevant	
Retrieved	$A \cap B$	$\overline{A} \cap B$	B
Not retrieved	$A \cap \overline{B}$	$\overline{A} \cap \overline{B}$	$\overline{B}$
	A	$\overline{A}$	

8.3.4.2 Precision and Recall of top-k

Relevance feedback from the user is applied to our system in the RF matching method to facilitate system learning. The number of images per round of interaction (i.e. the set of relevance feedback example images) should be small (Zhou and Huang, 2003), and is called top-k. We assign top-k to be 9 in accordance with the user's comfort and available computer display area.

The average top-k precision and recall are utilized to measure the system performance (Li et al., 2008; Chakrabarti et al., 2006), by employing Equations 8.1 and 8.2, with the number of images limited to top-k.

Higher precision and recall implies a more effective image retrieval system, therefore we measure system improvement by comparing the precision and recall of our three proposed methods.

8.4 Discussion of results

Experimental results for system effectiveness in terms of precision and recall are examined in section 8.4.1, case studies for retrieval are given in section 8.4.2, and section 8.4.3 summarizes the findings.

8.4.1 Experimental results

The average precision (over 12 different query images) for each top-k (from 1 to 200) of each query performed by the system is shown in Figure 8.6. The results indicate that RF matching gives the best average precision score, the feature matching is second, and SIFT matching is the poorest. The cumulative precision for RF matching is higher than the scores for feature matching and SIFT keypoint

descriptor matching by about 23 and 46 respectively. While, the average precision score for RF matching is higher than scores for feature matching and SIFT matching by about 1 % and 2 % respectively.
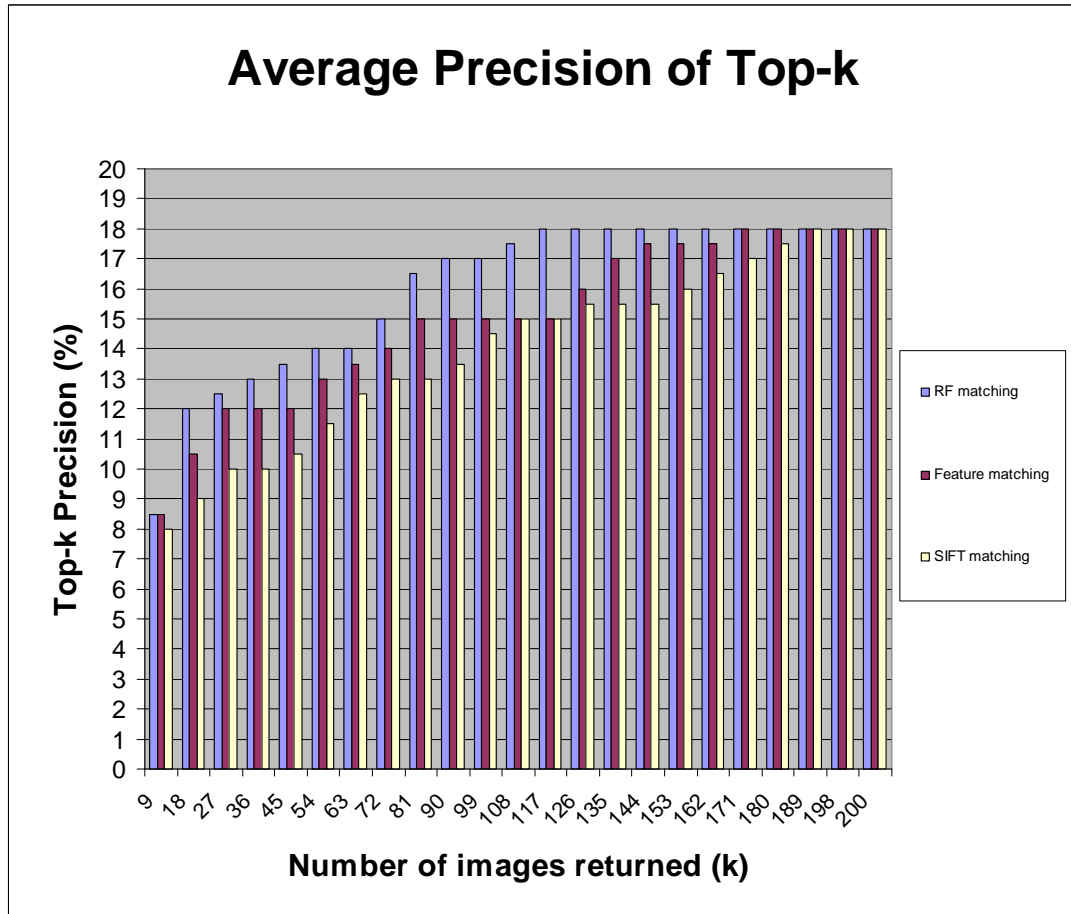
Figure 8.6 System precision for RF matching, Meta-feature matching, and SIFT matching

The cumulative and average precision of the validated systems are shown in Table 8.3.

Table 8.3 Cumulative and average system precision

Method	RF matching	Feature matching	SIFT matching
Cumulative	368.5	346	323
Average	16.02174	15.04348	14.04347826
Standard Deviation	2.673431	2.742399	3.048520668

Relevance feedback and ID3 are intended to provide high-level concepts to bridge the semantic gap when comparing similar images (Liu et al., 2007). Indeed from the results presented in Figure 8.6, both Meta-Feature matching and RF matching are performing better than standard SIFT matching for all page queries. RF matching starts to increase in performance notably after 18 images have been returned (just 10 percent of the database), possibly indicating that for

RF to be effective a certain minimum number of positive training examples must be provided when relying on ID3 for feedback classification.

The average (over 12 different query images) top-k recall for each query cycle performed by the system is shown in Figure 8.7.
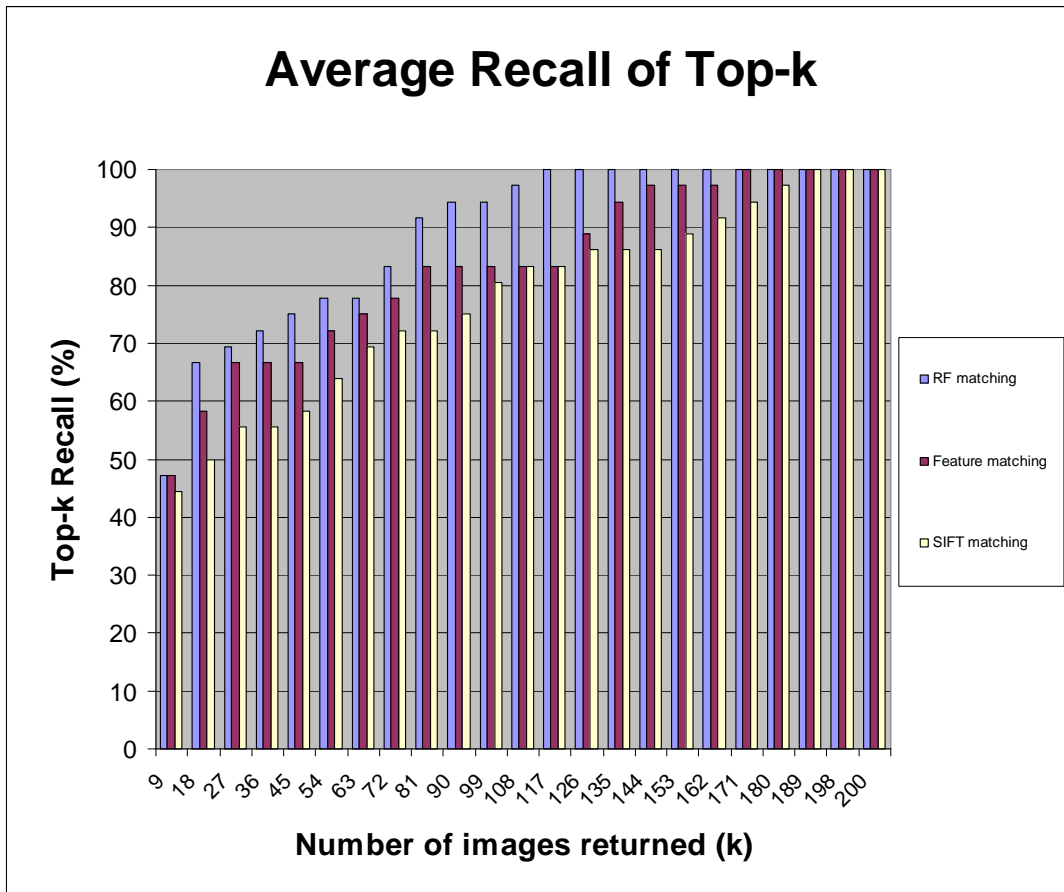
Figure 8.7 System average recall for RF matching, meta-feature matching, and SIFT

The results show that RF matching gives a better recall score for all top-k values. The cumulative and average recall of the systems is shown in Table 8.4.

Table 8.4 Cumulative and average system recall

Method	RF matching	Feature matching	SIFT matching
Cumulative	2047.21	1922.2	1794.43
Average	89.00913043	83.57391	78.01869565
Standard Deviation	14.85255959	15.23519	16.93610299

RF matching produces the best cumulative and average recall scores, feature matching is second, and SIFT keypoint descriptor matching is the poorest. The cumulative precision score for the system using RF matching is higher than the scores for feature matching and SIFT matching by about 125 and 253

respectively. The average recall score for RF matching is higher than scores for feature matching and SIFT matching by about 5.4 % and 11 % respectively. RF matching starts to increase in performance after 10 percent of the database images have been returned, once more indicating that improved feedback classification might be a route to improved performance. In this case the RF performance improvement is more pronounced, RF matching retrieving all similar images having returned 126 database images, while SIFT must return 189 images before finding all query images. Therefore RF matching would appear to be significantly better than SIFT at finding the most difficult to match residue of similar images.

Normalized precision (P_n) and normalized recall (R_n) are used to measure system performance in other trademark retrieval systems (Eakins et al., 1998; Jiang et al., 2006). They are calculated as follows:

$$R_n = 1 - \frac{\sum_{i=1}^{n} R_i - \sum_{i=1}^{n} i}{n(N-n)} ,$$
(Equation 8.3)

$$P_n = 1 - \frac{\sum_{i=1}^{n} (\log R_i) - \sum_{i=1}^{n} \log(i)}{\log(\frac{N!}{(N-n)!n!})} ,$$
(Equation 8.4)

R_i is the rank at which the relevant image i is actually retrieved,

n is the total number of similar images, and

N is the total number of test images.

These values rank retrieval performance from 0 (worst case) to 1 (perfect retrieval).

The normalized precision and recall for our system using RF matching were $P_n = 0.89 \pm 0.11$ and $R_n = 0.83 \pm 0.10$, while Artisan's performance (Eakins et al., 1997) was $P_n = 0.65 \pm 0.18$ and $R_n = 0.93 \pm 0.05$. A more recent system that applies Gestalt principles proposed by Jiang (Jiang et al., 2006) reports $P_n = 0.66 \pm 0.18$ and $R_n = 0.87 \pm 0.11$.

We employ same query set and trademark image database as in Artisan's system, but Jiang et al. used a different query set and trademark image database.

We can see that all three of our approaches achieve high performance: our system achieving a higher precision figure while the Eakins and Jian's perfoming better than ours in recall. Normalized precision measures the ability of a system to

filter information according to some given criterion (Zhang and Mostafa, 2002). These above results are therefore consistant with an imporved ability to determine the relevance of images returned in response to a query and therefore indicate that our approach is very promising, compared to other reported trademark retrieval systems that represent the state-of-the-art.

8.4.2 Retrieval Case Studies under Relevance Feedback

To investigate the operation of the system when retrieving trademark images, we now examine the retrieval data in detail. The retrieval results of the five closest matches for twelve query images are given in Table 8.5. The last top-k for each query corresponds to the final round of top-k images returned using RF matching. The relevant images for each query are marked by crosses (x).

The best retrieval result is always the smallest top-k and the worst is the last top-k in the results.

In Table 8.5, the ninth query used 9 images of last top-k to retrieve 4 similar images, this particular query image and relevant images are shown in Figure 8.8. A query image begins each row, the five matched images are then shown, and the last top-k appears at the end of a row. Similar images are specified by crosses.

| | (b1) | (b2) | (b3) | (b4) |
| (a) Query image | (b) relevant images |
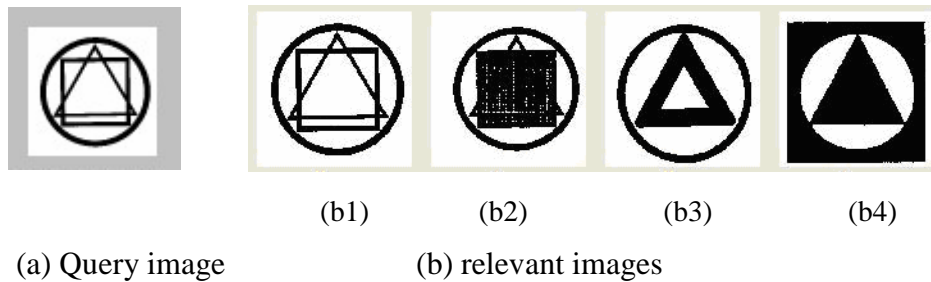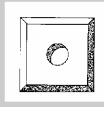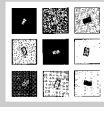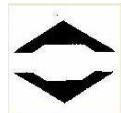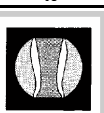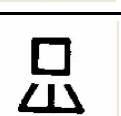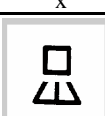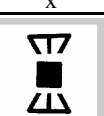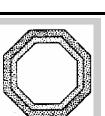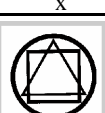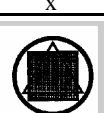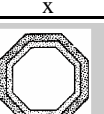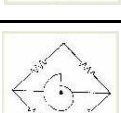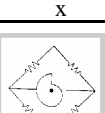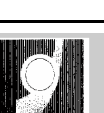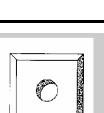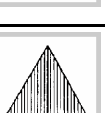
Figure 8.8 The best performing query image and relevant images

Three of the images in Figure 8.8 (b1, b2, and b4) contain a circle, rectangle, and triangle. Image b3 contains a circle and triangle and the capital letter Delta, Δ. Furthermore, b1 and b2 contain occluded elements. The best property to retrieve these similar images would appear to be global similarity because all of these images are similar as a whole image rather than being considered as a collection of components. This shows that the system can retrieve multiple, overlapping component images, and in this case the system selected meta-feature 6, which measures global similarity. The tree generated by ID3 for

this query is shown in Figure 8.9.

