Title: Multi-image query content-based image retrieval

Author: Fenghui Ren

Year: 2006

Repository DOI:

Multi-image query content-based image retrieval

Feng H. Ren
University of Wollongong

Multi-Image Query Content-Based Image Retrieval

A thesis submitted in fulfillment of the

requirements for the award of the degree

Master of Computer Science

from

UNIVERSITY OF WOLLONGONG

by

Feng Hui REN

School of Information Technology and Computer Science

October 2006

Dedicated to

my parents and my Bo

Declaration

This is to certify that the work reported in this thesis was done by the author, unless specified otherwise, and that no part of it has been submitted in a thesis to any other university or similar institution.

Feng Hui REN
October 3, 2006

Abstract

Content-based retrieval is based on the premise that the similarity measures in the feature space accord well with visual perceptual similarity. Furthermore, the query-by-example paradigm assumes that the query concept is well specified by the user via the example image supplied. The inadequacy of these assumptions has led to the development of several similarity measures and visual features that capture and describe colour, texture and edge information in images. The simultaneous use of multiple features, relevance feedback and more recently and the use of multiple example images in specifying the query are attempts to improve the accuracy at which the query concept can be captured. Results obtained so far are still far from the ideal because of inadequate knowledge of the human perceptual processes and this leads to the so called "Semantic Gap".

This thesis proposes a multi-image query-by-example content-based image retrieval scheme in which the significance of the components of feature vectors (intra-level) and the significance of the selected features (inter-level) are estimated through weight computation. These weights are used in calculating the feature distances and visual similarity between the query images and the database images. The hypothesis is that by incorporating the significance of features at both levels, the weighted visual similarity measure will yield improved retrieval performance (precision and recall rates). The model of the weight estimation and assignment is developed and experiments are conducted to validate the hypothesis. On average the proposed method improved the precision and recall rates in retrieval tasks on a database of natural images.

Acknowledgements

This work would not have been carried out so smoothly without the help, assistance and support of a number of people. I would like to thank the following people:

- My supervisor, Dr. Lei Ye, for always having a moment to spare, as well as inspiring and motivating me throughout my period of study.
- My co-supervisor, Prof. Philip Ogunbona, for sharing his knowledge without reservation, and for his patient guidance.
- Masters student, Jianqiang Wang, for his great help and support with my image database collection. He collected many digital images that I used in my experiments.
- Masters student, Ling Meng, for discussing various issues and for sharing ideas. He made this period of hard work interesting and relaxing. Special thanks go to Ling for his inspiration.
- My parents, for their endless love and support in my life.
- All the people who supported my image database building work, for their kindness, images and dedication of time.
- Last but not least, I wish to thank my Bo for her boundless love, support and care throughout my degree. I will never forget those days where she took care of me when I was sick and lying in bed.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Main Trends in Image Retrieval

As the usage of digital images in fields such as advertising, education, commerce, industry and even personal albums is rapidly expanding, image retrieval has recently become an active research area in terms of swift and effective utilisation and manipulation of these digital images. Currently, there are two main trends in image retrieval: text-based and content-based.

In the text-based approach, text and alphanumeric symbols are employed to describe images and submitted to the search engine as a query. The main advantages of this approach are: 1) the speed of the search is improved by exploiting techniques such as data modelling, multidimensional indexing and query evaluation method [12]; 2) users' semantic expectations in each query can be presented explicitly and easily by keywords; and 3) it is easy to construct and manage queries by utilizing standard query languages such SQL. Several text-based image retrieval systems [4][**?**][38] have been implemented and deployed in practice. However, the drawback of these systems can be early recognized. First, the task of annotating the database is labour intensive and secondly the annotation could be highly subjective. A new approach is therefore required to overcome these drawbacks.

The content-based approach has been proposed to address the shortcomings of text-based image retrieval systems. In order to eliminate the impact of user subjectivity in manual image annotation, it is proposed that the visual characteristics of images are used to replace the text annotation in image retrieval. By employing these visual characteristics, or so-called low-level features, such as colour, shape and texture, images are represented and retrieved according to the feature vectors. Content-based approach can search and retrieve images according to their visual characteristics, and has produced very promising results.

The performance of retrieval systems is usually measured in terms of precision and recall. Precision ($P(k)$) is the ratio of number of relevant retrieval images to the total number, k, of retrieval images. Precision is an indication of the efficiency of the retrieval. Recall is the proportion of desired results retrieved within all the relevant images.

1.2 Statement of the Problem

Content-based approach to image retrieval combines computer vision and information retrieval techniques and provide an effective retrieval systems. A review of the literature shows that most content-based image retrieval systems can achieve a considerable satisfaction level, around 70 percent precision when the recall is 100 percent. However, some issues still exist.(Yu:01) Currently, the main challenges in the content-based image retrieval systems are the so called "Semantic Gap" between the low-level features and the high-level semantic concepts.

"Semantic Gap" between Low-level Feature and High-level Concept

In the content-based approach, images are represented and retrieved according to their low-level features. However, sometimes, similar visual characteristics cannot always guarantee a satisfactory retrieval result. The observation can be explained by the so-called "Semantic Gap" between the low-level feature and the high-level concept. Since the machine performs very poorly in mapping the visual feature to the perceptual concept, it cannot really "understand" the users' query in terms of semantic concept and expectation. Therefore, a visually similar result may be very different from users' requirement and expectation. Sometimes, the user has in mind a concept so abstract that he himself does not know what he wants until he sees it. At that point, he may want images similar to what he has just seen or can envision. Again, however, the notion of similarity is typically based on high-level abstractions, such as activities taking place in the image or evoked emotions. Standard definitions of similarity using low-level features generally will not produce good results. In reality, the correspondence between user-based semantic concepts and system-based low-level feature is many-to-many. That is, the same semantic concept will usually be associated with different sets of image features. Also, for the same set of image features, different user could easily find dissimilar images relevant to their needs, such as when their relevance depends directly on an evoked emotion.(Li:03) For example, by using only the colour feature for retrieval, an image of a "yellow flower" will most likely be deemed to be similar to the image of a "yellow car". Therefore, in order to improve the retrieval performance of the content-based approach, this kind of "Semantic Gap" must first be bridged.

Effective Low-level Feature Similarity Measurement

As images are represented by feature vectors in the content-based approach, the visual similarities between images are measured by calculating the distance between their corresponding feature vectors. Therefore, an effective low-level feature similarity measurement could improve retrieval performance. Both linear and non-linear similarity measures can be used to obtain the distance value. For the non-linear approach, metrics employing fuzzy, probability and statistical techniques are common, and for the linear approach, the weighted Euclidean distance or City Block distance can be used. For the linear approach in particular, the weights for the distance calculation can also be adjusted by users or applications. The issue of how to create an effective low-level feature similarity measure that combines visual characteristics and the subjectivity of human perception is a key research area in image retrieval.

Combination of Multiple Low-level Features

In the content-based approach, multiple low-level features can be combined in order to achieve a accurate retrieval result. Referring to the example mentioned earlier, if both colour and shape features are used in the retrieval process, the image of a "yellow flower" can easily be distinguished from the image of a "yellow car" because of the different characteristics of shape between a "flower" and a "car". Following this approach, several content-based image retrieval systems, that combine multiple low-level features, are proposed (section 2.3.3). However, the retrieval results from single low-level features cannot be easily combined in order to improve retrieval performance. Several reasons could be adduced for this: 1) the physical meaning of each low-level feature is unique, and therefore the process of combination should not only employ a set of mathematical operations, but also possess a physical interpretation; 2) the range of distance values among low-level features vary, therefore a suitable normalisation strategy should be applied in advance; and 3) a user's perceptual subjectivity and expectations should also be considered in the process of combining low-level features in order to satisfy both visual and semantic requirements.

1.3 Goals of the Research

A number of problems have been highlighted in the previous section. Perhaps the most important is how to capture the query concept from users. Query-by-example has been shown to be successful in the literature [109]. This paradigm allows the users to pose their query by selecting images similar to the target. In this thesis the emphasis is on how to employ multiple images in improving the query posed in a query-by-example retrieval scenario.