Table 8.5 Retrieval results for the five closest matches for twelve example query images.

Query image	Retrieval result in Top 5	Last Top-k
1		43
2		35
3		75
4		100
5		117
6		9
7		9
8		9
9		9
10		65
11		9
12		9

Figure 8.9 A decision tree generated by ID3 for the query in Figure 8.8

The worst-case query result is the fifth query in Table 8.5 which used 117 images to retrieve 4 relevant images. The query image and relevant images are shown in Figure 8.9.

                     (b1)   (b2)   (b3)   (b4)

(a) Query image   (b) relevant images
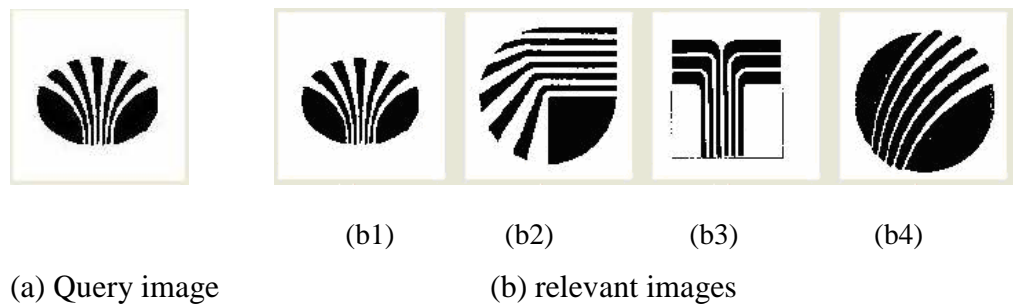
Figure 8.10 Poorest performing query image and relevant images

The system selected features 1, 2, 4, 5, 6, 14, and 15, which are orientation similarity, pattern similarity, nearness proximity, nearness proximity, pattern similarity, self similarity, and symmetry simplicity. The tree generated by ID3 for this query is shown in Figure 8.11. and the relative complexity of this tree is evident when compared to the tree produced by the best performing query, in Figure 8.9. This complexity suggest that ID3 is struggling to partition the meta-feature space to specify relevant images based on the training examples supplied by relevance feedback and as a consequence the tree generated is possibly overfitting to the supplied training data.
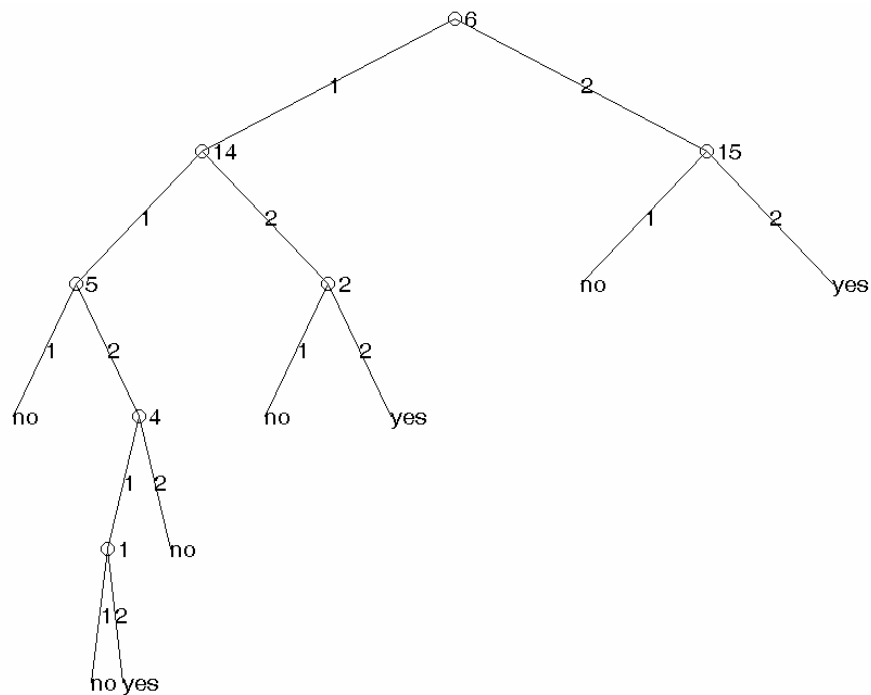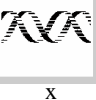
Figure 8.11 decision tree generated by ID3 of query in Figure 8.10

The query image consists of multiple linear components in several orientations, while the global relative position of the sub-components between the query and relevant images can vary. Therefore, these meta-features do not appear to be sufficiently diagnostic to capture the global banded pattern that clearly appears to constitute similarity. In this case the inclusion of a simple global spatial frequency feature might have been sufficient to detect this type of image similarity. However, had the self-similarity meta-feature been selected by ID3 early in the search, then it may also have picked up the relevant images more quickly. Possibly the main explanation is that the first pass over the database relies only on meta-feature matching, where all meta-features have equal weight. Only when the first similar image has been found is it possible to define what is meant by similarity, up until that point only dissimilarity can be defined. In other words, until the first similar image is found the image cues which are diagnostic of similarity cannot be readily induced. Therefore, an interesting possibility would be for the user to supply not only a query but also an example of a similar

image to this query. The retrieval results are based on the selected features and relevance feedback. Then, we show two examples of the effect of relevance feedback by changing the relevant images in Table 8.6. A query image begins a row, the five matched images are shown in next, and the last top-k appears at the end of a row. Similar images are specified by crosses.

Table 8.6 The retrieval results of the five closest matches when modifying relevance feedback for the two example query images in Table 8.5.

Query image	Retrieval result in Top 5	Last Top-k
1	 x x	21
2	 x x	33

In the first example, we have selected query image 9 of Table 8.5 to interrogate the trademark database. The second relevant image selected is different to that selected Table 8.5 and consists of several components within overall octagonal configuration. Both images are similar in their similarity and proximity appearance properties. The system selected features 8 and 4, which measure the scale ratio between matched keypoint sets and the spatial spread of the matched points. Both the tree generated by ID3 for this query, shown in Figure 8.12, and a subset of the retrieval images returned are different to those generated in the first study, as a consequence of the different appearance propeties being captured via feature seleection directed through relevance feedback.
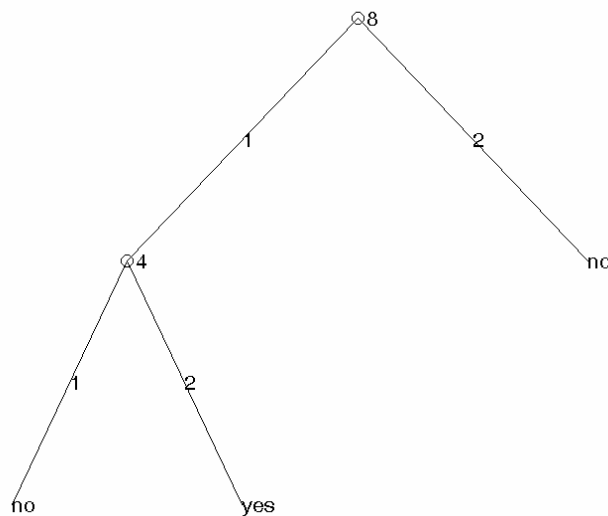
Figure 8.12 Decision tree generated by ID3 for query 1 in Table 8.6

In the second example, we have selected query image 5 of Table 8.5 to interrogate the trademark database. The second relevant image selected is different to that selected in Table 8.5 and consists of multiple components within an overall double helix configuration. Both images are similar in their similarity appearance property. The system selected features 17, 9, and 1, which represent sub-component similarity, scale ratio between matched keypoint sets, and global rotation. The tree generated by ID3 for this query is shown in Figure 8.13. Once more, we demonstrate differences in the tree generated and similar images returned when the relevance feedback supplied is modified.
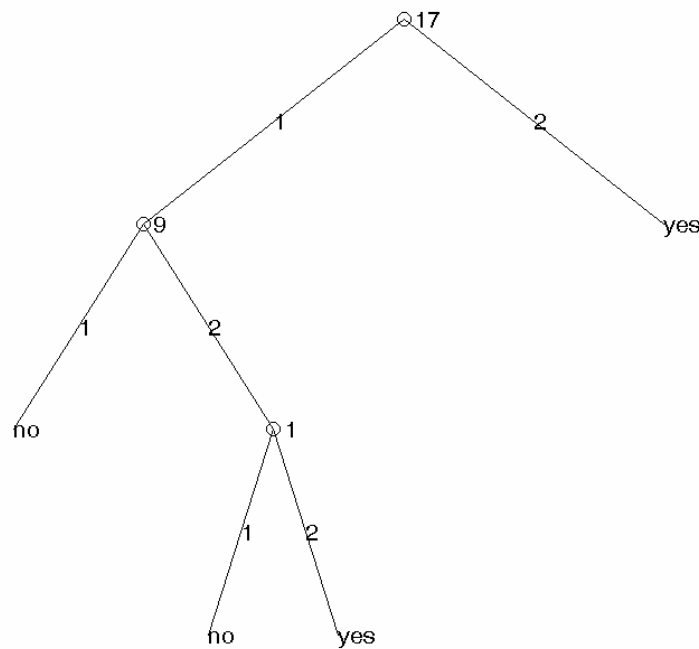
Figure 8.13 Decision tree generated by ID3 for query 2 in Table 8.6

8.4.3 Experimental Summary

Use of SIFT appears to give a significant improvement over Eakin's system since local features are being matched and the similar trademark images appear to be similar in a global sense. Accordingly, standard SIFT, measuring numbers of matching features or GHT similarity, would appear to be appropriate for matching under these conditions. Matching using the meta-features alone is also more successsful than using SIFT, suggesting that the benefits of grouping are indeed being realised, as discussed below.

In the initial experiments we observed that there is the potential for grouping to yield an improvement, and our grouping approach as implemented is

achieving a significant improvement in discrimination. However, the degree of benefit endowed by our system is expected to be contingent upon the degree of grouped structure present within the images comprising the specific trademark image collection being searched. Given that method used for comparing the meta-features is comparatively unsophisticated, it may be possible to realise even greater performance improvements by using individual classifiers based on each measure, appropriately weighted as in Adaboost (Martínez-Contreras et al., 2009), and always include all features, suitably weighted.

The experiments show that using RF matching gives significantly better performance than either SIFT matching or meta-feature matching alone. These results are consistent with the evidence presented that supports the hypothesis that RF matching can be used to select appropriate local features for GBPG properties. Therefore, relevance feedback indeed appears to provide more information to enable the system to select appropriate features for measuring image similarity, given that ID3 can learn relevance classifications.

We have also demonstrated that the system will respond to changes in relevance feedback, generating decision trees based on meta-feature selections that attempt to capture the visual appearance properties represented in the relevant images. Therefore, RF matching appears to support flexible decision-making based on such relevant image selections.

RF matching manages to improve the effectiveness of the system to a significant degree. In Figures 8.6 and 8.7, the system using RF matching achieves better results over the entire search session because, we believe, sufficient feedback information has been gathered to allow ID3 to generate appropriate rules to classify similar images by learning to select appropriate meta-features based on GBPG properties.

8.5 Conclusion

We have demonstrated that our system that supports perceptual grouping by GBPG properties and high-level concepts. The GBPG properties are calculated by grouping the spatial configuration of keypoint locations into meta-features, and high-level concepts are captured by measn of relevance feedback and decision tree classification as described in chapters 6 and 7.

Our system based on meta-features that encode high-level concepts can improve effectiveness by retrieving multiple GBPG properties in similar trademarks, when compared to our system using SIFT kepoint descriptor matching alone or meta-feature matching alone. Furthermore, our RF based system would appear to hold the potetnial to improve trademark image retrieval performance.

In the next chapter, we summarise the contributions of the research and discuss for the future work.

Chapter 9

Conclusion and Future work

This thesis has investigated a practical technique for trademark image retrieval by perceptual grouping. The technique is based on defining meta-features which are calculated from the spatial configurations of local image features in order to imitate perceptual grouping. User relevance feedback has been integrated within our approach to allow human judgment to influence the definition of image similarity when retrieving trademark images. We summarise the contributions and discusses ideas for future work in this chapter.

9.1 Objectives Revisited

A stated in Chapter 1, the problem addressed in this thesis can be posed by the question "how can we help people identify a putative trademark as being sufficiently original?" To answer this question we set out to investigate how to analyse a trademark image, how to identify those trademarks that are most similar, and how to organize their presentation to the best effect. Since trademark images are composed of potentially complex elements, it was realised that a means for representing these image sub-structures would be important. In addition it was realised that being able to capture human perceptual judgement in terms of what constitutes image similarity would also be important, since human beings ultimately arbitrate in disputes of whether a trademark is deemed novel or not.

When this work was initiated few techniques were available for partial shape matching that supported multi-component retrieval. Furthermore, many of the techniques for image database retrieval that did exist required an exact image segmentation, which is difficult to achieve with the level of reliability required for a real trademark retrieval application. In addition, few researchers had attempted to apply principles derived from human perception for shape retrieval that would

allow perceptually meaningful configurations of trademark image components to be represented and classified. Finally, since global image features are not suitable for retrieving occluded or connected components in an image, it was realised that interest points, i.e. local features, as adopted by many vision researchers (Schmid and Mohr,1997; Lowe,1999; Wolf,2000; Sebe,2001) for general image database retrieval would have to be extracted and matched in order to compare trademark images reliably. Therefore, the main aim of this research has been to develop a method for solving the partial matching and shape perception problem by investigating the following questions:

1. How can interest points be used to distinguish trademark images?
2. Which interest point techniques are most accurate when applied to distorted trademark images (noise, rotation, and scale)?
3. How can perceptual grouping methods serve to group interest points and represent these within a shape descriptor (i.e. a meta-feature vector)?
4. Which techniques can be used to exploit shape descriptors (meta-feature vectors) when retrieving abstract trademark images?

9.2 Summary of Contributions

The main contributions of this thesis address the above questions and are summarized as follows:

- Analysis and application of point matching to trademark image matching and retrieval
- Grouping local features into meta-features for trademark image retrieval
- Relevance feedback based on classifying meta-features
- An implementation and evaluation of a computer-based abstract trademark image retrieval system

9.2.1 Analysis and application of point matching to trademark image matching and retrieval

We investigated the application of interest points to abstract trademark image retrieval. Interest points have been successfully used to recognise objects (Lowe, 1999; Wolf, 2000; Sebe & Lew, 2003) but there had been no prior reported research on how to retrieve similar abstract trademark images by interest points, since most retrieval systems concentrated on global features rather than local

features when this work was initiated. Local features can reflect both local and global image characteristics, but approaches based on global features had been more widely reported since approaches based on local features tended to be more computationally expensive and complex. The issues to be addressed were:

- Where to apply local feature extraction?

- What local features should be utilized?

- How to use extracted local image features?

From the outset, we proposed to use interest points because they support Biederman's concept that humans recognize images by distinctive elements. In Chapter 4, we demonstrated that many detectors can extract the same areas in transformed trademark images, and that it is possible to retrieve similar trademark images based on interest points. From our study of interest point detectors, section 9.2.5 below, it became apparent that the Harris and Chabat detectors could potentially perform sufficiently well to serve as the basis for local feature extraction in trademark image retrieval. We also realised that interest points alone are likely to be an insufficient representation on their own and that further discrimination power would be required in terms of uniquely characterising the local appearance of the interest point. Accordingly, we identified the SIFT local feature extraction algorithm, also based on the Harris detector, as a good candidate for characterising the appearance of local features in a scale and rotation independent manner. We then conducted a number of experiments described in Chapter 5 to establish the suitability of SIFT for trademark image retrieval.

By integrating the SIFT feature detector with the General Hough Transform, David Lowe (Lowe, 2004) was able to perform basic scale and rotation invariant grouping of SIFT keypoint descriptors (local features). Matching keypoint descriptors which have been extracted from compared query and database image can be clustered in Hough space and a peak in this space reveals a common scale, rotation, and translation between similar configurations of keypoints. Such similar configurations are said to be "non accidental" and the utility, i.e. matching performance, of this approach when comparing trademark

images was investigated and verified in Chapter 5. We believe that the above work represents the first systematic study into the use of the SIFT algorithm for trademark retrieval. The above work addresses the first two objectives posed in this thesis, namely, how interest points can be applied to trademark image retrieval and which type of interest points should be used in this context.

While retrieval based on SIFT produced promising results, we believed that it was possible to obtain improved results by: firstly, attempting to develop an algorithm that quantifies aspects of human visual perception of shape; secondly, by then guiding the selection of retrieved images, represented quantitatively in terms of measures based on human perception, using relevance feedback classification.

9.2.2 Grouping local features into meta-features for trademark image retrieval

Higher-order visual perception plays an important role in human judgement of image similarity (Goldmeier, 1972; Eakins, 1997). In addition, shape is important for identifying abstract trademark images which contain multiple graphical elements. In chapter 6, we proposed a new technique for measuring certain visual properties of the appearance of such multiple graphical elements by defining meta-features based on interpreting the spatial configuration of matching interest points. We initially proposed 27 meta-features to support perceptual grouping using non-accidental properties derived from interest points to serve as the foundation for our computational basis for visual shape perception.

In order to verify that the above meta-features were capable of measuring appropriate visual appearance properties, we conducted an experiment whereby sets of test images were generated such that each set represented a different appearance property and each image of each set represented the chosen appearance property to a different degree. By correlating the responses from each of our meta-features to the degree of each appearance property presented in each test image (details in chapter 6 and appendix B) it was possible to verify that the meta-features did indeed respond to specific visual appearance properties and determine which meta-feature responded to which particular appearance property,

or properties.

In order to eliminate redundant meta-features, Principle Component Analysis (PCA) was applied to the above correlation results to allow a factor analysis of the meta-features to be conducted. As a result, only 17 meta-features were found to be worth retaining for the purpose of imitating perceptual grouping in the following three Gestalt properties: proximity, similarity, and simplicity. The meta-features implemented compute: global rotation, global pattern similarity, global pattern overlap, spread of the match points, structural configuration of the match points, scale different of matched pattern, self similarity, symmetry, and sub-component similarity.

From the basic experiments conducted, it appeared that we were indeed able to capture and quantify Gestalt proximity, similarity, and aspects of simplicity within stylised image examples. It was also clear from the correlation experiments that many of the meta-features we developed were substantially correlated, which could in turn result in a significant degree of "cross-talk" between different modes of visual appearance. In principle, we could have defined a new set of meta-features based on using the correlation response eigenvectors to decouple their response to the different modes of visual appearance input. Unfortunately, this approach would render the new feature set dependent upon the training data used to generate the eigenvectors, there being no guarantee that such decoupled meta-features would perform adequately when applied to a substantially different dataset. Therefore, we decided simply to retain only the best performing meta-features, i.e. those correlating most highly with a specific mode of visual appearance with the expectation that these meta-features would be more likely to retain a reasonable degree of independence from the image subject matter itself. Since the entire gamut of 17 meta-features would be available for making similarity comparison judgements, it was hypothesised that it would still be possible to disambiguate different types of visual appearance property.

A compounding factor in the use of the developed meta-features is that several visual appearance properties are likely to co-exist within any one trademark image, and as such would signal simultaneously present appearance

properties that would be difficult to disambiguate from cross-talk artifacts. Two further limitations of the proposed scheme are due to the limited representation power afforded by the chosen meta-features themselves and due to the global nature of their application to represent the sets of keypoints extracted from and matched between pairs of compared trademark images. Alternative meta-feature representations might couple to improved underlying feature representations, e.g., contour, or line fragment, extraction mechanisms might be processed produce classical Gestalt continuation groupings. As indicated in the future work section, hierarchical clustering techniques may be able to detect sub-groupings when clustering local features.

Despite the above limitations of the developed meta-features, they did appear to show promise as a means of introducing a simplified model of human visual perception to tackling the trademark similarity problem. Therefore, in order to mitigate the limitations of the simple experimental conditions applied to validate the operation of the meta-features we had developed, a subsequent validation test using a database of real trademark images was then undertaken, as detailed in Chapter 8 and discussed here in section 9.2.4. Similar work on meta-features based on Gestalt grouping of local features had not been reported in the current computer vision literature at the time of undertaking this work. The above work addresses the third objective set out in this thesis, namely, how to group local features such that similarities between trademark images, based on visual appearance properties, can be computed and represented numerically.

9.2.3 Relevance feedback based on classifying meta-features

Human judgment ultimately determines which trademark images are deemed to be similar. In order to incorporate this judgment within the developed trademark retrieval system, we decided to adopted relevance feedback to allow the visual appearance properties of relevant and non-relevant images to be determined by example. Given the limited training data available when constructing a relevance classifier, the intrinsically non-parametric machine learning algorithm ID3 was selected to construct decision trees by means of rule induction.

The learned relevance classifier concept was explored initially by

designing an experiment to determine if visual similarity could be defined by an arbitrary perception threshold according to the degree of presence of a visual appearance property using the image test set described in section 9.2.2. By setting an arbitrary similarity threshold on a single visual appearance property, it was possible to demonstrate that ID3 could construct decision trees that would successfully identify visual appearance similarity thresholds based on selecting and comparing the values generated by appropriate types of meta-feature. In other words, we were able to show that if a particular visual appearance property, such as symmetry was deemed to be similar at a certain (arbitrary) perception threshold, by supplying meta-features extracted from feedback images depicting this desired similarity threshold (i.e. relevant and non-relevant examples, respectively above and below the desired appearance property threshold), then ID3 could induce a decision tree that would correctly classify input images depicting varying degrees of this appearance property correctly. Furthermore, the meta-features selected by ID3, by means of which it constructed decision trees to classify appearance properties, were usually in accordance with those anticipated by their design function and also in accordance with those verified using the correlation procedure described in 9.2.2.

A limitation of the above experiment is of course that the test set was deliberately simplified in order to be able to determine whether it is indeed possible to specify visual similarity by means of relevance feedback in the context of visual appearance properties. Therefore, motivated by the above promising results a further validation experiment was conducted, as described in Chapter 8 and further comments in section 9.2.4, to determine if relevance feedback could indeed improve trademark image retrieval performance.

We believe that our approach to capturing high-level visual concepts encoded by means of meta-features and specified by example through relevance feedback and decision tree classification to support flexible trademark image retrieval to be wholly novel. This work addresses the fourth objective specified in this thesis, namely, "Which techniques can be used to exploit shape descriptors (meta-feature vectors) when retrieving abstract trademark images?"

9.2.4 An implementation and evaluation of a computer-based abstract trademark image retrieval system

The main contribution of this thesis is the framework to build an effective computer-based abstract trademark image retrieval system and thereby address the principal question posed in this thesis: "how can we help people identify a putative trademark as being sufficiently original?"

In our prototype trademark retrieval system, relevance feedback has been implemented such that a human operator can select relevant images when presented with a set of nine trademark images retrieved in response to submitting a query trademark image. Image similarity is computed during this initial query image by computing the meta-feature vector from the query image and comparing these with the meta-feature vector extracted from each database image. The vector dot product is computed between compared meta-feature vectors and the dot product score is used to rank the returned images in sequence of similarity. Meta-features extracted from this retrieved trademark image set, having been labelled as relevant or not relevant are then used to construct a decision tree classifier using the ID3 algorithm. In subsequent query cycles all current and prior relevant and non-relevant images are used to build a new classifier.

We evaluated the system with real trademark images that contain several Gestalt properties in order to measure the system performance, as detailed in Chapter 8. Three operating modes were investigated: image retrieval by matching basic SIFT image features, image retrieval by matching meta-features and finally image retrieval by meta-feature matching and relevance feedback classification using meta-features. To maxim use the number and quality of SIFT matches obtained, SIFT keypoint descriptor comparisons (including those used to construct meta-features) were made using nearest neighbour matching as opposed to log-likelihood matching. This validation experiment used the same query set and the trademark image database as in Artisan's Evaluation (Eakins, 1997) comprising 12 trademark query images were used to search a database of 200 trademark images. Meta-feature matching produced a database search performance improvement over searching using the standard SIFT algorithm. However, the relevance feedback mode exhibited significantly better performance

than both the basic SIFT or meta-feature matching approaches. Better results using relevance feedback were observed to occur when relevant feedback images had been accrued by searching just over 10 percent of the database. This observation tends to suggest that the relevance feedback mode is able improve retrieval effectiveness after a training data has been supplied.

It was possible to verify that relevance feedback was indeed able to influence the images returned in response to a specific image query by observing the decision trees generated and the image set returned for at least two query runs where different images were selected during each run as being relevant while the same query image was used in reach run. For each run, different images were returned as a consequence of different images being selected as being relevant. Similarly, different decision trees were selected according to the dominant visual appearance property being active in the images returned as similar to the query, Chapter 8.

It must also be noted that the system produced highly viable normalised precision and recall figures; for the relevance feedback mode the highest normalised precision recorded being 0.89 and the highest normalized recall recorded being 0.83.

We compared the retrieval performance our system with two other state-of-the-art image trademark retrieval systems and as mentioned above the validation experiment used the same trademark image query and database set as in Artisan's evaluation. Unfortunately, we did not have access to the same query set and image database as used in Jiang et al's system evaluation. Using relevance feedback, our system achieves higher average normalised precision than both Eakins' Artisan (Eakins et al., 1998) system and a more recent system developed by Jiang et al. (Jiang et al., 2006). Although we are using different numbers of image data with Artisan system and a different query set and image data set with Jiang et al's system, our approach would appear to have the potential to improve retrieval effectiveness.

We believe that the system we describe above comprising SIFT local image features in combination with visual appearance meta-features and support

for relevance feedback based image similarity represents an entirely new development in trademark image retrieval.

Minor contributions include:

9.2.5 A study of interest point detectors

The evaluation of interest point detectors is reported in Chapter 4. We investigate and evaluate interest point detectors that calculate interest points directly from an image. We choose four effective detectors for our experiments comprising the Harris detector, Chabat detector, SUSAN detector, and Wavelet-based detector. We measured the repeatability of these interest point detectors when applied to trademark images to which known spatial have been applied. Our results revealed that the Harris detector has the best repeatability and Chabat detector also offers good results, with more than 50% repeatability. Accordingly, the Harris and Chabat detectors appeared to have the potential to be used in trademark image retrieval. The above results contribute to addressing the first objective in this thesis, namely the analysis of point matching in trademark images.

9.3 Future work

This section identifies potential directions for future work for research initiated by this thesis.

- In the current system, SIFT keypoint descriptors and meta-features are generated offline, leading to a large number of feature vectors and the increase in the required runtime computation may well be sufficiently low to allow feature vectors to be extracted "on the fly". There is considerable potential for improving feature vector comparisons in the current implementation. For example, a comprehensive indexing mechanism by means of hashing functions could be implemented to reduce search time when comparing query feature vectors with those of the database. Likewise, Hierarchical K-means clustering as used in CBIR could be adopted here to feature accelerate comparisons (Fukui et al, 2004;

Murtagh, 1983).

- We adopted local features, i.e. kypoints, to support three Gestalt properties which can be directly measured by appearance meta-features. Hence, better image features or perhaps specialized features based on the SIFT feature extraction framework might have the potential to increase the system's perceptual grouping abilities. For examples, features derived from curve fitting could be used to measure continuity, and features based on local gradients computed at two locations simultaneously could be used to measure co-linearity. These extended features could be integrated within the system to generate structure vectors that support additional Gestalt properties.

- It would be interesting to investigate use of image contours and their descriptors in the future because they could give more discrimination power to measure foreground shapes. For instance, the eigenvalues computed from contours (Tsai, D.-M. et al., 1999) and local curvature estimates computed around contours (Fishler and Wolf, 1994) which could be used to distinguish particular component boundary shapes. This method might also be used to address the segmentation problem by applying statistical learning models to contours (Cootes et al., 1995) and then characterising the contour shape in PCA space.

- We used the Hough space accumulator to cluster components represented by keypoints. More sophisticated grouping mechanisms should perform recursive parsing of image components into sub-components in order to represent more subtle visual shape or pattern characteristics. For example, the multiple histograms of similar sub-components should be further examined (Ankerst et al., 1999; Wolf et al., 2000). Moreover, hierarchical K-means clustering potentially represents a better approach for representing visual sub-patterns, and is used in CBIR to search for target objects (Fukui et al, 2004; Murtagh, 1983).

- Explicit encoding of image sub-components could serve the calculation of self-similarity or allow image sub-components to be matched. For

instance, a KD-tree could be used to facilitate multidimensional search for each component. This encoding could support meta-feature vectors in a high-dimensional space and provide a new structure for efficient similarity search.

- Alternative classifiers and classification schemes should be investigated in order to better categorise groups of feature vectors. For example, the support vector machine (SVM) classifier is popular for efficient clustering and has the potential to optimise the discrimination of similar images. We could investigate the impact of perceptual grouping when using different methods of relevance classification.