1.4 Contributions of this Research

The main contributions of this thesis are:

1. According to the experimental results on the real-world image database, two (Colour Structure Descriptor and Edge Histogram Descriptor) of six MPEG-7 visual descriptors are selected as the most effective descriptors for real-world image retrieval using combined multiple features.

2. By employing the multi-image query approach, a novel method to modify weights for distance measurement between images in the content-based image retrieval area is proposed. Usually, a descriptor is represented in the form of a vector and the similarity between images is the distance between the vectors. In the proposed approach, by analysing the relationships between the components of vectors from the query images, the salient features of the query images can be captured. Furthermore, by employing both the mean and the standard deviation, the weight for each component in the distance calculation is modified dynamically according to the significance of the component in the feature vector. Experimental results illustrate the effectiveness of this approach.

3. Usually, a single descriptor is not comprehensive enough to imitate a human's visual system, since a human perceives an image from all aspects of its visual characteristics, including colour, shape and texture. Therefore, multiple features are usually combined in order to improve retrieval performance. In this thesis, a novel approach to modifying features' weights for the Linear Combination Method is proposed. According to the query images submitted by a user, both the similarities and differences between the queries for each descriptor are analysed and compared. The features whose characteristics are similar within the query will be considered as significant features, while others are non-significant features. According to the extent of its significance, each feature will automatically be assigned a suitable weight value for combination. In general, the more significant a feature is, the higher the allocated combination weight value will be. Experimental results demonstrate that the proposed approach outperforms the equal weight approach.

4. A Content-Based Image Retrieval System (CBIRS) is built based on the MPEG-7 Visual Descriptors. CBIRS supports both single and multi-image query searches. CBIRS supplies five descriptors for image searching and users can also select one or more descriptors during the retrieval process according to their interests. In the case of multi-image and multi-feature based image retrieval, CBIRS can dynamically modify feature components' weights and combination weights as well. As shown in the experiments, the retrieval results are closer to users' expectations by employing weights modification as proposed.

1.5 Thesis Organization

Each chapter starts with an introduction and background material, and concludes with a summary. The contents of the chapters are summarised below.

Chapter 1 introduces the research topic, its main goals, and provides research contributions related to the thesis.

Chapter 2 is a literature review of the key concepts and major research in the area of image retrieval. From the point of view of techniques, image retrieval is subdivided into several techniques, such as text-based, single feature, multiple feature and multiple image query. For each of these techniques, a detailed introduction and evaluation are presented. Several existing content-based image retrieval systems are also discussed. Finally, a comparative analysis is presented to summarise both the differences and similarities among the techniques.

Chapter 3 introduces a new approach for weights modification. This approach is based on the multiple image query. By employing both first and second moments of the row and column vectors of the feature matrix, the relationships between features and their components are analysed and the significance of features are estimated. The weights used in the distance computation is dynamically modified. Experimental results show the improvement that can be gained through the proposed method. On average the precision ratio is increased by 10 percent comparing with constant weights approach.

Chapter 4 introduces the application of proposed methods to multi-image query CBIR system: CBIRS and Impressio. Example are presented to show how the system works. The effectiveness of the systems is then discussed.

Chapter 5 summarises the thesis and draws some conclusions. Some new directions for further work are discussed in this final chapter.

Chapter 2

Literature Review

2.1 Introduction

Recent advances in semiconductor technology have resulted in the available of high quality CMOS and CCD image sensors, and the various image capture and generation devices. At the same time, both commercial and government agencies have found the advantage of being able to capture, transfer and store vast amount of visual data. This trend has been facilitated by the ubiquitous Internet. There is also the visual data that are generated by the average consumers for personal and entertainment usage. Attending these developments is the issue how to manage the accumulated data in terms of annotation, indexing and retrieval.(Yu:02)

A picture is worth a thousand of words. Therefore, undoubtedly, the use of digital images promoted a more efficient and convenient approach for the exchange of information. However, as a result of development in information transfer and storage techniques, gigabytes of new image information are being generated, stored and transmitted on PCs and the Internet every day [35]. It is very difficult, therefore, to access and retrieve these digital images without an appropriate scheme. In order to achieve more efficient image browsing, searching and retrieval results, image retrieval research has become a central theme in digital image organisation. The challenge of how to promote a new retrieval approach or modify existing strategies to improve retrieval performance thus became the goal faced by researchers.

In Figure 2.1, an overview of the image retrieval is given. In general, the image

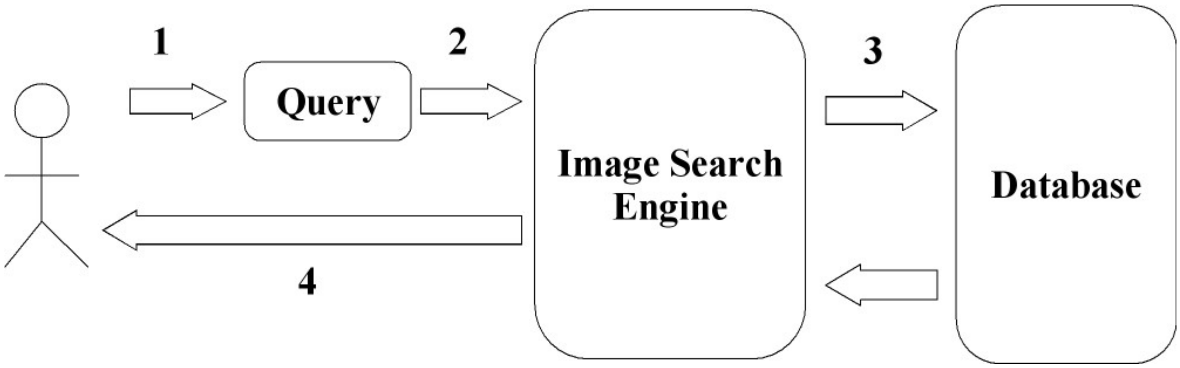


Figure 2.1: Image retrieval overview

retrieval process contains four steps as follows:

1. The user extracts and summarises his/her ideas which include both semantic and visual perception firstly, then represents them in forms of keywords [38] , Sketched queries [12], example [106] or multi-example [157] as the **Query**.

2. The **Query** is submitted to the **Image Search Engine** in forms of text [38], feature vector [2], region-of-interest [135] or multiple representations [52].

3. The **Image Search Engine**, according to the **Query** submitted by the user, searches and retrieves images in the **Database**. This **Database** can be an image database or meta data database. Different retrieval strategies could be employed in order to achieve a satisfactory retrieval performance, such as multiple feature strategy [123], neural network strategy [69] and relevance feedback strategy [112].

4. The retrieval results with different strategies are gathered and ranked as the final retrieval result and returned to the user. Methods such as linear function [49], fuzzy logic function [139] and Combining Multiple Experts (CME) [51] can be used to generate the final retrieval result.

In the following sections, some of the major approaches towards image retrieval, from well-known text-based image retrieval methods to the more recent multi-image query content-based image retrieval approach, will be introduced. Figure 2.2 depicts a categorization of image retrieval techniques that will form the basis of subsequent discussion. For each technique, the principle, some retrieval systems and evaluation are presented. Also, a detailed comparative analysis is presented to illustrate the advantages and disadvantages of the various image retrieval approaches. In the conclusion, a summary is given and the problems and gaps are highlighted.

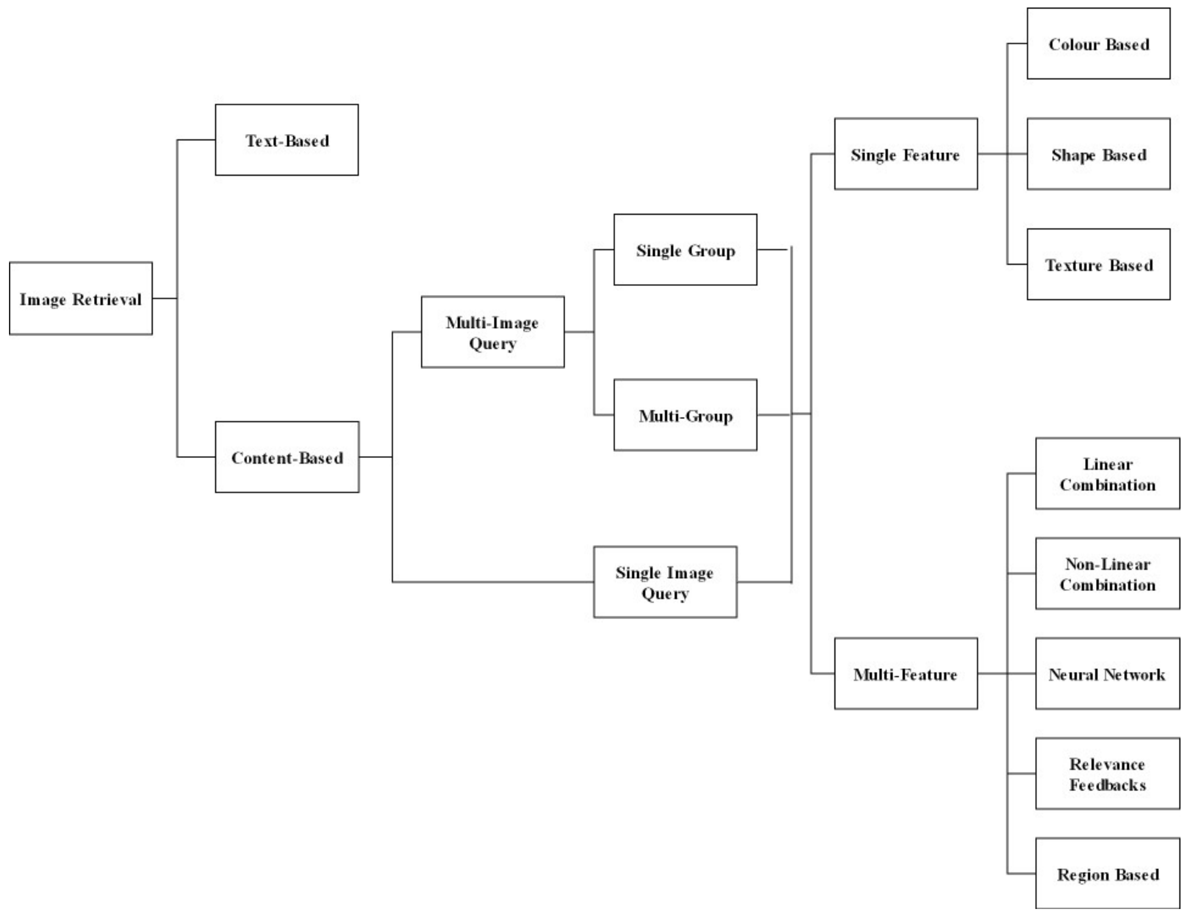


Figure 2.2: Categorization of the image retrieval approaches.

2.2 Text-Based Image Retrieval

Early image retrieval techniques were only focused on index searching or keywords matching of the text that somehow refered to the image, and were referred to as text-based image retrieval [35][105][108]. In [38], Guglielmo and Rowe developed an image retrieval system for the Naval Air Warfare Center Weapons Division of Canada. In this system, search processing was based on descriptive captions of the image. Through decomposing and analysing the text given by the user, the system could locate the captioned image in the database. Although this kind of text-based system can solve the problem of searching images according to captions in the database, it nevertheless still has limitations in terms of both the image database and the user input.

- The major limitation in terms of the image database is image annotation. Since the text-based retrieval locates the target through matching keywords, the image database is thus required to be annotated beforehand. However, when considering the effort associated with manual annotation, the problem of annotation becomes an issue with large database.

- Another issue is that the task of describing image content is highly subjective. The perspective of textual descriptions given by an annotator could be different from the perspective of the user. A picture can mean different things to different people. It can also mean different things to the same person in different circumstances. Furthermore, words used to describe content can vary from one person to another [40]. Therefore, the user is often required to remember valid words (i.e., keywords), how these keywords correlate with the concepts that he or she wishes to find, and how the keywords may be combined to formulate queries.

- According to the principles of text-based image retrieval, this kind of retrieval strategy cannot in fact be deemed as image retrieval but just a text or keywords search engine.

- Lastly, images are very rich in content, but text may be not rich enough to describe images. Therefore, the text-based image retrieval strategy cannot be expected to always achieve a satisfactory retrieval performance.

In order to overcome the drawbacks of text-based retrievals outlined above, content-based image retrieval emerged in the early 1990s as a promising means to describe and retrieve images [119].

2.3 Content-Based Image Retrieval

This section gives a detailed introduction to Content-Based Image Retrieval (CBIR). First, the structure of CBIR is introduced. Then, both the single image query and multiple image query approaches are discussed and compared. Furthermore, a number of representative content-based image retrieval systems are introduced to describe some of the basic ideas in CBIR and highlight their distinct characteristics.

2.3.1 CBIR Overview

Plesae see print copy for Figure 2.3

Figure 2.3: Overview of CBIR systems [120]

In contrast to the text-based approach, content-based image retrieval systems allow users to search for images by forming queries based on descriptors extracted from images or associated with the images. As shown in Figure 2.3, users usually have an idea of a proposed image in their mind. CBIR systems allow users to express a description of this objective image with a suitable query based on similarities in terms of semantic features, structure, and visual appearance. Typically, according to [120], CBIR systems allow users to form queries in one or more of the following ways [108][105]:

- by selecting or providing an example image;
- by graphically sketching a query image;
- by expressing the query in a structured query language such as SQL;
- by filling in fields for query-by-example.

Once a user submits a query, the system will search images in the database according to the query's descriptive information and score the images in the database in terms of their similarity to the query image. Finally, the system will rank the resulting images based on their similarity scores, such as top-k best match images [51], and return the resulting images to the user.

In content-based image retrieval (CBIR), images are described according to their visual content rather than text, using attributes such as colour, texture, and the shape of objects [128]. Features, which are used to describe the visual characteristics of images, are automatically extracted from the images by the feature extractor. Therefore, in content-based image retrieval, the strategy of text-based annotation is discarded and image retrieval can be applied to a large number of images. Another merit of content-based image retrieval is the objectivity of the retrieval process. Since subjective descriptions of images are not used and images are only described according to their visual features, the retrieval results will thus be more visually similar to the query image than the results from text-based image retrieval. Lastly, convenience is another advantage of the content-based image retrieval system. Instead of constructing redundant and perplexing keywords for retrieval precision, users only need to provide the content-based image retrieval system with a query image, and the system will ensure that it finds the images that are the most similar to the query in the database.

Nevertheless, content-based image retrieval has some drawbacks. One of them is that it is difficult to standardise the definition and extraction methods of image features. Consequently, many different feature descriptors have been developed to capture aspects of image content. This situation does not promote inter-operability between content-based image retrieval systems.

In order to standardise the representation of visual content and interwork between different retrieval systems, the ISO MPEG Group initialised the "MPEG-7 Multimedia Description Language" work item in 1997 [3]. In 2001, the MPEG-7 international standard was published, which defined standardised descriptions and description systems and allowed users or agents to search, identify, filter and browse audiovisual content [84][15]. The visual descriptors were a most important part of the MPEG-7 standard and could be divided into colour descriptors, shape descriptors and texture descriptors. By extracting these visual descriptors from images, the characteristics of images could be presented in a very easy and distinct way and images could be searched and retrieved with these descriptors [118][123][45]. The detail information about these visual descriptors are introduced in later sections.