- The trademark image system we present in this thesis only supports explicit relevance feedback. However, there is a better approach to relevance feedback that potentially might increase retrieval effectiveness in our system. Hopfgartner and Jose (2007) demonstrate that the inclusion of both explicit and implicit relevance feedback can improve retrieval effectiveness in the textual domain. They apply six implicit feedback categories: highlighting, keyframe selection, sliding bar annotation, metadata viewing, video browsing, video play duration (Hopfgartner and Jose, 2007). It is possible to apply some categories of implicit feedback to improve our system, e.g. detecting selection and then de-selection of any given image as being relevant, and also the decision time taken by the user to make selections. The combination of both types of feedback might assist the system to better capture the user's perception of image similarity to thereby allow the system to extract the most effective meta-features for each query.

- Given that the cost of missing a similar trademark image is potentially very significant, additional explicit cues might be supplied by the user. For example, in addition to supplying a query image, an example (or several examples) of similar trademarks were also provided, the system should then be better able to construct a classifier that can represent trademark similarity (as opposed to dissimilarity). This approach has the potential to improve retrieval performance during the early search cycles when not

many (or indeed any) similar images might have yet been discovered if only the query example were available. Also, if the user had the option of annotating the query image to select particularly diagnostic components, the keypoints associated with such regions could be increased in weight to bias the similarity score when assessing image similarity.

Appendix A

The proposed local features

We proposed 27 meta-features to measure perceptual grouping in trademark images. The meta-features are described in the following sections.

A.1 Summation of scale differences between sets of matched keypoints

We propose to calculate overall scale differences in order to measure size similarity between matching sets of matching SIFT keypoint descriptors. We calculate summation of the scale difference between pairs of matched keypoints (F_1) by the following.

$$F_1 = \sum_{i=1}^{n} (SQ_i / SM_i) \qquad \text{(Equation A.1)}$$

where:

n is the number of matched keypoints in the maximum accumulator (see

section 5.3.2 for details)

SQ is the scale of the matched keypoint in the query image.

SM is the scale of the matched keypoint in the database image.

A.2 Summation of orientation differences between sets of matched keypoints

We propose to calculate overall orientation differences in order to measure orientation similarity between sets of SIFT keypoint descriptors. We calculate summation of the orientation difference of pairs of matched keypoints (F_2) by the following.

$$F_2 = \sum_{i=1}^{n} abs(OQ_i - OM_i) \qquad \text{(Equation A.2)}$$

where:

n is the number of matched keypoints in the maximum accumulator.

OQ is the orientation of the matched keypoint in the query image.

OM is the orientation of the matched keypoint in the database image.

A.3 Moments from matched keypoints

We propose to compute Moments from the locations of matched keypoints to quantify their spatial configuration. The moments are calculated by:

$$m_{pq} = \sum_{i=1}^{N} x_i{}^p y_i{}^q \qquad \text{(Equation A.3)}$$

where *x* and *y* are the location of each matched keypoint in the image.

N is the number of the matched keypoints.

p and *q* are the orders of the computed moments.

In this research, we computed the moments up to third order in order to analyse orthogonal transformations (Mukundan & Ramakrishnan, 1998). Therefore, the extend features comprise 10 moments defined by:

$$F_3 = m00 \qquad\qquad \text{(Equation A.4)}$$

$$F_4 = m01 \qquad\qquad \text{(Equation A.5)}$$

$$F_5 = m02 \qquad\qquad \text{(Equation A.6)}$$

$$F_6 = m03 \qquad\qquad \text{(Equation A.7)}$$

$$F_7 = m10 \qquad\qquad \text{(Equation A.8)}$$

$$F_8 = m11 \qquad\qquad \text{(Equation A.9)}$$

$$F_9 = m20 \qquad\qquad \text{(Equation A.10)}$$

$$F_{10} = m22 \qquad\qquad \text{(Equation A.11)}$$

$$F_{11} = m30 \qquad\qquad \text{(Equation A.12)}$$

$$F_{12} = m33 \qquad\qquad \text{(Equation A.13)}$$

A.4 Standard Deviation (SD) of the residual spatial mismatch (error) between query and database keypoint locations following alignment via an affine transformation

We propose to use SD (standard deviation) to compute the distance-error between sets of matched keypoints. This feature measures the residual error between query and database keypoint locations registered by an affine transformation. The algorithm for calculating the SD of distance-error between sets of pairs of matched keypoints is shown in Figure A.1.

SD of error-distance of sets of matched keypoints algorithm

Step 1: Extract matched keypoints from the maximum GHT accumulator.

Step 2: Calculate Affine parameters of matched keypoints.

The solution is suggested by Lowe (Lowe, 2004).

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} m_1 & m_2 \\ m_3 & m_4 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \qquad \text{(Equation A.14)}$$

where (u,v) is the keypoint from the database image.

(x, y) is the keypoint from the query image.

$m1, m2, m3,$ and $m4$ are affine parameters.

t_x and t_y are the translation parameters.

Then, we can use at least 3 points to calculate the affine parameters by

$$\begin{bmatrix} x_1 & y_1 & 0 & 0 & 1 & 0 \\ 0 & 0 & x_1 & y_1 & 0 & 0 \\ & & \dots & & & \end{bmatrix} \begin{bmatrix} m1 \\ m2 \\ m3 \\ m4 \\ tx \\ ty \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \vdots \end{bmatrix} \qquad \text{(Equation A.15)}$$

Figure A.1 The algorithm to calculate SD of residual spatial mismatch (error) between query and database images following alignment via an affine transformation

SD of error-distance between sets of matched keypoints

Step 3: calculate affine transformation of query keypoints by Equation A.14.

Step 4: find the distance error (D_i) between each pair of matched database

keypoint and affine registered query keypoint using Equation A.16.

$$D_i = L_i \bullet L_j \qquad \text{(Equation A.16)}$$

where L_i and L_j are SIFT descriptors.

Step 5: calculate mean (Avg_{Di}) and stand deviation (Std_{Di})

of all error distances (D_i) in Step 4.

$$Avg_{Di} = \frac{\sum_{i=1}^{n} D_i}{n} \qquad \text{(Equation A.17)}$$

$$Std_{Di} = \sqrt{\frac{\sum_{i=1}^{n} (D_i - Avg_{Di})}{n}} \qquad \text{(Equation A.18)}$$

where n is the number of matched keypoints.

Step 6: find the inliner point error distance (D_j) according to

(Jung and Lacroix, 2001).

if $D_i >= (Avg_{Di} + (2 * Std_{Di})$
   $D_j = D_i$

end

Figure A.1 The algorithm to calculate SD of residual spatial mismatch (error) between query and database keypoint locations following alignment via an affine transformation (continue)

SD of error-distance between a pair of matched keypoints

Step 7: calculate the feature (F_{13}) by the standard deviation of all distances in Step 6.

$$F_{13} = \sqrt{\frac{\sum_{j=1}^{k}(D_j - Avg_{Dj})}{k}}$$ (Equation A.19)

where

Avg_{Dj} is the mean of the inlier distances that are calculated by Equation A.17.

m is the number of inlier distances.

Figure A.1 The algorithm to calculate the SD of residual spatial mismatch (error) between query and database keypoints following alignment via an affine transformation (continue)

A.5 Similarity score of matched keypoints

We investigate the similarity score from SIFT matching. The similarity score is obtained using the SIFT algorithm and is explained in section 5.3.2 of Chapter 5.

$$F_{14} = \frac{\sum_{i=1}^{n} D_{lowest}(i)}{n}$$ (Equation A.20)

where n is the number of matched keypoints.

D_{lowest} is the similarity score that calculated from Equation 5.11 in section 5.3.2 of Chapter 5.

A.6 The total number of matched keypoints

The total number of matched keypoints is defined to be the number of keypoints found in the GHT accumulator bin with the maximum value (see section 5.3.2 for details).

$$F_{15} = n$$ (Equation A.21)

where n is the number of matched keypoints.

A.7 Mean of scale differences between sets of matched keypoints

We propose to calculate mean of the scale difference (F_{16}) between matched keypoints in order to measure size similarity between sets of matching SIFT keypoint descriptors. The feature is calculated by the following.

$$F_{16} = \frac{\sum_{i=1}^{n}(SQ_i / SM_i)}{n}$$
(Equation A.22)

where:

n is the number of matched keypoints in the maximum accumulator.

SQ is the scale of the matched keypoint in the query image.

SM is the scale of the matched keypoint in the database image.

A.8 Median of scale differences between sets of matched keypoints

We propose to calculate the median scale differences (F_{17}) in order to measure size similarity between sets of matching SIFT keypoint descriptors. If n is an even number then the feature is calculated by:

$$F_{17} = \frac{(SQ_{(n/2)} / SM_{(n/2)}) + (SQ_{(n+1)/2} / SM_{(n+1)/2})}{2}$$
(Equation A.23)

If n is an odd number then the feature is calculated by:

$$F_{17} = (SQ_{(n+1)/2} / SM_{(n+1)/2})$$
(Equation A.24)

where:

n is the number of matched keypoints in the maximum GHT accumulator entry.

SQ is the scale of the matched keypoint in the query image.

SM is the scale of the matched keypoint in the database image.

A.9 RMS of scale differences between sets of matched keypoints

We propose to calculate the RMS (Root-Mean-Square) scale differences (F_{18}) in order to measure size similarity between sets of matching SIFT keypoint descriptors. This feature is computed by:

$$F_{18} = \frac{\sqrt{\sum_{i=1}^{n}(SQ_i / SM_i)^2}}{\sqrt{n}} \qquad \text{(Equation A.25)}$$

where:

n is the number of matched keypoints in the maximum GHT accumulator.

SQ is the scale of the matched keypoint in the query image.

SM is the scale of the matched keypoint in the database image.

A.10 Mean of orientation differences between sets of matched keypoints

We propose to calculate the mean orientation differences (F_{19}) in order to measure orientation similarity between sets of matching SIFT keypoint descriptors. This feature is calculated by the following.

$$F_{19} = \frac{\sum_{i=1}^{n} abs(OQ_i - OM_i)}{n} \qquad \text{(Equation A.26)}$$

where:

n is the number of matched keypoints in the maximum GHT accumulator entry.

OQ is the orientation of the matched keypoint in the query image.

OM is the orientation of the matched keypoint in the database image.

A.11 Median of orientation difference between sets of matched keypoints

We propose to calculate the median orientation difference (F_{20}) in order to measure orientation similarity between sets of matching SIFT keypoint descriptors. If n is an even number then this feature is calculated by:

$$F_{20} = \frac{abs(OQ_{(n/2)} - OM_{(n/2)}) + abs(OQ_{(n+1)/2} - OM_{(n+1)/2})}{2} \quad \text{(Equation A.27)}$$

If n is an odd number then this feature is calculated by:

$$F_{20} = abs(OQ_{(n+1)/2} - OM_{(n+1)/2}) \quad \text{(Equation A.28)}$$

where:

n is the number of matched keypoints in the maximum GHT accumulator entry.

OQ is the orientation of the matched keypoint in the query image.

OM is the orientation of the matched keypoint in the database image.

A.12 RMS of orientation difference between sets of matched keypoints

We propose to calculate the RMS (Root-Mean-Square) orientation difference (F_{21}) in order to measure orientation similarity between sets of matched SIFT keypoint descriptors. The feature is computed by:

$$F_{21} = \frac{\sqrt{\sum_{i=1}^{n}(abs(OQ_i - OM_i))^2}}{\sqrt{n}} \quad \text{(Equation A.29)}$$

where:

n is the number of matched keypoints in the maximum GHT accumulator entry.

OQ is the orientation of the matched keypoint in the query image.

OM is the orientation of the matched keypoint in the database image.

A.13 Ratio of matched keypoints /total keypoints in query image

We propose to measure overlap proximity by calculating the ratio of matched keypoints to total keypoints in the query image. This feature (F_{22}) is calculated as folllows:

$$F_{22} = \frac{n}{nq} \qquad \text{(Equation A.30)}$$

where:

n is the number of matched keypoints in the maximum GHT accumulator entry.

nq is the total number of keypoints in the query image

A.14 Ratio of matched keypoints / total keypoints in database image

We propose to use the ratio of matched keypoints to total keypoints within the database image to measure overlap proximity. This feature (F_{23}) is calculated as follows:

$$F_{23} = \frac{n}{nm} \qquad \text{(Equation A.31)}$$

where:

n is the number of matched keypoints in the maximum GHT accumulator entry.

nm is the total number of keypoints of database image

A.15 Self similarity

The self similarity of keypoints extracted from an image is computed by averaging the relative frequency of similar keypoints in a self similarity histogram. We select 10 as the maximum number of significant self-similar keypoints, in accordance with the number of items that can be held in short-term visual memory by humans (Miller, 1956). Each of the 10 most highly ranked

keypoints will match to similar keypoints in the remainder of the set of keypoints extracted from a particular image and we count this matching frequency for the 10 most self-similar keypoints in a self similarity histogram. Our *self similarity* measure is computed by taking the mean of all points in the self similarity histogram and is summarised in Figure A.2.

Self similarity algorithm

Step 1: Compute the nearest neighbour (D_{lowest}) of each interest point and add it to the array of distances of each keypoint ($Ad_L(L_i)$).

   The D_{lowest} is described in section 5.3.2 of Chapter 5.

Step 2: Sort $Ad_L(L_i)$ by ascending order and select top l0 keypoints.

Step 3: For each selected keypoint.

   Step 3.1: Calculate the distance of selected keypoint and the remaining

      keypoints (D_i) from Equation A.6.

   Step 3.2: If the distance less than 0.2 (the self similarity threshold) then

         Increment the self similarity count for this keypoint by one.

      End

   Step 3.3: save the self similarity count for this keypoint to

      a similarity histogram(*Sim*) and continue to Step 3.1 until all

      selected keypoints have been processed

Step 4: calculate average of all similarity histograms

$$F_{24} = \frac{\sum_{i=1}^{nSim} Sim_i}{nSim} \qquad \text{(Equation A.32)}$$

   where *nSim* = 10

Figure A.2 The self similarity algorithm

A.16 Vertical Symmetry

We proposed an algorithm for computing the vertical symmetry of matching keypoints. Vertical symmetry is defined here to be the median distance from the centre distance of all matched keypoints to the *x* axis. The algorithm to calculate the vertical symmetry of matching keypoints is presented in Figure A.3.

Vertical symmetry of matching keypoints

Step 1: Calculate axis of in-plane rotation (Mukundan & Ramakrishnan, 1998). The axis rotation (*Shift_axis*) is calculated by:

$$Shift\_axis = 0.5 * tan^{-1}((2*m11) / (m20 - m02))$$ (Equation A.33)

where *m11*, *m20*, and *m02* are moments from Equation A.3.