2.3.2 Query-by-Example

Please see print copy for Figure 2.4

Figure 2.4: Intra and Inter level searching [106]

In general, CBIR contains both intra-level and inter-level search, as shown in Figure

2.4. At the intra-level, Query-by-Example (QBE) uses low level structural feature descriptions of the example image to retrieve visually similar images from the database according to a similarity metric. The performance of the system largely depends on the feature selected and extracted from the images and the distance metric used to measure the similarity. The choice of the feature used for similarity retrieval should reflect the underlying user expectation in the query. Therefore, a correct representation which can express the users' expectations and a suitable parameter set for the similarity calculation is very important. Intra-level search involves comparing each component in two images to find the similar feature vector, the result indicating the extent of the similarity between these two images. However, an intra-level search alone is not enough. In many cases, users' expectations cannot appropriately be articulated with only one feature because this can only represent one aspect of the image's property. Therefore, an inter-level search is proposed. In order to improve the performance of retrieval systems, multiple features are proposed to be used in the retrieval process and individual results are combined together to obtain the final results. An inter-level search means searching images through all the selected low-level features. Different visual feature vectors, such as colour, shape and texture, are extracted from the image. Through comparing these feature vectors, images that hold similar characteristics are found as a final result[106].

In the next subsection, approaches to content-based image retrieval are subdivided into several branches and introduced separately. First, depending on the number of images in the query, the single image query approach and the multi-image query approach are separated into different branches in content-based image retrieval. For the single query approach, both single feature and multi-feature approaches are introduced. In the single feature scheme, the colour, shape and texture-based image retrieval approaches are evaluated, and in the multi-feature scheme, several approaches are also introduced to combine multi-features' results in order to achieve a better retrieval performance. These methods include linear and non-linear combinations, the neural network approach, the relevance feedback approach and the region-based approach. For the multi-image query retrieval branch, according to how to use the query images, both the single group and multi-group approaches are evaluated. Finally, through summarising and comparing these content-based image retrieval systems, both the advantages and disadvantages of each of them are revealed, respectively.

2.3.3 Single Image Query Approach

Depending on the number of features employed in the system, the single feature and multi-feature approaches are introduced respectively. In this subsection, content-based image retrieval with the single image query approach is reviewed.

Single Feature Approach

In this type of content-based image retrieval system, features are used separately. In the following subsections, some typical content-based image retrieval systems with single features, such as colour, shape and texture, are introduced and evaluated, respectively.

Colour Feature

We will introduce some colour-based CBIR systems, since colour is the most popular and widely-used feature in CBIR [119][105][108]. In [143], Wang utilised the Dominant Colour Descriptor (DCD) in the MPEG-7 visual descriptor [84] to extract the major colours from images. Since the DCD can only represent images according to their prominent colours, each image is described by a small number of dominant colour values and their statistical properties, including percentage and variance [84]. By comparing the value and percentage of each dominant colour between images, the degree of similarity between the query image and to-be-judged images in the database could be estimated. The advantage of this system is that images are retrieved according to users' perception of colours by employing the dominant colours in images. The disadvantage lies in the lack of consideration for the location of each dominant colour. Hence, images with different colour layouts could be considered as similar. In order to overcome this drawback, some descriptors which can provide information about spatial colour distribution are employed in some other colour-based image retrieval system.

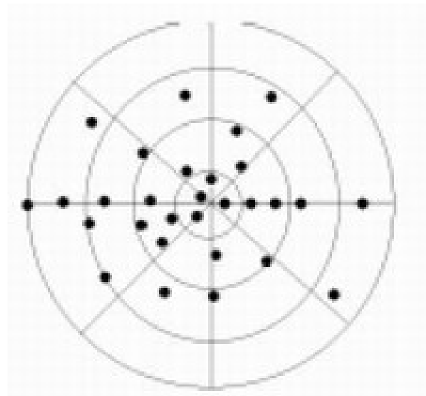


Figure 2.5: An example of an inner annular region
The distribution density vector is calculated by counting the number of points in each sector, from the inner annular region to the outer ones and within each annular region. The vector is: (1, 1, 0, 1, 1, 0, 0, 0, 2, 2, 1, 1, 3, 0, 2, 0, 1, 1, 1, 2, 2, 1, 1, 0, 1, 0, 0, 2, 1, 0, 0, 1). [104]

Rao [104] claimed that pixels could be divided into different regions according to their location. Firstly, a centre pixel is selected and several inner annular regions are generated surrounding the centre point. Then those inner annular regions are homogeneously divided into sectors. In order to obtain an image's spatial information, this system counts the number of pixels in each sector and gathers them to generate the spatial feature vector. An example of an inner annular region and feature vector extraction is displayed in Figure 2.5. Although, this proposed approach could describe the spatial information of pixels with different colours, without an efficient region partition and centre pixel location algorithm, however, the retrieval performance of this CBIR system cannot be guaranteed. For example, if the query image is polychrome and different colours are mixed together, the retrieval performance will be very low.

In [90], Beng tried to extract information about the spatial distribution of colour by dividing the image into different single-coloured clusters. For any two to-be-judged images, only when most clusters with the same colour overlapped in the image space could these two images be considered as similar. By judging the degree of overlap for each two clusters with a similar colour, the colour's spatial distribution is considered in the retrieval process. Although this approach takes into account both colour and spatial distribution for each colour, the issue of the number of clusters and the degree of overlap makes the system performance very unstable. The performance of the two approaches in [104] and [90] indicates the need for an efficient region or cluster partition method.

Please see print copy for 2.6

Figure 2.6: An example of 2×2 grid [124]

Instead of dividing images into single-coloured clusters, Stehling [124] divided images into fixed grids to solve the issue of cluster partition. For each of the grids, the local colour histogram is extracted together with the grid index. In the retrieval process, the colour histogram and index of the grids were compared to judge similarity. In Figure 2.6, an example image divided into 2×2 grids is shown, the index for each grid being [0,0], [0,1], [1,0] and [1,1] (from left to right, top to bottom). According to the similarity measurement, the left image is not considered similar to the right one, because of the different colour histogram in grid [0,1]. Although this system introduced a very efficient way to partition images and extract spatial information about colour, it does not, however, consider the size of the image when partitioning images. For example, the system would perform badly if the dimensions of images in the database spanned a broad range.

Huang et al. [43] solved this problem by restricting the size of grid but not the number, and used 3×3 or 5×5 blocks of pixels. In order to create a comprehensive representation of colour spatial information, their system extracted both the local spatial moment histogram and the local directional difference unit histogram from images. The former histogram describes the colour distribution in the image through computing the mean and the standard deviation of the pixel block, while the latter histogram is used to capture salient changes in different spatial directions in an image. In the retrieval process, one colour histogram and two colour spatial histograms are gathered to search images similar to the query. According to the experimental results, this approach produced a better performance than other colour-spatial based CBIR systems.

In [136], Tico et al. claimed that the traditional method of colour histogram creation results in quite a large number of bins with trivial colour differences between adjacent bins. Therefore, a new method of colour histogram creation is proposed. In contrast to the traditional method, the proposed approach subdivides a colour space (e.g. RGB, HSV colour space [83]) into a certain number of bins and then counts the number of pixels each bin contains. This proposed method is based exclusively on both the hue component and the intensity component in the achromatic image region. The colour appearance of the image is described using a relatively small number of bins.

Please see print copy for Figure 2.7

Figure 2.7: Two iso-colour planes with differing amounts of structure [118]

The colour structure descriptor (CSD) represents an image by both the colour distribution of the image (similar to a colour histogram) and the local spatial structure of the colour [118]. The additional information about colour structure makes the descriptor sensitive to certain image features which the colour histogram is blind. In Figure

2.7, we display a pair of images, each of them consisting of two *iso-colour planes*[1]. The left image is highly structured, whereas the right one is less so. The *structure* of an iso-colour plane is the degree to which its pixels are clumped together relative to the scale of an associated structuring element. Each image contains exactly 12 pixels in its black plane and 13 pixels in its white plane. Hence, they are indistinguishable, based solely on the information in their two-bin colour histogram. However, their two-bin CSD descriptors are very different and thus the images can easily be distinguished in an indexing or retrieval application based on the CSD. The CSD is identical in form to a colour histogram but is semantically different. Specifically, the CSD is a 1D array of 8 bit-quantised values,

$$CSD = \overline{h}_s(m), m \in \{1, \ldots, M\} \tag{2.1}$$

where M is chosen from the set $256, 128, 64, 32$ and where s is the scale of the associated square structuring element.

Pleases see print copy for Figure 2.8

Figure 2.8: HMMD colour space [118]

Since the CSD is generated in the HMMD (Hue-Max-Min-Diff) colour space (Figure 2.8), we will first briefly introduce this colour space. The HMMD colour space is closer

[1]An image quantised to N colour is composed of N iso-colour planes. The n^{th} plane is the set of all pixels having the n^{th} quantised colour, $n \in [1, N]$.

to a perceptually uniform colour space. The transform equation between RGB and HMMD can be formulated as follows [118]:

$$Max = max(R, G, B), \tag{2.2}$$

$$Min = min(R, G, B), \tag{2.3}$$

$$Diff = Max - Min, \tag{2.4}$$

$$Sum = (Max + Min)/2, \tag{2.5}$$

$$Hue = \begin{cases} 0, & \text{Max == Min;} \\ 60 * (G - B)/Diff, & \text{Max==R \&\& G} \geq \text{B;} \\ 360 + 60 * (G - B)/Diff, & \text{Max==R \&\& G} < \text{B;} \\ 60 * (2 + (B - R)/Diff), & \text{Max==G;} \\ 60 * (4 + (R - G)/Diff), & \text{Otherwise.} \end{cases} \tag{2.6}$$

Therefore, a total of five components are identified in this colour space. However, a set of three components, H, Max, Min or $H, Diff, Sum$, is sufficient to form the HMMD colour space and specify a colour point. The semantics of each component is distinct and is described as follows. Hue ($H \in [0°, 360°]$) specifies one colour family from another, as in red from yellow, green, blue or purple. Max ($Max \in [0, 1]$) specifies how much black colour is present, giving a flavour of shade or blackness. Min $Min \in [0, 1]$ specifies how much white colour is present, giving a flavour of tint or whiteness. Diff ($Diff \in [0, 1]$) specifies how close a colour is to pure colours, giving a flavour of tone or colourfulness. Finally, Sum ($Sum \in [0, 1]$) specifies the brightness of the colour.

Extraction of a CSD is a three-step process:

1. A 256-bin CS Histogram is extracted from an image represented in the 256 cell-quantised HMMD colour space. If the image is in another colour space, then it must be converted to HMMD and re-quantised prior to extraction.

2. If the bins of CS Histogram K ($K < 256$) is desired, then the bins are unified to obtain a K-bin CS Histogram.

3. The values of each of the K bins are nonlinearly quantised in accordance with the statistics of colour occurrence in typical consumer imagery.

In order to compute the CSD, an 8×8 grid window is used. Even though the total number of samples is kept fixed at 64, the spatial extent of the structuring element scales with the image size. The following simple rule determines the spatial extent of the structuring element (equivalently, the sub sampling factor) given the image size:

$$\rho = max\{0, \lfloor log_2 \sqrt{W \times H} - 7.5 \rfloor\}, \tag{2.7}$$

$$K = 2^{\rho}, \tag{2.8}$$

$$E = 8 \times K, \tag{2.9}$$

where

$$\begin{cases} W, H, & \text{image width and height, respectively;} \\ E \times E, & \text{spatial extent of the structuring element;} \\ K, & \text{sub-sampling factor.} \end{cases} \tag{2.10}$$

For similarity matching, the ℓ_1 distance measure $D(A, B)$ can be adopted for two image histograms A and B, as in the following equation:

$$D(A, B) = \sum_{i=0}^{K} |h_A(i) - h_B(i)|, \tag{2.11}$$

where $h_A(i)$ and $h_B(i)$ represent the normalised histogram bin values of image A and image B, respectively. K is from the set $\{256, 128, 64, 32\}$.

In addition to the descriptors and approaches mentioned above, many other colour descriptors have also been introduced in various papers, such as the HSV colour histogram [125], the region colour descriptor [128], the colour chromaticity histogram [137], the colour spatial feature histogram [43] and the fuzzy colour histogram [140] etc. The MPEG-7 visual standard also defines some colour descriptors that can express both colour and spatial information, such as the Colour Layout Descriptor(CLD) [53] and the Dominant Colour Descriptor [144][101][143].

Please see print copy for Figure 2.9

Figure 2.9: An example of a Region-Based Shape Descriptor [118]

Please see print copy for Figure 2.10

Figure 2.10: An example of a Contour-Based Shape Descriptor [118]

Shape Feature

Shape is also an important attribute of objects in images and several content-based image retrieval systems have exploited it. In MPEG-7 visual descriptors, both the Region-Based Descriptor (RBD) and the Contour-Based Shape Descriptor (CBD) [118][10] are used to describe the shape of the image. In Figures 2.9 and 2.10, two sample images of the MPEG-7 RBD and CBD are shown. Figure 2.9 gives examples of shapes that can best be described by shape regions rather than contours. Images contained in either of the sets (a) and (b) would be rated as similar and dissimilar to the ones in the remaining sets. For example, the images in set (a) would be identified as being similar and dissimilar to the ones in set (b). The images in Figure 2.10 are suitable for the contour descriptor. We can thus see the difference between these two descriptors. The former takes into account both the shape's outline and inner content, while the latter only considers the shape's contour distribution. The efficiency and application of the contour shape descriptor was well presented in [124][128].

However, most of the existing shape descriptors are usually either application dependent or non-robust, making them undesirable for generic shape description. For example, when the shape of the image is a circle, the existing shape descriptors cannot handle it. Zhang [153] proposed a generic Fourier descriptor (GFD) to overcome this drawback of existing shape representation techniques. By employing a modified polar Fourier transform (MPFT), an image is first transferred into a normal two-dimensional rectangular image in Cartesian space, because it is much easier to extract shape information from a normal two-dimensional rectangular image. Then the image's radial and angular frequency are extracted from the new image through the polar Fourier transform. Generally, for any given shape image $f(x, y)$, the MPFT is defined as follows:

$$PF(\rho, \theta) = \sum_r \sum_i f(r, \theta_i) exp[j2\Pi(\frac{r}{R}\rho + \frac{2\Pi i}{T}\theta)] \quad (2.12)$$

where $0 \leq r = \sqrt{[(x - x_0)^2 + (y - y_0)^2]} < R$ and $\theta_i = i(2\pi/T)$ $(0 \leq i < T)$; (x_0, y_0) is the centre of the mass of the shape; $0 \leq \rho < R$, $0 \leq \phi < T$. R and T are maximum value of radial and angular. The ρ and θ stand for the ρ^{th} radial and the θ^{th} angular frequency respectively. The experimental results indicated that the GFD outperforms the Zernike Moment Descriptor (ZMD), which has been proposed to MPEG-7 as a shape descriptor. The properties of Zhang's method are as follows [153]:

- It captured spectral features in both radial and circular directions;
- It was simple to compute;
- It was more robust and perceptually meaningful;
- The physical meaning of each feature is clearer.

In [102], a grid-based shape descriptor is used to extract shape features from images. For a given shape region, a grid consisting of fixed-size square cells is placed over it, so as to cover the entire shape region as shown in Figure 2.11. By assigning a "1" to a cell with at least 25% pixel coverage by the region and a "0" to the other cells, a cell map filled with "0" and "1" is constructed. Then a scan order traversal results in a binary sequence of 1s and 0s as a representation of the shape feature. Using Figure 2.11 as an example, the transfer result should be "00000000 11000000 11110000 01111000 00011110 00011110 00111000 00100000". According to the approach mentioned above, the smaller the grid size, the more accurate is the shape representation, although this

Please see print copy for Figure 2.11

Figure 2.11: Original shape region [102]

is at the cost of more storage and computation requirements. Furthermore, a scale and rotation normalisation is carried out to make the representation invariant to scale and rotation. According to the experimental results, the grid-based shape descriptor exceeds the performance of the region-based approach in both [72][103].