Step 2: Rotate matching keypoints by - *Shift_axis.*

$$ry = x* sin(- Shift\_axis) + y*cos(- Shift\_axis)$$ (Equation A.34)

$$rx = x*cos(- Shift\_axis) -y*sin(- Shift\_axis)$$ (Equation A.35)

where (*x*, *y*) is the matched keypoint.

Step 3: Calculate the centre of rotated keypoints by mean of all rotated keypoints.

$$cx = \frac{\sum_{i=1}^{n} rx}{n}$$ (Equation A.36)

$$cy = \frac{\sum_{i=1}^{n} ry}{n}$$ (Equation A.37)

where *n* is the number of matching keypoints

Figure A.3 The algorithm used to calculate vertical symmetry of matching keypoints

Vertical symmetry of matching keypoints

Step 4: Calculate the error distance of each pair of keypoints in each axis.

Step 4.1: separate rotated keypoints to two sets

If $rx > cx$

Add this keypoint to set1 (*pset1*)

Else

Add this keypoint to set2 (*pset2*)

End

Step 4.2: calculate the error distance (D_{err}) of each keypoints in two sets

$$D_{err}(i) = \min(D_L(pset1(i), \forall pset2)) \qquad \text{(Equation A.38)}$$

where

$$D_L(p_1(x_1, y_1), p_2(x_2, y_2)) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

(Equation A.39)

Step 5: Calculate the median of all error distances in Step 4. The vertical symmetry (F_{25}) is the median of the error distances to the x axis.

$$F_{25} = Med(D_{err}) \qquad \text{(Equation A.40)}$$

where *Med*() is the median function and is calculated by Equation A.27 and A.28.

Figure A.3 The algorithm to calculate vertical symmetry of matching keypoints (continued)

A.17 Horizontal Symmetry

We proposed an algorithm for computing the horizontal symmetry of matching keypoints. Horizontal symmetry is defined here to be the median distance from

the centre distance of all matched keypoints to the *y* axis. The algorithm to calculate the horizontall symmetry of matching keypoints is presented in Figure Figure A.4.

Horizontal symmetry of matching keypoints

Step 1: Calculate axis of in-plane rotation by Equation A.33
 	
 Step 2: Rotate matching keypoints by Equations A.34 and A.35
 	
 Step 3: Calculate the centre of rotated keypoints by mean of all rotated keypoints by Equations A.36 and A.37.
 	
 Step 4: Calculate the error distance of each pair of keypoints in each axis.
 	
     Step 4.1: separate rotated keypoints to two sets
 	
         If $ry > cy$
 	
             Add this keypoint to set1 (*pset1*)
 	
         Else
 	
             Add this keypoint to set2 (*pset2*)
 	
         End
 	
     Step 4.2: calculate the error distance (D_{err}) of each keypoints in two sets by Equation A.38.
 	
 Step 5: Calculate the median of all error distances in Step 4. The horizontal symmetry (F_{26}) is the median of the error distances to the *y* axis.
 	
     $F_{26} = Med(D_{err})$ (Equation A.41)
 	
     where *Med*() is the median function and is calculated by Equation A.27 and A.28.

Figure A.4 The algorithm to calculate horizontal symmetry of matching points

A.18 The average of multi-peak GHT scores

Trademark images can typically containing multiple components that may themselves be self-similar and also transformed independently in terms of their relative positions, orientations and scales when attempting to match the local features of such types of image, multiple GHT peaks are generated when their matching keypoints are projected into Hough space. This phenomenon is produced by the matching keypoints from each corresponding component generating its own peak (matched keypoint cluster) in Hough space. Many researchers suggest that only the two or three dominant components which characterise an object are required to recognize that object (Biederman, 1987; Kirkpatrick, 2001). Hence, we only consider three maxima of the GHT accumulator, corresponding to three components which may now have been independently (2D affine) trasnformed, to account for the dominant (in terms of numbers of matching keypoints) three matching sub-groups. By computing the average of the multi-peak GHT scores we are able to generate a summary score based on the best three matching image components. We calculated by the average score of the first three maxima of GHT accumulator as follows:

$$F_{27} = \frac{\sum_{i=1}^{npeak} Score(i)}{npeak} \qquad \text{(Equation A.42)}$$

where $npeak = 3$.

$$Score(i) = \frac{\sum_{j=1}^{ni} D_{lowest}(j)}{ni} \qquad \text{(Equation A.43)}$$

where ni is the number of matched keypoints in the ith row of the GHT accumulator.

D_{lowest} is calculated by Equation 5.11 (see section 5.3.2 for details).

Appendix B

Validation of proposed meta-features

We explain the experimental framework for validating perceptual grouping in the system according to appearance properties and Gestalt properties.

We now investigate the ability of the meta-features implemented to measure visual appearance properties in the system. This section presents hypotheses, objectives, and research questions.

We have to consider several questions, including:

1. Of the 17 meta-features designed to measure visual appearance properties that have been implemented, how many visual appearance properties can the system represent?

2. What is the most appropriate meta-feature to capture each appearance property?

To answer these questions, we utilize the implementation described in the following sections.

B.1 Implementation methods

The experiments were separated into nine experiments designed to validate each of the nine appearance properties expressed in isolation by each test image set. There are 9 properties to be investigated: global rotation, global pattern similarity, global pattern overlap, spread of the matched points, structural configuration of the matched points, scale difference between matched patterns, self similarity, symmetry, and sub-component similarity. We select meta-features to represent each appearance property tested in each experiment as listed in Table B.1.

Table B.1 Appearance properties investigated and proposed meta-features

Experiment number	Appearance property	Proposed meta-features
1	Global rotation	1, 10 and 11
2	Global pattern similarity	2, 6 and 7
3	Global pattern overlap	12 and 13
4	Spread of the matched points	3 and 4
5	Structural configuration of the matched points	5
6	Scale difference between matched patterns	8 and 9
7	Self similarity	14
8	Symmetry	15 and 16
9	Sub-component similarity	17

In the experiment, we tested the utility of the appearance properties implemented in the system. The experiment verified which appearance properties in the system are capable of being represented, and which meta-feature is most appropriate for measuring each appearance property.

The test image sets were each defined to vary between degrees of a specific visual property such as rotation or similarity. In each experiment, we found the dominant meta-feature which best correlates with the varying degrees of the appearance property expressed by each of the test images within a set, and the associated meta-feature values extracted from each of the test images. We then compared the result of proposed (i.e. as hypothesised) best meta-features and those meta-features observed to correlate best. The results reveal the correspondence between the percentage of proposed meta-features proposed and the best correlating meta-features for each appearance property. We have taken the number of appearance properties that the system is able to express to correspond to properties supported by meta-features that both yield the best correlation performance that also corresponds to the appearance property for which it was designed. As the result, we can summarize the utility of the appearance properties in the system. We describe the experimental procedure in the following section.

B.2 Experimental procedure

In this section, we explain the experimental procedure for setting up the experiments for evaluating the system properties (appearance properties and

Gestalt properties). We show the test images in section B.2.1, describe experimental processes in section B.2.2, and explain the measurement method in section B.2.3.

B.2.1 Test images

The following experiment used nine different sets of test images, each set designed to express a different appearance property under investigation. Each individual test set comprises five images, where the degree of each appearance property within each sequence of test images is progressively reduced. The test sets are shown in Figures B.10-B.20, each set depicting a different appearance property for each experiment. In each test image set the first image (a) is used as a reference image from which a set of meta-features are extracted by comparison with each of the remaining images in the test set. Under ideal conditions, the first image (a) would correlate perfectly with itself and then correlate progressively (linearly) less strongly with each subsequent image, (b)-(e), in the remainder of the test set (Ahmad and Ibrahim, 2006).

Experiment 1: We evaluate the ability of the system to estimate the relative global rotation differences between reference and remaining test images by calculating the correlation between the meta-feature values and the degrees of global rotation of the test images. The set of global rotations comprise: 0 (reference), 10, 20, 30, and 40 degrees respectively, Figure B.1.

(a) (b) (c) (d) (e)
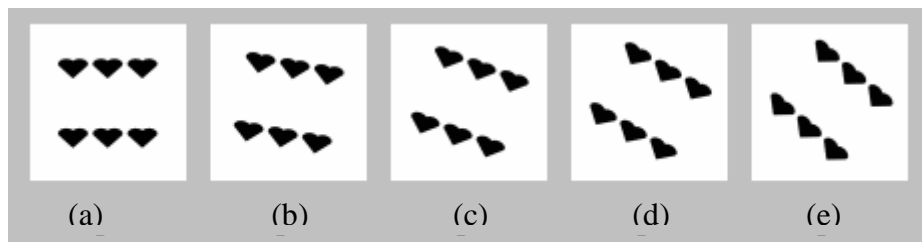
Figure B.1 Test images in experiment 1 (Global rotation).

Experiment 2: We aim to determine the ability of the system to evaluate global pattern similarity by correlating the meta-feature values with a progressively decreasing degree of global pattern similarity.

Global pattern similarity is defined in terms of the number of similar sub-components shared by compared images. Therefore, the test set in experiment 2 is

arranged according to the number of modified components in each of the test images, and comprises 0, 1, 2, 3, and 4 component modifications sequentially, Figure B.2.
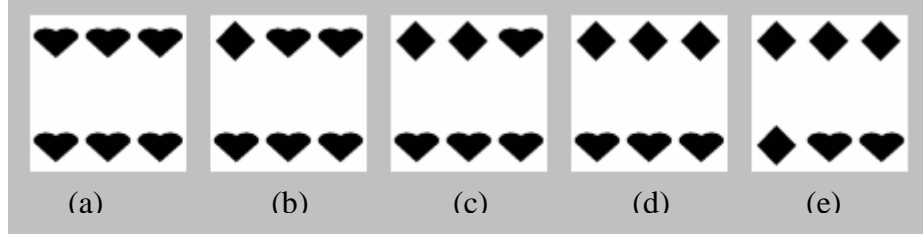
Figure B.2 Test images in experiment 2 (Global pattern similarity).

Experiment 3: We aim to evaluate the ability of the system to evaluate the global pattern overlap between compared images by finding the correlation between the meta-feature values and the degrees of pattern overlap of the test images. Accordingly, the test set is arranged in order of the number of common components, in this case 0, 1, 2, 3, and 4 components consecutively, Figure B.3.
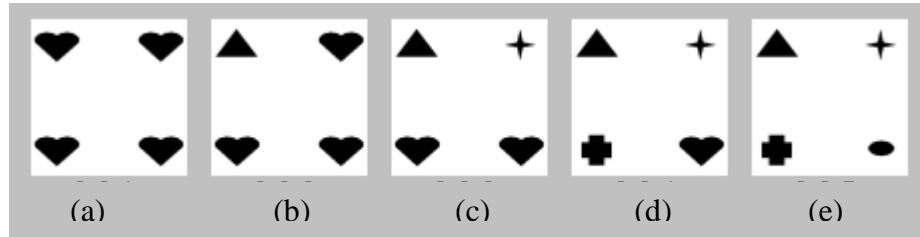
Figure B.3 Test images in experiment 3 (Global pattern overlap).

Experiment 4: The relative proximities, or spread, of components in compared images is evaluated by correlating the meta-feature values to the degree of spread expressed in the test image set. Accordingly, the test image set in experiment 4 is arranged according to the distance between a pair of components in the test images. The distances of the two components depicted are: 25, 50, 75, 100, and 125 percent of image width. The test set in experiment 4 is shown in Figure B.4.
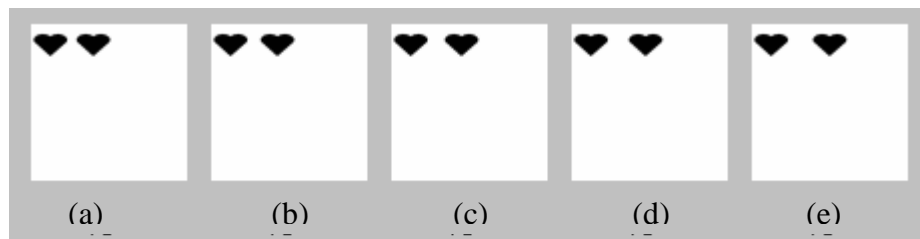
Figure B.4 Test images in experiment 4 (Spread of the matched points).

Experiment 5: The configuration similarity of sub-components within compared images is evaluated by correlating the meta-feature values to the degree of relative deformation of sub-component configurations expressed in the test image set. This test image set is arranged according to the distance apart of two components depicted in this set and these distances comprise 86.5, 66.5, 46.5, 26.5, and 6.5% (image X, Y dimensions) in sequence, Figure B.5.

<div align="center">(a)   (b)   (c)   (d)   (e)</div>
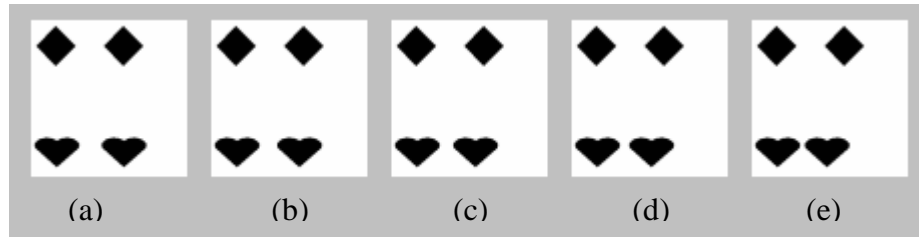
Figure B.5 Test images in experiment 5 (Structural configuration of matched points).

Experiment 6: The relative difference in scale of the sub-components within compared images is evaluated by correlating the meta-features with the degree of scale difference expressed in the test image set. The test image set is arranged according to the scale of the global patterns expressed in each image of this test set and these scales comprise factors of full size, 0.8, 0.6, 0.4, and 0.2, Figure B.6.

<div align="center">(a)   (b)   (c)   (d)   (e)</div>
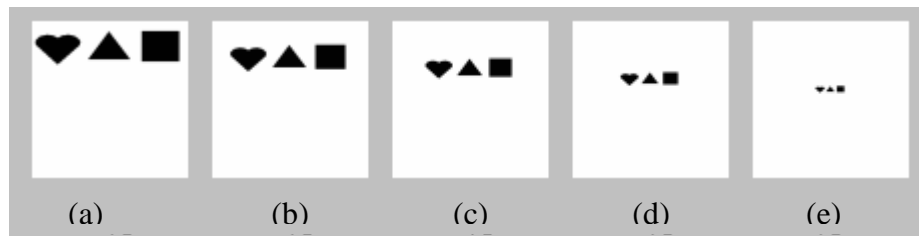
Figure B.6 Test images in experiment 6 (Scale difference of matched pattern).