Texture Feature

Please see print copy for Figure 2.12

Figure 2.12: A reordering example of an 8×8 DCT block [48]

Texture feature is another alternative choice for the content-based image retrieval system [130]. In [48], in order to represent the DCT coefficients $C(\mu, \nu)$ within an $N \times N$

block in a multi-resolution form, the coefficients were firstly reordered into $(3log2N+1)$ multi-resolution sub-bands as shown in Figure 2.12. Then, by applying equations 2.13, 2.14 and 2.15, the mean μ_i, standard deviation σ_i and energy E_i, corresponding to each DCT coefficient sub-band were calculated. Finally, by using μ_i, σ_i and E_i as feature components, the texture feature vector could be constructed as in equation 2.16. By comparing the feature's components, the dissimilarity between the textures of an image could be judged.

$$\mu_i = \int\int |S_i(x,y)|dxdy \tag{2.13}$$

$$\sigma_i^2 = \int\int \lfloor |S_i(x,y)| - \mu_i \rfloor^2 dxdy \tag{2.14}$$

$$E_i = \int\int |S_i(x,y)|^2 dxdy \tag{2.15}$$

$$f = [\mu_0, \sigma_0, E_0, \mu_1, \sigma_1, E_1, \ldots, \mu_{log_2N}, \sigma_{log_2N}, E_{log_2N}] \tag{2.16}$$

Please see print copy for Figure 2.13

Figure 2.13: Computing the Edge Histogram Descriptor [118]

The Edge Histogram Descriptor (EHD) [118][1][74][145] adopted by MPEG-7, is a representation of local edge distribution in images. Specifically, by dividing the image space into 4×4 sub-images, as shown in Figure 2.13, the local-edge distribution for each sub-image can be represented by a histogram. The $S_i(x,y)$ is the value of pixel at point (x,y). f is the set of the mean (μ), the deviation (σ) and the energy (E) for

Please see print copy for Table 2.1

Table 2.1: Semantics of the histogram bins of the EHD [118]

each pixel of the image. Edges are broadly grouped into five categories: the vertical, horizontal, 45° diagonal, 135° diagonal and the nondirectional edge. Therefore, a total of $5 \times 16 = 80$ histogram bins are required. These bins are non-uniformly quantised using 3 bits/bin, resulting in a descriptor of 240 bits in size. The semantics of the 80-bin EHD are summarised in Table 2.1.

To compute the edge histograms, each of the 16 sub-images is further subdivided into image blocks. The size of these image blocks is associated with the image size and assumed to be a power of 2. The number of image blocks per sub-image is kept constant, independent of the original image dimensions, by scaling their size appropriately. Five edge detectors (four directional selective detectors and one isotropic operator as shown in Table 2.2) are then applied to each of the blocks, treating each block as a 2×2 pixel image. Those image blocks whose edge strengths exceeds a certain minimum threshold are used in computing the histogram. If the maximum of these edge strengths exceed a certain preset threshold, then the corresponding image block is considered to be an edge block. An edge block contributes to the edge histogram bins. Each of the image blocks labeled as edge blocks contribute to the appropriate bin of the histogram descriptor. These values are normalised to $[0, 1]$. A nonlinear quantisation of the bin values results in a 3 bits/bin representation.

The 80 bins of the local-edge histogram in Table 2.1 are the standardised normative

Please see print copy for Table 2.2

Table 2.2: Edge detectors and filters [118]

semantics for the EHD. Nevertheless, local-edge histograms alone may not be sufficient for an effective image matching. Some global-edge distributions, as well as the local ones, are therefore used. Specifically, edge distribution information for the whole image space, and some horizontal and vertical semi-global-edge distributions, as well as local ones, are required to improve the matching performance. The calculations of the global-edge histograms and semi-global-edge histograms are created easily and directly from the 80 local histogram bins. For the global-edge histogram, the five types of edge distributions for all the sub-images are accumulated. Similarly, for the semi-global-edge histograms, subsets of sub-images are grouped, as shown in Figure 2.14. In this case, there are 13 different segments. The corresponding edge histograms for each segment are then generated, using the local-edge histograms. Combining the local, the semi-global and the global histograms together, one can construct a total of 150 bins $[80bins(local) + 5bins(global) + 65bins(13 \times 5, semiglobal)]$ for similarity matching. For similarity matching, the ℓ_1 distance measure $D(A, B)$ can be adopted for two image histograms A and B as in the following equation:

$$D(A,B) = \sum_{i=0}^{79} |h_A(i) - h_B(i)| + 5 \times \sum_{i=0}^{4} |h_A^g(i) - h_B^g(i)| + \sum_{i=0}^{64} |h_A^S(i) - h_B^S(i)| \quad (2.17)$$

where $h_A(i)$ and $h_B(i)$ represent the normalised histogram bin values of image A and image B, respectively. $h_A^g(i)$ and $h_B^g(i)$ represent the normalised bin values for the global-edge histograms of image A and image B, respectively, which are obtained from the corresponding local histograms $H_A(i)$ and $H_B(i)$. Similarly, $h_A^S(i)$ and $h_B^S(i)$ represent the histogram bin values for the semi-global-edge histograms of image A and B, respectively. Since the number of bins in the global histogram is relatively smaller than that in the local and semi-global histograms, a weighting factor 5 is applied in Formula 2.17.

In [93], Dong et al. employed the EHD in their image retrieval system. In order

Please see print copy for Figure 2.14

Figure 2.14: Segments of sub-images for semi-global histograms [118]

to improve retrieval performance, they proposed using global and semi-local edge histograms to evaluate the similarity between images. Experiments on test images for the MPEG- 7 core experiment have shown that the proposed method yields a better retrieval performance, particularly in terms of semantic similarity.

In addition to all the features mentioned above, some other features could also be utilised in the content-based image retrieval system. However, irrespective of the features employed, experimental results have shown that searching images based on a single feature is inefficient and insufficient. This is because the human visual system perceives images using several aspects of the visual perception at the same time, including colour, shape and texture. Therefore, in order to improve retrieval performance and make retrieval results approach human expectations, features that represent different aspects of visual perception should be employed simultaneously to retrieve images.

Multi-Feature Approach

It is apparent that images can be described by several features simultaneously. Such descriptions are rich and should accord with human perception. However, it is not clear how humans combine features to achieve both description and observation. Feature extraction and similarity measures differ depending on the feature under consideration. It is not a straightforward task to combine them in a retrieval process. The problem is that of feature fusion, and several methods have been proposed in the literature. In the following paragraphs, several approaches to multi-feature usage are introduced in different subsections, including linear combination, non-linear combination, and neural networks.

Linear Combination Function

The most common approach to combining features is the Linear Combination Function (LCF). Let vector $\mathbf{d}$ stand for the local distances and $\mathbf{w}$ stand for combination weights that indicate the importance of each feature. The overall distance by combining all selected features can be calculated by LCF as follows:

$$D = \mathbf{d} \times \mathbf{w}^T, \tag{2.18}$$

$$\mathbf{d} = [d_1, \ldots, d_i, \ldots, d_I], \tag{2.19}$$

$$\mathbf{w} = [w_1, \ldots, w_i, \ldots, w_I], \tag{2.20}$$

$$\sum_{i=1}^{I} w_i = 1. \tag{2.21}$$

In [49], Iqbal and Aggarwal employed both colour and texture features in their retrieval system to improve retrieval performance. Both colour and texture features are assigned the same importance level for all queries. Although the experimental results performed better than those for the single feature, it is also found, however, that the equal weighting or fixed weighting approaches are not robust when considering the variance in query image and image database. Therefore, in order to make the retrieval algorithm more robust and flexible, Vadivel et al. did a series of experiments on their own image database [138] to try to find the relationship between features. A detailed study of the performance of different combinations of weights to colour (w_c) and texture (w_t) on a large image database ($28,168$ images) showed that the texture feature weight (w_t) in the range of $w_c \pm 0.1$ to $w_c \pm 0.2$ performed better than the other combinations, but no more than 10 percent in precision ratio. The maximum precision ratio is 81 percent. Although the weighting used in [138] was not fixed, the work did not present an analytical method of determining the weights. The result quoted in the work is database dependent.