Experiment 7: The relative self similarity between compared images is evaluated by correlating the meta-features with the degree of self similarity expressed in the test image set. This test image set is arranged according to the number of duplicated components depicted in each image of this test set. The numbers of duplicated components comprise 6, 5, 4, 3, and 2, Figure B.7.

<div align="center">(a)   (b)   (c)   (d)   (e)</div>
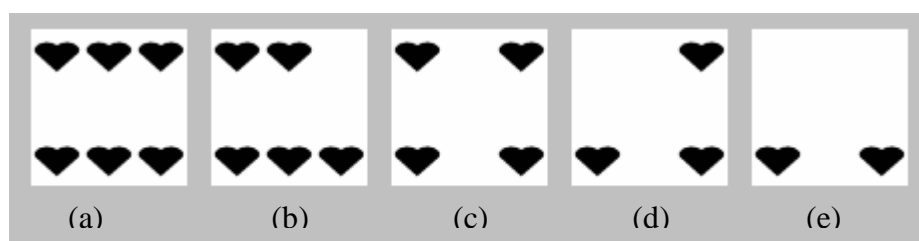
Figure B.7 Test images in experiment 7 (Self similarity).

Experiment 8: The relative symmetry between compared images is evaluated by correlating the meta-features with the degree of symmetry expressed in the test image set. This test image set is arranged according to the degree of asymmetry depicted in each image of this test set. The degrees of asymmetry comprise 0, 20, 40, 60, and 80 percent, as in Figure B.8 depicting vertical symmetry variations and Figure B.9 depicting horizontal symmetry variations.
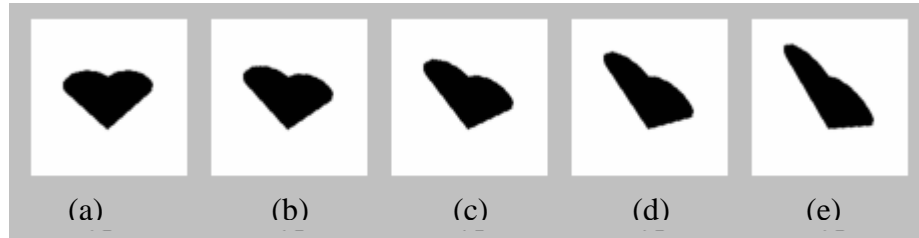
Figure B.8 Test images in experiment 8 (Vertical symmetry).
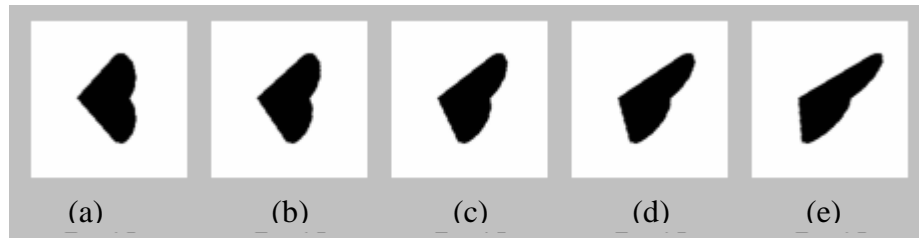
Figure B.9 Test images in experiment 8 (Horizontal symmetry).

Experiment 9: The relative sub-component similarity between compared images is evaluated by correlating the meta-features with the degree of sub-component similarity expressed in the test image set. This test image set is arranged according to the percentage of unchanged components depicted in each image of this test set. Since each image comprises four components, the percentages of unchanged components comprise 100, 75, 50, 25, and 0, Figure B.10.
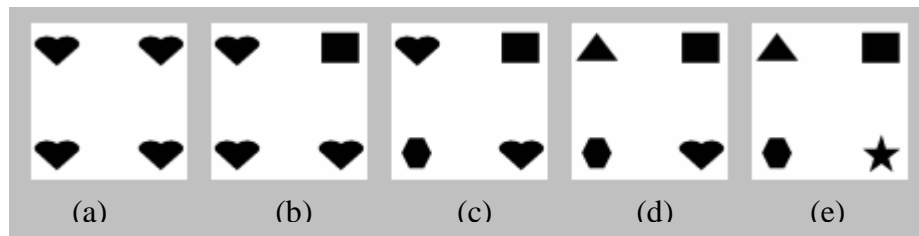
Figure B.10 Test images in experiment 9 (Sub-component similarity).

In the next section, the experimental process for all the experiments is described.

B.2.2 Experimental process

In each experiment, we use a different image set to test each individual appearance property. The appropriate test, depicting a specific appearance property under investigation, is input to the system which extracts SIFT features from each image in this set. The system then matches the SIFT descriptors extracted from the first image (a) of this test , set to the SIFT descriptors extracted from the remaining images (b)-(e) of this set using the GHT, as overviewed in Figure B.11 and detailed in section 5.3.2. The system then calculates the 17 meta-features described in section 6.1, and correlates each meta-feature value for each match to the appearance property degree in the corresponding test image.

```
┌──────────────┐                    ┌──────────────────┐
│    Test      │                    │ The first image  │
│   Images     │                    │      (a)         │
│   (b)-(e)    │                    └──────────────────┘
└──────────────┘
                  ╭─────────────────────────╮
                  │        System           │
                  │ 1. SIFT key point detection, │
                  │ 2. Calculate SIFT matching score │
                  │ 3. Calculate Meta-features │
                  ╰─────────────────────────╯
                ┌────────────────────────────┐
                │ Meta-feature vectors and their │
                │  correlation with each visual  │
                │    appearance property         │
                └────────────────────────────┘
```
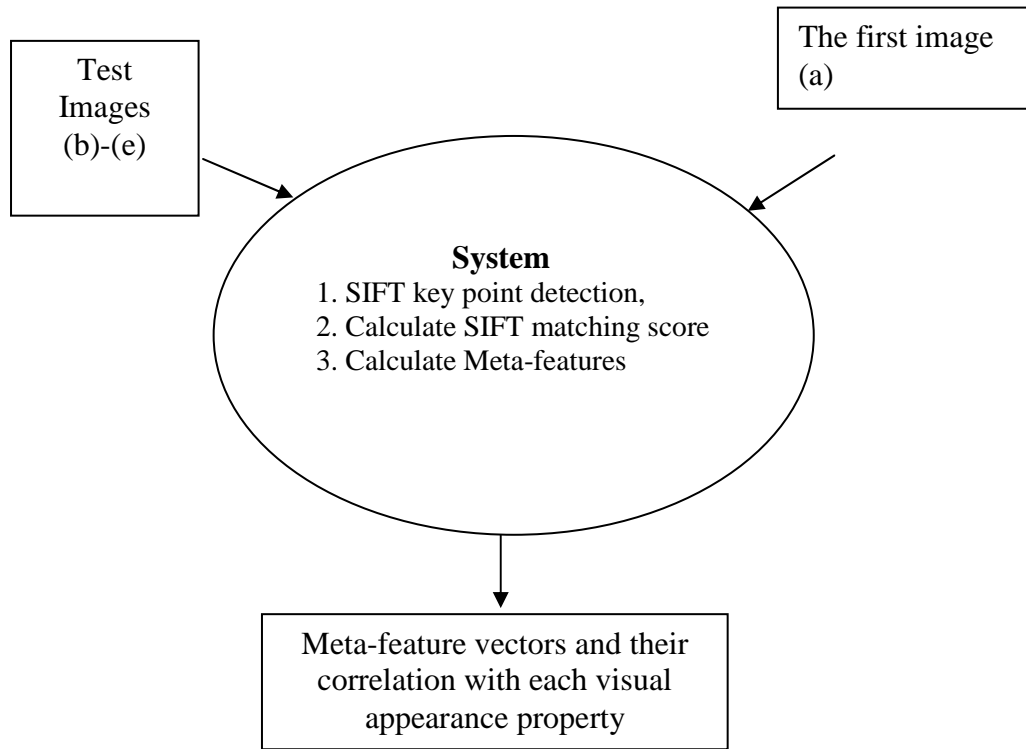
Figure B.11 The experimental process

For each experiment, dominant meta-features are selected which exhibit a high degree of correlation between the degree of appearance property expressed in the test image set and the meta-feature vector. For example, in experiment 1 meta-feature 11 was found to correlate best with the degree of rotation induced in image test sequence 1, and was therefore chosen to represent the global relative rotation appearance property. We compare the dominant meta-features with the proposed (hypothesised) meta-features (details in Table B.1) in order to select the most appropriate proposed meta-feature for each appearance property. The number of appearance properties in the system is counted by the number of the

same proposed and the appropriate proposed meta-features.

B.3 Experimental results

The correlation between meta-features and the degree of similarity of the test images is shown in Table B.2.

Table B.2 The top five highest correlation values between meta-feature values and the degree of appearance property expressed in the test images in experiments 1 to 9.

Gestalt property	Meta-feature order	1	2	3	4	5
1. Rotation	Meta-feature no.	11	10	15	6	8
	value	0.9981	0.9971	0.9277	-0.9051	-0.8770
2. Similarity	Meta-feature no.	7	12	6	13	2
	value	-0.9782	-0.9782	-0.9759	-0.9654	-0.9475
3.Overlap	Meta-feature no.	7	12	17	2	13
	value	-0.9880	-0.9014	-0.8944	-0.8341	-0.7698
4. Spread	Meta-feature no.	9	6	8	3	14
	value	-0.9880	-0.9855	-0.916	-0.8880	0.8660
5. Structural	Meta-feature no.	5	7	2	12	13
	value	-0.9799	0.9449	0.9449	0.9449	0.9196
6. Scale	Meta-feature no.	8	9	1	15	16
	value	-0.942	-0.910	-0.8395	0.8310	0.8160
7. Self	Meta-feature no.	14	2	7	12	3
	value	0.9856	0.8933	0.8932	0.8932	0.8825
8.1. V. Sym.	Meta-feature no.	16	11	10	15	14
	value	0.9801	0.9731	0.9520	0.9201	0.9192
8.2. H. Sym	Meta-feature no.	11	1	15	10	14
	value	0.9981	0.9936	0.9679	0.9663	0.9312
9. Sub component	Meta-feature no.	2	3	7	12	17
	value	0.9622	0.9497	0.9449	0.9449	0.8674

The details of the proposed meta-features versus their best correlating meta-features are shown in Table B.3.

Table B.3 The proposed and best correlating meta-features for each experiment.

Experiment number	Proposed meta-feature(s)	Best correlating meta-feature
1	1, 10, 11	11
2	2, 6, 7	7
3	12, 13	12
4	3, 4	3
5	5	5
6	8, 9	8
7	14	14
8.1	16	16
8.2	15	15
9	17	17

A proposed meta-feature and best correlating meta-feature are in congruence in experiments: 1, 2, 5, 6, 7, and 8.1, corresponding to meta-features: 11, 7, 5, 8, 14 and 16 respectively.

For experiments 3, 4, 8.2, and 9, the best meta-features based on the proposed meta-features are 12, 3, 15, and 17 respectively.

The best meta-features have similar correlation values to the highest correlation meta-features, and take each appearance property into account as described in section 6.5 and further analysis in section B.4.

B.4 Experimental Analysis

The best meta-features for experiments 1, 2, 3, 4, 5, 6, 7, 8.1, 8.2, and 9 were found from the proposed meta-features that are meta-features 11, 7, 12, 3, 5, 8, 14, 16, 15 and 17 respectively. In experiments 1, 5, 6, 7 and 8.1, each best correlating meta-feature had the highest correlation value. In experiment 2, meta-features 7 and 12 have the same correlation value. However, meta-feature 7 best reflects the global similarity because it takes similarity of the total number of matched keypoints into account. Therefore, meta-feature 7 is the better meta-feature for experiment 2. In experiment 3, meta-features 7 and 12 have similar correlation values. However, meta-feature 12 best reflects the global pattern overlap because it takes proximity property of the percentage of matched points/ total keypoints in query into account. Therefore, meta-feature 12 is the better meta-feature for experiment 3. Interactions between proximity and similarity properties are reported (Han, 2004). In experiment 4, meta-features 3, 6, 8, and 9

have high correlation values. However, meta-feature 3 best reflects the spread of the matched points because it takes proximity property of moment from matched keypoints in query into account while meta-features 6, 8, and 9 are based on similarity measurement. Therefore, meta-feature 3 is the better meta-feature for experiment 4. In experiment 8.2, meta-features 1, 11, and 15 have similar correlation values. However, meta-feature 15 best reflects the symmetry because it takes horizontal symmetry into account while meta-features 1 and 11 are based on rotation measurement. Therefore, meta-feature 15 is the better meta-feature for experiment 8.2. In experiment 9, meta-features 2, 3, 7, 12, and 17 have high correlation values. However, meta-feature 17 best reflects the sub-component similarity because it takes the average of multi-peak GHT scores into account while meta-features 1 and 11 are based on another similarity and proximity measurement. Therefore, meta-feature 17 is the better meta-feature for experiment 9.

The results show that the proposed meta-features for each appearance property are consistent according to their design function. We summarize the appearance property of each best meta-feature in Table B.4.

Table B.4 Summary of appearance properties in each best correlation meta-feature.

Experiment no.	Meta-feature no	Measurement function	Appearance property
1	11	Median of orientation difference for a pair of matched point sets	global rotation
2	7	Total number of matched keypoints	global pattern similarity
3	12	Percentage of matched points/ total keypoints in query	global pattern overlap
4	3	Moment (m02) from matched keypoints	spread of the matched points
5	5	SD of error-distance for a pair of matched keypoints	structural configuration of the matched points
6	8	Mean of scale differences between a pair of matched keypoint sets	scale different of matched pattern
7	14	Self similarity	self similarity
8.1	16	Vertical symmetry	Symmetry
8.2	15	Horizontal symmetry	Symmetry
9	17	The average of multi-peak GHT scores	sub-component similarity

In addition, the best nine meta-features are grouped into three Gestalt properties in the following list:

- Similarity is used to group similar parts of an image and can be expressed as size, orientation, pattern, shape, or value.

- Proximity is used to group connected areas or close components in an image and can be expressed by nearness, touch, overlap, or combinations.

- Simplicity is used to group multiple parts into a simple component and can be expressed by symmetry, regularity, or smoothness.

We summarise the system's appearance and Gestalt properties according to their best correlation meta-features in next section.