In order to assign suitable weights to different features automatically, Shao et al. [117] proposed an automatic feature weight assignment approach based on a genetic algorithm. First, the problem of weight assignment is transferred into an optimisation problem. In order to obtain a maximum recall and precision performance [120], in each generation the best performance feature weighting is kept, and new weighting is regenerated for all the other features by a crossover method. Finally, the optimal weight could be generated for each feature. According to the paper [117], the proposed approach had the ability to assign suitable weighting to different features when combining the retrieval results. However, some drawbacks still exist in this method, which could be improved as follows:

1. Although suitable feature weights could be assigned in most cases, the genetic algorithm did not, however, guarantee to find the satisfied solution in certain rounds by the user. Therefore, in order to avoid infinite iteration, the algorithm should be terminated automatically after a certain time has passed.

2. The speed performance of this approach is very low. In order to generate an optimal solution, many iterations would be required, so some alternative improvement could be used to increase the speed, such as initialising the population according to the feature characteristics or users' requirements, although not randomly.

In summarising this linear combination approach, we can see some advantages, which are as follows:

1. It is easy to implement and control, since through changing only the weighting for different features, the retrieval result can be improved.

2. The physical meaning of weighting is very obvious: a higher weighting standing for greater importance of the current feature and vice versa.

Some disadvantages are nevertheless also evident, since users' visual perception cannot be easily represented by the linear weighting. In order to mimic users' visual perception in the retrieval process, therefore, the non-linear combination approach has been proposed.

Non-Linear Combination Function

Please see print copy for Figure 2.15

Figure 2.15: An example of fuzzy attributed relational graphs [58]

After taking a different approach based on psychological studies of human visual perception, Tamura et al. [130] claimed that the linear combination function is not suitable for feature combination when taking human perception into account. Therefore, Verma and Kulkarni [139] proposed a fuzzy-neural approach to interpret colour and texture features first, and then combine the results with neural-fuzzy, fuzzy AND and binary AND techniques. According to their analysis, the fuzzy-neural approach provided a significant improvement in performance. Androutsos et al. [60] even pointed out that retrieval performance may be reduced by some features, so they used fuzzy aggregation, such as logical AND and OR, to include or exclude some features. However, these fuzzy approaches mentioned above only apply to the feature combination level. In [58], Krishnapuram et al. developed a complicated feature extraction and combination system. First, the image is divided into regions and marked by linguistic labels. Then, through employing Fuzzy Attributed Relational Graphs (FARGs), they use nodes to represent image regions and edges between nodes to represent spatial relationships between regions. Finally, all images are converted to FARGs and a fuzzy graph-matching algorithm is employed to compare FARGs. In Figure 2.15, an example of FARGs is shown. Each node in the FARG represents a region in the image and edges between the corresponding nodes represent the relationships between regions. For each node, some attributes are extracted from the region and represented by λ_i^A. A is the set of these attributes. The spatial relationship between node i and node k is represented by $\rho^A(c_i, k)$, which may be one of r (r $\in$ left-of, right-of, above, below, surrounded-by). The advantage of this system is the representation of images in a fuzzy approach, which is similar to human perception [58]. The retrieval performance, around 90 percent for recall in average, is obtained. However, the proposed method is only tested on a database of over 1000 images.

In [51], the Combining Multiple Experts (CME) approach is employed in the combination of features. In this approach, the order of image, not the similarity value, are combined together according to the Borda Count method [113][151]. According to the studies, this system can combine features by considering their significance, although some issues also exist as follows:

- The multi-feature combination only considers the ranked order for each selected feature, therefore the physical meaning of each feature is discarded during the combination process.

- The ranked order cannot fully express the level of similarity to the query, especially for multi-features, since the physical meaning of each feature is different from all the others. Therefore, combining them together according to the ranked order sometimes does not make sense.

However, since the above method did not provide enough flexibility in modelling users' queries, Kushki et al. [60] proposed a framework to generate decisions for artistic repositories' image retrieval. The basic idea about their framework is that instead of

Please see print copy for Figure 2.16

Figure 2.16: UFSC overall structure [60]

performing a direct aggregation of different features' results, they construct another decision-level in their system which can employ the fuzzy logic principles to model conceptual queries. As shown in Figure 2.16, all the descriptor decisions based on features are transferred to the multiplexing and descriptor selector decisions unit. The multiplexing element (MUX) performs the actual selection of descriptor decisions $d_{i,j}$ and passes them to the appropriate aggregation mechanism a_k before the overall aggregation is performed (if necessary). Descriptor distances $D_{i,j}$ are passed through membership functions $\mu_{i,j}$ in order for descriptor decisions to be obtained. By employing a set of aggregation operators, various logical conceptual queries could be addressed. The family of aggregation operators used in this system are the family of quasi-linear compensatory operators, which range from a t-norm to a t-conorm. The general form of this class of operators could be expressed as:

$$QL = f^{-1}((1-\gamma)(T(x_1, \ldots, x_n)) + \gamma f(S(x_1, \ldots, x_n))), \tag{2.22}$$

where $x_1, \ldots, x_n$ are the elements being aggregated, T and S represent a t-norm and t-conorm respectively. With these aggregating operators, the conceptual queries could be expressed from the logical AND($\gamma = 0$) to the logical OR($\gamma = 1$). The merit of this system is its flexibility in the sense that it offers the ability to model a wide variety of conceptual queries rather than boolean expressions, Euclidean distance or the weighted average. According to the experimental results, compared with the Weighted Average method, the proposed method increases the precision ratio by 20 percent and recall ratio by 10 percent, respectively.

As well as these systems mentioned above, more systems concerning the non-linear combination of multi-features are reported in [91], [61], [67] and [29].

Neural Network Approach

Zeng [152] developed a neural network to assign weight to features. In contrast to the former training approach, this neural network randomly divided the training set S into two subsets, which are a training set S_1 (2/3 of S) and a holdout set S_2 (1/3 of S) in order to train and estimate the network respectively. The number of nodes, H, is automatically determined by monitoring the retrieval accuracy (precision) on S_2. Each time, the retrieval accuracy is calculated for a given number of nodes, H. The H will be increased by 1 in each iteration until the retrieval accuracy achieves a local maximum value. Finally, the feature weight for an input node i (feature i) could be decided as:

$$W_i = \sum_{j=1}^{H} \sum_{k=1}^{K} |V_{i,j} \times V_{j,k}|, \tag{2.23}$$

where $V_{i,j}$ and $V_{j,k}$ are the weights from input node i to node j and from node j to output node k, respectively. Equation 2.23 indicates that if a feature is important, it will have more influence on the output nodes by propagating forward through the nodes, and vice versa. An example of a possible neural network is shown in Figure 2.17. It can be seen that the input nodes are weighted by the middle nodes and sent to output.

Please see print copy for Figure 2.17

Figure 2.17: Diagram of neural network [152]

In [66], Laaksonen et al. proposed a neural self-organising technique for content-based image retrieval. By employing the TS-SOM [56][55], the neural units (images), which possess similar characteristics to the selected features are located closer to each other on the surface of each TS-SOM layer. By doing this, the authors propose that visually and also semantically-similar images have been mapped near each other on the map. Sometimes, however, possessing similar low-level characteristics does not mean sharing a similar semantic meaning. Employing human interactivity, users are required to grade each image as either relevant or irrelevant. The relevant images are assigned with a positive value, while negative values are given to irrelevant images. By employing a low-pass filter to separate the positive and negative values, relevant images are separated from irrelevant ones and gathered together. Then each image is given a qualification value, which depends on the local denseness of the positive value. Features that fail to coincide with users' conceptions always produce lower qualification values than those descriptors that match users' expectations. Finally, for each image, all the qualification values from the different descriptors are added together. Twenty images with the highest total qualification values are generated as a result of the query round. However, two negative aspects are also found in this system:

- Labelling each image as either relevant or irrelevant is a great deal of work for an application that has limited use.