B.5 System properties

The system can measure nine appearance properties that can be considered to implement three forms of Gestalt grouping by using the best correlating meta-features. The experimental findings and analysis details are given in sections B.3 and B.4, and a summary of the system Gestalt properties and the appearance properties according to the best correlating meta-feature is given in Table B.5.

We summarise the validation experiment in next section.

Table B.5 Summary of appearance and Gestalt properties according to the best correlating meta-feature.

Best correlation meta-feature	Appearance property	Gestalt property
11	global rotation	similarity
7	global pattern similarity	similarity
12	global pattern overlap	proximity
3	spread of the matched points	proximity
5	structural configuration of the matched points	proximity
8	scale different of matched pattern	similarity
14	self similarity	similarity
16	Symmetry	simplicity
15	Symmetry	simplicity
17	sub-component similarity	similarity

B.6 Validation experiment summary

The validation experiments show that the system has implemented all the predicted appearance properties. In experiments 1, 2, 5, 6, 7, and 8.1, the highest and best correlation meta-features were the same. Due to coupling between the appearance properties in the images, the test image sets could simultaneously exhibit variation in more than one appearance property. In experiments 3, 4, 8.2, and 9, the best meta-features had a similar value to many of the other meta-feature correlation scores. Nine appearance properties are represented in the system, and the correlation results indicate that these meta-features may be able to measure appearance properties.

References

Abbasi, S., F. Mokhtarian, et al. (1999) Curvature scale space image in shape similarity retrieval. *Multimedia systems* 7, 467-476.

Agichtein, E., et al. (2006) Improving web search ranking by incorporating user behavior information. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 19 – 26.

Aguirre, R., M. G. Linguraru, M.A. Gonzalez Ballester (2007) *Statistical Bone Shape Analysis for Image Free Surgery LV* ,121-129.

Aguirre, W. E. (2007) *The pattern and process of adaptive radiation: lessons from a three spine stickleback adaptive radiation*. Doctoral Dissertation, Stony Brook University.

Ahmad, I., and M.T. Ibrahim (2006) Image Classification and Retrieval using Correlation. *Computer and Robot Vision, 2006. The 3rd Canadian Conference on*, 60 – 70.

Allezard N., Dhome M., and Jurie F. (2000) Recognition of 3D Textured Objects by Mixing View-Based and Model-Based Representations. *, International Conference on Pattern Recognition* 1, 960 - 963.

Alwis, S. and J. Austin (1999) Trademark image retrieval using multiple features. *The challenge of Image Retrieval research workshop*.

Ankerst, M., Kastenmüller, G., Kriegel, H.-P., & Seidl, T. (1999) 3D Shape Histograms for Similarity Search and Classification in Spatial databases. *Proceeding of the 6th International Symposium on Spatial Database*, 207 - 226.

Annand, R. (2000) Report on the community trade mark system user satisfaction survey. *International Trademark Association*.

Arntson, A. E. (2006) *Graphic Design Basics*. 5th ed. Wadsworth Publishing.

Baker, S. (1998) *Design and evaluation of feature detectors*. Doctoral Dissertation, School of Arts and Sciences, Columbia University.

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

Bandyopadhyay, S., C. A. Murthy, et al. (1995) Pattern classification with genetic algorithms. *Pattern Recognition Letters* 16, 801-808.

Bashir F., Khanvilkar S., Schonfeld D., KhokharA. (2004) Multimedia systems: content-based indexing and retrieval, *Electrical and retrieval*.

Bebis, G. N. and G. M. Papadourakis (1992) Object recognition using invariant object boundary representations and neural network models. *Pattern Recognition* 25(1), 2-44.

Ben-Av, Mercedes Barchilon, and Dov Sagi. (1995) Perceptual grouping by similarity and proximity: Experimental results can be predicted by intensity autocorrelations. *Vision Research* 35(6), 853-866.

Bentley, Jon Louis. (1975) Multidimensional binary search trees used for associative searching. *Communication ACM* 18(9), 509-517.

Bhattacharjee, S. (1999) End-Stopped Wavelets for Detecting Low-Level features. *Proceedings of SPIE* 3813, 732-741.

Bhattacharjee, S. K. and T. Ebrhimi (1999) *Image Retrieval Based on Structural Content*. Technical Report, Infoscience, EPFL.

Biederman, I. (1987) Recognition-by-Components: A theory of human image understanding. *Psychological Review* 94(2), 115-147.

Biederman, I. (1995) Visual object recognition. An Invitation to Cognitive Science. S. F. Kosslyn and D. N. Osherson, MIT Press. 2, 121-165.

Biederman, I. (2007) Recent psychophysical and neural research in shape recognition. N. Osaka, I. Rentschler, & I. Biederman (Eds.) Object Recognition, Attention, and Action. Ch. 5, 71-88.

Bovik, A. L., Ed. (2000) Handbook of image and video processing, Academic Press.

Boyer, K. L. and S. Sarkar, Eds. (2000) Perceptual organization for artificial vision systems, Kluwer Academic Publishers.

Bruce, V., P. R. Green, et al. (1996) Visual perception physiology, psychology, and ecology. Bath, UK, Psychology Press.

Bryant, J. H. (1987) USPTO's automated trademark search system. World Patent Information 9(1), 5-9.

Burges, Christopher J. C. (1998) A Tutorial on Support Vector Machines for *Pattern Recognition. Data Mining Knowledge Discovery* 2, no. 2 (1998), 121-167.

Buscher, G., et al. (2008) Eye movements as implicit relevance feedback. Conference on Human Factors in Computing Systems, 2991-2996.

Chabat, F., G. Z. Yang, et al. (1999) A corner orientation detector. Image and Vision Computing 17, 761-769.

Chakrabarti, K., et al. (2006) Ranking Objects by Exploiting Relationships: Computing Top-K over Aggregation. Proceedings of SIGMOD, 371-382.

Chan, D. Y.-M. and I. King (1999) Genetic algorithm for weights assignment in dissimilarity function for trademark retrieval. Lecture Notes in Computer Science. The Netherlands, Springer, 557-565.

Chaumette, F. (2004) Image moments: a general and useful set of features for visual servoing. *Robotics, IEEE Transactions on* 20, no. 4 ,713-723.

CIPO (2002) Table of Vienna classification codes, Canadian Intellectual Property Office.

Claus, P. (2002) Survey of the annual technical reports on the trademark information activities of industrial property offices in the year 2000. World Patent Information 24, 211-220.

Cootes, T. F., Taylor, C. J., Cooper, D., & Graham, J. (1995) Active shape models their training and application. Computer Vision and Image Understanding, 61(1), 38-59.

Datta, R., Joshi, D., et al. (2008) Image Retrieval: Ideas, Influences, and Trends of the New Age. ACM Computing Surveys, vol. 40, no. 2.

Eakin, J. P., K. J. Riley, et al. (2003) Shape feature matching for trademark image retrieval. International Conference on Image and Video Retrieval.

Eakins, J.P., K. Shields and J.M. Boardman (1996) Artisan—A Shape Retrieval System based on Boundary Family Indexing, Storage and Retrieval for Image and Video Databases IV (Proceedings of SPIE 2670), I.K. Sethi and R.C. Jain, eds., SPIE Press, Bellingham, Wash., 17-28.

Eakins, J. P. (1997) Trademark image retrieval. SIRA conference from Images to Knowledge, London.

Eakins, J. P. (2001) Retrieval of still images by content. Lectures on information retrieval, Springer-Verlag NewYork, Inc., 111-138.

Eakins, J. P. (2003) Human image perception and shape matching. To be publish.

Eakins, J. P., J. M. Boardman, et al. (1998) Similarity retrieval of trademark images. *IEEE multimedia* 5(2), 53-63.

Eakins, J. P., J. M. Boardman, et al. (1996) Retrieval of trademark images by shape feature-the ARTISAN project. *IEE Colloquium on Intelligent Image Databases*, London.

Eakins, J. P., J. D. Edwards, et al. (2001) A comparison of the effectiveness of alternative feature sets in shape retrieval of multi-component images. *Storage and Retrieval for Media Databases*, 196-207.

Eakins, J. P., M. E. Graham, et al. (1997) Evaluation of a trademark image retrieval system. *Proceedings of the 19th annual BCS-IRSG colloquium on IR research.*

Eakins, J. P., K. Shields, et al. (1996) ARTISAN - a shape retrieval system based on boundary family indexing. *Proceedings of SPIE* 2670, 17-28.

Eklund, N.H.W., and K.F. Goebel. (2006) Using Meta-Features to Boost the Performance of Classifier Fusion Schemes for Time Series Data. *International Joint Conference on Neural Networks*, 3223-3230.

Enser, Peter, and Christine Sandom. (2003) Towards a Comprehensive Survey of the Semantic Gap in Visual Image Retrieval. *Image and Video Retrieval* 2728,163-168.

Farag, W.E.; Abdel-Wahab, H. (2003) A human-based technique for measuring video data similarity. *Proceedings of the 8th IEEE International Symposium on Computers and Communication* 2, 769 - 774.

Farin, G. (1997) *Curves and surfaces for computer aided geometric design a practical guide*. San Diego, USA., Academic press.

Field, A. (2000) *Discovering Statistics using SPSS for Windows*, SAGE Publications.

Fishler, M. A. and H. C. Wolf (1994) Locating perceptually salient points on planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(2), 113-129.

Fortuna, L., G. Rizzotto, et al. (2001) *Soft Computing*, Springer-Verlag.

Freeman, W. T. and E. H. Adelson (1991) The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13(9), 891- 906.

Fukui, M., et al. (2004) Size-Independent Image Segmentation by Hierarchical Clustering and Its Application for Face Detection. *Lecture Notes in Computer Science* 3316, 686-693.

Geradts, Z. (2002) Content-Based Image Retrieval from Forensic Image Databases. *Journal of Forensic Science* 47(2), 285-292.

Geradts, Z., H. Hardy, et al. (2001) Evaluation of contents based image retrieval methods for a database of logos on drug tablets. *Proceedings of SPIE* 4232, 553-562.

Giacinto, G., and F. Roli (2005) Instance-Based Relevance Feedback for Image retrieval, *Proceedings of Advances in Neural Information Processing Systems* 17, 489-496.

Gibson, B. M., et al. (2007) Nonaccidental Properties Underlie Shape recognition in Mammalian and Nonmammalian Vision. *Current Biology* 17, 336–340.

Goldmeier, E. (1972) Similarity in visually perceived forms. *Psychological Issues* 8(1), 1-35.

Goldstein, E. B. (1999) *Sensation and perception*, ITP.

Gori, M., M. Maggini, et al. (2003) Edge-backpropagation for noisy logo

---

recognition. *Pattern Recognition* 36(1), 103-110.

Gosselin, F. and Schyns, P. G. (2001) Bubbles: a technique to reveal the use of information in recognition tasks. *Vision Research* 41, 2261–2271.

Grigorescu, S.E., Petkov, N. (2002) Comparison of texture features based on Gabor filters. *IEEE Transaction on Image Processing* 11(10), 1160-1167.

Grossberg, S., E. Mingolla, et al. (1997) Visual brain and visual perception: how does the cortex do perceptual grouping? *Trends Neurosci* 20(3), 106-111.

Gundersen, G. A. (2000) *Trademark Searching*. International Trademark Association.

Han, Shihui (2004) Interactions between proximity and similarity grouping: an event-related brain potential study in humans. *Neuroscience Letters* 367(1), 40-43.

Han, J. W. and L. Guo (2002) New image retrieval approach based on interest points. *Proceedings of SPIE* 4862, 187-197.

Haralick, R. M. (1979) Statistical and structural approaches to texture, *Proceedings of the IEEE* 67(5), 786-804.

Harris, C. and M. Stephens (1988) A combined corner and edge detector. *Proceedings of 4th Alvey Vision Conference*, 147-151.

Havaldar, P., G. Medioni, et al. (1996) Perceptual grouping for generic recognition. *International Journal of Computer Vision* 20(1/2), 59-80.

Heidemann, G. (2004) Combining spatial and colour information for content based image retrieval. *Computer vision and Image Understanding* 94(1-3), 234-270.

Horn, B. K. P. (1975) *Image intensity understanding*, AIM

Hornegger, J., Niemann, H., et al. (2000) Appearance-based object recognition using optimal feature transforms. *Pattern Recognition* 33, 209-224.

Hopfgartner,F. Jose,J.M. (2007) Evaluating the Implicit Feedback Models for Adaptive Video Retrieval. *Proceedings of the 9th ACM SIGMM International Workshop on Multimedia Information Retrieval*, 323-331.

Hsiao, S., Chou, J. (2006) A Gestalt-like perceptual measure for home page design using a fuzzy entropy approach. *International Journal Human-computer studies* 64, 137-156.

Hsu, C. and M. Shih (2002) Content-based image retrieval by interest points matching and geometric hashing. *Proceedings of SPIE* 4923, 80-90.

Hu, M. (1962) Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory* 8(2), 179-187.

INTA (2003) *Nontraditional Trademarks*. International Trademark Association. 2004.

Jaana Kekalainen. (2005) Binary and graded relevance in IR evaluations-Comparison of the effects on ranking of IR systems, *Information Processing and Management* 41(5), 1019-1033.

Jacobs, D. (2000) What makes viewpoint invariant properties perceptually salient? a computation perspective. In: Boyer, K. L. (ed), *Perceptual Organization for Artificial Vision Systems*, Kluwer Academic Publishers, 121-138.

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

Jain, A. K. and A. Vailaya (1998) Shape-Based Retrieval: A Case Study with Trademark Image Databases. *Pattern Recognition* 31(9), 1369-1390.

Jiang, H., C. Ngo, et al. (2006) Gestalt-based feature similarity measure in ttrademark database. *Pattern Recognition* 39, 988-1001.

Jordan, C. (2005) *Comparison of Blind Relevance Feedback Algorithms Using controlled queries*. Master Thesis, Dalhousie University.

Jugessur, D. and G. Dudek (2000) Local appearance for robust object recognition. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition management* 41(5), 1019-1033.

Kang, H.-B. and E. L. Walker (1994) Characterizing and controlling approximation in hierarchical perceptual grouping. *Fuzzy sets and systems* 65, 187-223.

Kekalainen, Jaana. (2005) Binary and graded relevance in IR evaluations-- Comparison of the effects on ranking of IR systems. *Information Processing and Management* 41(5), 1019-1033.

Keskustalo, H., Järvelin, K., & Pirkola, A. (2006) The effects of relevance feedback quality and quantity in interactive relevance feedback: A simulation based on user modeling. *Lecture Notes in Computer Science*, *3936*, 191-204. Berlin, Heidelberg: Springer.

Kimia, B. B., I. Frankel, et al., Eds. (2000) *Euler spiral for shape completion. Perceptual organization for artificial vision systems*, Kluwer Academic Publishers.

Kirkpatrick, K. (2001) Object recognition. In: Cook, R. G. (ed.), *Avian visual cognition* [On-line]. Available: www.pigeon.psy.tufts.edu/avc/kirkpatrick.

Kotoulas, L., I. Andreadis (2005) Image analysis using moments. *Proceedings of ICTA'05*, 360–364.

Kruger, N. and F. Worgotter (2002) Multi-modal Estimation of Collinearity and Parallelism in Natural image sequence. *Computation in Neural* Systems 13(4), 553-576.

Lee, D. H. and S. K. Jung (2001) Delaunay Triangles for Image-Based Motion Retargeting. *Deformable Avatars*, Kluwer Academic Publishers, 158-168.

Levine, M. D. (1985) *Vision in man and machine*, McGraw-Hill.

Li, Guoliang, Jianhua Feng, Feng Lin, and Lizhu Zhou. (2008) Progressive Ranking for Efficient Keyword Search over Relational Databases. *Sharing Data, Information and Knowledge* 5071,193-197.

Liu, Y., Zhang, D., et al. (2007) A survey of content-based image retrieval with high-level semantics. *Pattern Recognition* 40, 262-282.

Logothetis, N.K., and Sheinberg, D.L. (1996) Visual object recognition. *Annual Review of Neuroscience* 19, 577-621.

Loncaric, S. (1998) A survey of shape analysis techniques. *Pattern Recognition* 31, 983-1001.

Long, F., H. Zhang, et al. (2003) Fundamentals of content-based image retrieval. *Multimedia Information Retrieval and Management*.

Loupias, E. and N. Sebe (1999) *Wavelet-based salient points for image retrieval*. Research Report, Laboratoire Reconnaissance de Formes et Vision, INSA Lyon.

Lowe, D. G. (1985) *Perceptual organization and visual recognition*, Kluwer Academic Publishers.

Lowe, D. G. (1999) Object recognition from local scale-invariant features. *Proceedings of the international conference on computer vision*, 1150-1157.

Lowe, D. G. (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision* 60(2), 91-110.

Machado, A. M. C., Marinho, C. N. J. (2003) An image retrieval method based on factor analysis. *SIBGRAPI*, 191-196.

Makhoul, J., et al. (1999) Performance Measures for Information Extraction. *Proceedings of DARPA Broadcast News Workshop,* 249-252.

Manikandan, S., Rajamani, V. (2008) A Mathematical Approach for Feature Selection & Image Retrieval of Ultra Sound Kidney Image Databases. *European Journal of Scientific Research* 24(2), 163-171.

Manjunath, B. S., Ohm, J. (2001) Color and Texture Descriptors. *IEEE Transactions on Circuits and Systems for Video Technology.*

Manning C., Raghavan P. and Schütze H. (2008), *Introduction to Information Retrieval*, Cambridge University Press.

Marshall, S. (1989) Review of Shape coding techniques. *Image and Vision Computing* 7(4), 281-294.

Martínez-Contreras, Francisco, Carlos Orrite-Uruñuela, and Jesús Martínez-del-Rincón. (2009) AdaBoost Multiple Feature Selection and Combination for Face Recognition. *Pattern Recognition and Image Analysis* 5524, 338-345.

Murtagh, F. (1983) A survey of recent advances in hierarchical clustering algorithms. *The Computer Journal* 26, 354-359.

Mehrotra, R. and J. E. Gary (1995) Similar-shape retrieval in shape data Management. *IEEE computer* 28(9), 57-62.

Mehtre, B. M., M. S. Kankanhalli, et al. (1997) Shape measures for content based image retrieval: a comparison. Information processing and management 33(3), 319-337.

Mikolajczyk, K. (2002) *Detection of local features invariant to affine transformations*. Ph.D. thesis, Institut National Polytechnique de Grenoble, France.

Mikolajczyk, K. and C. Schmid (2003) A performance evaluation of local descriptors. *IEEE Conference on Computer Vision and Pattern Recognition*, 257-264.

Mikolajczyk, K. and C. Schmid (2005) A Performance Evaluation of Local Descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27(10), 1615-1630.

Miller, G. A. (1956) The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *Psychological Review* 63, 81-97.

Ming, C. Y. (1999) *Shape-based image retrieval in iconic image database*. Master's thesis, Chinese University of Hong Kong

Mitchell, T. (1997) *Machine Learning*, McGraw-Hill.

Mohan, R. and R. Nevatia (1992) Perceptual organization for scene segmentation and description. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14(6), 616-634.

Mokhtarian, F. and F. Mohanna (2001) Enhancing the Curvature Scale Space corner detector. *Proceedings of Scandinavian conference on Image analysis*, 145-152.

Mokhtarian, F. and R. Suomela (1998) Robust Image Corner Detection trough Curvature Scale Space. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20(12), 1376-1381.

Mukundan, R. and K. R. Ramakrishnan (1998) *Moment Functions in Image Analysis: theory and applications*, World sciencetific Publishing.

Murray, S. O., D. Kersten, et al. (2002) Shape perception reduces activity in human primary visual cortex. *Proceedings of Neuroscience* 99(23), 15164-15169.

Othman, A. and Martinez, K. (2008) Colour appearance descriptors for image browsing and retrieval. *SPIE Electronic Imaging: Multimedia Content Access: Algorithms and Systems*.

Parker, J. R. (1997) *Algorithms for image processing and computer vision*, John Wiley & Sons, Inc.

Paay, J. and Kjeldskov, J. (2007) A Gestalt Theoretic Perspective on the User Experience of Location-Based Services. *Proceedings of the 19th Australasian conference on Computer-Human Interaction: Entertaining User Interfaces*, 283 – 290.

Paquet, Eric, Marc Rioux, Anil Murching, Thumpudi Naveen, and Ali Tabatabai. (2000) Description of shape information for 2-D and 3-D objects. *Signal Processing: Image Communication* 16(1-2), 103-122.

Petrou, M. and P. Bosdogianni (1999) *Image Processing: the fundamentals*, John Wiley & Sons, Ltd.

Prokop, R.J. and A.P. Reeves (1992) A survey of moment based techniques for unoccluded object representation. *Graphical Models and Image Processing* 54 (5), 438-460.

Ravela, S. and R. Manmatha (1999) *Multi-Modal Retrieval of Trademark Images Using Global Similarity.* Technical Report, University of Massachusetts, Amherst.

Rodriguez, J., and F. Perronnin (2008) Local Gradient Histogram Features For Word Spotting Unconstrained Handwritten Documents, *ICFHR*, Montreal, Canada, 7-12.

Rome, E. (2001) Simulating Perceptual clustering by Gestalt principles. *25th Workshop of the Austrian Association for Pattern Recognition*. 191-198.

Rui, Y., T. S. Huang, et al. (1997) Image retrieval: past, present, and future. *Proceedings of international Symposium on Multimedia Information processing*.

Safar, M., C. Shahabi, et al. (1999) Image retrieval by shape: a comparative study. *IEEE International Conference on Multimedia and Expo*.

Sarkar, S. and K. L. Boyer (1994) *Computing perceptual organization in computer vision*, World scientific publishing.

Saund, E. and Mahoney, J. (2004) Perceptual Support of Diagram Creation and Editing, *Lecture Notes in Computer Science* 2980, 424-427.

Scassellati, B., S. Alexopoulos, et al. (1994) Retrieving images by 2D shape: a comparison of computation methods with human perceptual judgments. *Proceedings of SPIE and Retrieval for image and vision database II*, 2-14.

Schmid, C. and R. Mohr (1997) Local gray value invriants for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19(5), 30-535.

Schmid, C., R. Mohr, et al. (2000) Evaluation of Interest Point Detectors. *Internaional Journal of Computer Vision* 37(2), 151-172.

Schmid, C. e. a. (2001) Evaluation of interest point detectors, NEC Research Index Report.

Schwartz, E L, R Desimone, T D Albright, and C G Gross. (1983) Shape

recognition and inferior temporal neurons. *Proceedings of the National Academy of Sciences of the United States of America* 80(18), 5776 -5778.

Sebe, N. and M. S. Lew (2003) Comparing salient point detectors. *Pattern Recognition Letters* 24, 89-96.

Sebe, N. a. L., M. (2001) Salient points for content-based retrieval. *Proceedings of 12th British Machine Vision Conference*, 401-410.

Shafer, G. (1976) A mathematical theory of evidence, Princeton University Press.

Shih, J. L. and L. H. Chen (2001) A new system for trademark segmentation and retrieval. *Image and vision computing* 19, 1011-1018.

Sluzek, A. (2005) On moment-based local operators for detecting image patterns. *Image and Vision Computing* 23(3), 287-298.

Smith, L. I. (2002) *A Tutorial on Principal Components Analysis*. Available from <http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_compone nts.pdf> [Accessed 10th December 2007]

Smith, S. M. and J. M. Brady (1997) SUSAN - A new approach to low level image processing. *International Journal of Computer Vision* 23(1), 45-78.

Srikanth, R., R. George, et al. (1995) A variable-length genetic algorithm for clustering and classification. *Pattern Recognition Letters* 16, 789-800.

Stanchev, P. L. (2001) Content-based image retrieval systems. *Bulgarian computer science conference*, Sofia, Bulgaria.

Tahri, O. and Chaumette, F. (2005) Point-based and Region-based Image Moments for Visual Servoing of planar objects. *IEEE transactions On Robotics*, vol. 21(6), 1116-1127.

Tanaka, Keiji. (1996) Inferotemporal Cortex and Object Vision. *Annual Review of Neuroscience* 19, no. 1 (1996), 109-139.

Tao, Y. and W. I. Grosky (1998) Delaunay triangulation for image object indexing: A novel method for shape representation. *Proceedings of SPIE* 3656, 631-642.

The UK Patent office (2001) *Applying to register a trade mark*. The UK Patent office.

The UK Patent office (2001) *The UK Patent office online trade mark search*. The UK Patent office.

The UK Patent office (2003) *A brief history of trade marks*. The UK Patent office.

The UK Patent office (2004) *Applying for an International Trade Mark*. The UK Patent office.

Todman, A. G., and E. Claridge (2000) Low-level grouping mechanisms for contour completion. *Information Sciences* 125, 19-35.

Trajkovic, M. and M. Hedley (1998) Fast corner detection. *Image and Vision Computing* 16, 75-87.

Tsai, D.-M., H.-T. Hou, et al. (1999) Boundary-based corner detection using eigenvalues of covariance matrices. *Pattern Recognition Letters* 20, 31-40.

United state patent and trademark office (2004) *Trademark Electronic search System (Tess)*. United state patent and trademark office.

van Rijsbergen, C. J. (1979) *Information retrieval*, London: Butterworths.

Vasseur, P., C. Pegard, et al. (1999) Perceptual organization approach based on Dempster-Shafer theory. *Pattern Recognition* 32, 1449-1462.

Viola, P. and Jones, M. (2001) Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on*, I-511- I-518.

Vogels, Rufin, Irving Biederman, Moshe Bar, and Andras Lorincz. (2001)

Inferior Temporal Neurons Show Greater Sensitivity to Nonaccidental than to Metric Shape Differences. *Journal of Cognitive Neuroscience* 13(4), 444-453.

Walker, E. L. and H.-B. Kang (1994) Fuzzy measures of uncertainty in perceptual grouping. *Proceedings of FUZZ-IEEE*.

Wang, J., and Chang, W. (1999) A two-stage matching scheme for effective and efficient similar shape retrieval. *IEEE Transactions on Circuits and Systems for Video technology*.

Ware, Colin, and Roland Arsenault. (2004) Frames of reference in virtual object rotation. *Proceedings of the 1st Symposium on Applied perception in graphics and visualization*, 135-141.

Wertheimer, M. (1923) *A Sourcebook of Gestalt Psychology*. New York, Humanities Press.

Winterfeldt, B. J., D. Lohnes, et al. (2002) *Historical trademarks: in use since 4,000 B.C.* INTA Bulletin Archive.

WIPO (2004) *About the Nice classification*. World Intellectual Property Organization.

WIPO (2004) *International Classification of the Figurative Elements of Marks (Vienna Classification).* World Intellectual Property Organization.

WIPO (2004) *Nice classification*. World Intellectual Property Organization.

WIPO (2004) *Vienna Classification of the Figurative Elements of Marks Fourth Edition*. World Intellectual Property Organization.

Wolf, C. (2000) *Content based image retrieval using interest points and texture features*. Technical Report, Institute of Computer Aided Automation, Vienna University of Technology.

Wolf, C., Jolion, J.-M., & Bischof, H. (2000) Histograms for Texture based Image Retrieval. *Proceedings of the OEAGM*, 169-176.

Wu, J. K., C. P. Lam, et al. (1996) Content-based retrieval for trademark registration. *Multimedia tools and applications* 3, 245-267.

Xu, X. et al. (2009) Using relevance feedback with short-term memory for content-based spine X-ray image retrieval. *Neurocomputing* 72, 2259-2269.

Yin, P.-Y. and C.-C. Yeh (2002) Content-based retrieval from trademark databases. *Pattern Recognition Letters* 23(13), 113-126.

Zhang, Junliang & Javed Mostafa. (2002) Comparing Two Approaches of Generating Interest Profiles for Information Filtering: Interest Inferred from Typical User Actions Versus Rating of Content. *CAIS Proceedings*, 192 - 203.

Zheng, X., et al. (2008) Interest point based medical image retrieval. *Lecture Notes in Computer Science* 4987, 118-124.

Zhu, P. and P. M. Chirlian (1995) On critical point detection of digital shapes. *IEEE Transactions on Pattern Analysis and Machine* Intelligence 17(8), 737-748.

Zhou, X. S., T.S. Huang (2000) CBIR: from low-level features to high-level semantics. *Proceedings of the SPIE, Image and Video Communication and Processing* 3974, 426–431.

Zhou, X. S. and Huang, T. S. (2003) Relevance feedback in image retrieval: a comprehensive review. *Multimedia systems* 8, 536-544.

Zhou, Zhi-Hua, Ke-Jia Chen, and Hong-Bin Dai. (2006) Enhancing relevance feedback in image retrieval using unlabeled data. *ACM Transaction*

Information Systems 24, no. 2, 219-244.

Zitova, B., J. Kautsky, et al. (1999) Robust detection of significant points in multiframe images. *Pattern Recognition Letters* 20, 199-206.