- Since the qualification value assigned to each image is dependent on the local denseness of the positive value, each image's qualification value is therefore impacted by neighbouring images, which could reduce retrieval performance. For example, if more negative images are located on the surface, even though the positive images are very similar to the query, the qualification value for them is nevertheless still very low. These positive images will thus not be found in the final top twenty result.

More information about the neural network approach to image retrieval can be found in [8][81]. Review of the literature showed that notwithstanding the use of linear or non-linear combination approach the retrieval results is not significantly improved. For example performance in the retrieval results indicate a precision ratio of no more than 50 percent, when recall equals 100 percent. After analysing these methods, it is found that the main drawback was that few of these systems employed users' intentions to guide the retrieval process. The retrieval process was based only on low-level features extracted from images. However, even though images contain similar low-level features, there is no guarantee that they are the same as users' expected results. Some approaches are therefore proposed to modify the initial retrieval results according to users' expectations.

Relevance Feedback Approach

According to the research results of Rui and Huang [112], the major limitations of the former content-based image retrieval systems are:

1. They ignore the gap between high-level concepts and low-level features. Since the computer cannot map low-level features to high-level concepts as a human being can, this results in images with only low-level features that do not correspond with users' expectations.

2. They ignore the subjectivity of human perception of visual content. Human perception of images may be very different from person to person, according to personality, circumstances and visual content. Therefore, even with the same query image, the results expected by users may be very different. Without taking this characteristic into account, the performance of a CBIR system cannot be improved.

Please see print copy for Figure 2.18

Figure 2.18: The retrieval process [112]

The relevance feedback approach is the most popular method for overcoming the two issues mentioned above.

The idea of using relevance feedback to improve retrieval performance in CBIR systems was first formally proposed in [46] by Huang et al. in 1996. The basic idea and structure of the relevance feedback technique is illustrated in Figure 2.19 and Figure 2.20. In each iteration, users' high-level concepts are transferred to the system

Please see print copy for Figure 2.19

Figure 2.19: Conceptual functionality of the query-feedback algorithm [46]

in the form of feedback, and the expectations from users will be used to modify the low-level features and their retrieval results. The feedback system will not be terminated or output the modified merged result until the satisfactory results are generated or a certain number of iterations are finished.

In their classic paper, Huang and Yong [112] engaged users in the retrieval process. The structure of the retrieval process is shown in Figure 2.18. First, the object O is represented by features from f_1 to f_i. Also the query Q is represented by these features. Second, for each component of the feature r_{ij}, a suitable weight value w_{ijk} is assigned. By employing these weights, the distance between two features are calculated. Third, all features' distance are combined together by employing weights w_i. The feedback from users is ranked into five different levels according to the users' judgement (from highly relevant to highly non-relevant). Through comparing the initial results with the users' relevance feedback, in each iteration, the features' weighting will be modified according to the relevant level. Weighting for features in which users are interested will be increased, otherwise the weighting will be decreased by employing a linear combination function. Because users' perceptions have been considered in this system, the modified results are much more in accordance with users' expectations.

Please see print copy for Figure 2.20

Figure 2.20: Overall structure of query-feedback system [46]

However, Doulamis [22] pointed out that human perception does not fully associate with linear function, therefore the degree of relevance level could not perhaps fully indicate users' actual expectations. The modified features weighting could thus perhaps not fully represent users' expectation. It is difficult to obtain the degree of relevance level for each image, given the database size and time required.

Huang and Yong [106][109] therefore proposed another relevance approach to overcome the drawback of the degree of relevance. In [106][109], by assuming the rank of the weights matrix as one, they transformed the relevance problem into an optimisation problem. By applying Lagrange multipliers, the proposed formulation can generate the optimal weighting for features. The optimal solution tells us that if the distances among feedback images for one feature are smaller, this feature should receive a higher weight, and vice versa. However, there are still some limitations, which will affect the wide application of this proposed approach.

- Depending on the size of the image database and the speed of performance, giving feedback on all images in the results is laborious and difficult.

- The application of this approach is impacted by the similarity distance measures.

Ye et al. proposed another semantic relevance feedback approach in [73]. In their retrieval system *iFind*, they set up a keyword database associated with the image database. Each keyword is linked to one or more images in the database. The degree of relevance of the keywords to the semantic content of the image is represented as the weight on each link. Also, each image could be assigned multiple keywords with different degree of relevance. The structure of a semantic network is shown in Figure 2.21. For the retrieval process, in each iteration, each semantic relevance is modified by the keywords linking with the feedback images. Finally, both semantics and low-level features relevance feedback are combined to achieve the final results. Although the authors deemed that using both semantics and low-level features feedbacks together could achieve a significant improvement in the retrieval results, nevertheless they did not illustrate an efficient way to combine them together.

Please see perint copy for Figure 2.21

Figure 2.21: Semantic network [73]

In [134], Tian et al. claimed that in most relevance feedback retrieval systems, the dynamically updated low-level feature weights strategy is only based on a user's positive feedback, i.e., in [109] only the relevant images are considered. Therefore, they propose modifying this approach by utilising both positive and negative feedback. By employing the Support Vector Machines (SVM) method [133], the positive and negative feedback (images) are separated. The SVM learning results are used to update the weight of preference for relevant images. Priority is given to positive feedback with a large distance to the hyperplane determined by the support vector. The experimental results showed that this proposed approach display a reasonable improvement over the normal relevance feedback approach.

In [148], Wu and Zhang also proposed a category-based search to separate irrelevant features from relevant ones for the relevance feedback mechanism. By defining the factor of the dominant range and the discriminative factor, irrelevant features are successfully isolated and only relevant features will be assigned large weights. According to the experiment, on a Corel image set (with 31.438 images), at least 15 percent improvement on average precision and recall is achieved over the normal relevance feedback approach. More information about the relevance feedback strategy can be found in [156], [70] and [47].

Region-Based Approach

In addition to the relevance feedback approach, region-based image retrieval is another approach to eliminate the semantic gap between high-level concepts and low-level features. The motivation of region-based image retrieval is that a typical query image includes both relevant objects and irrelevant image areas (including background), and the traditional global features would extract both of them from the query image. Therefore, the traditional retrieval often fails in representing users' interests in the query image and the effectiveness of the CBIR system would be limited by these irrelevant areas. The region-based approach retrieved images based only on the Region-of-Interest (ROI) that is selected by users, but not on the whole query image. Since the users' subjectivity was embodied in the process of selecting interesting regions, so this approach was guaranteed to retrieve images according to users' expectations.

In [135], through combining user-defined Region-of-Interest and spatial layout, a higher retrieval efficiency is achieved; a 15 percent increase in precision for global region and 7 percent increase for the layout region approach. In the first step, the query image is divided into $n \times n$ non-overlapping image blocks. Based on the percentage of overlap between the user-defined ROI and the image block, the similarity distances for each image is calculated by linearly combining the individual image block similarity distance as follows:

$$D_j = \sum_n \sum_i W'_{n,i} S_j(n,i), j = 1, \ldots, N, \tag{2.24}$$

$$W'_{n,i} = \lambda W_{n,i}. \tag{2.25}$$

where D_j is the overall similarity distance of the j^{th} image in the database to the query image Q, $S_j(n,i)$ and $W_{n,j}$ are the similarity distances and their corresponding weight of the feature f_i in the n^{th} block of the j^{th} image in the database. $W'_{n,i}$ is the updated weight modified by the ratio of overlap between the users-defined ROI and each image block. Therefore, in the user-defined ROI approach, a greater weight will be assigned to the image block that contains the ROI and thus better retrieval results can be obtained than in the global approach. However, a drawback of this system is that it only supposes a single ROI query, when users may in fact be interested in more than one region at the same time. The application of this system is therefore limited.

Tian proposed a multiple Region-of-Interest image retrieval system to solve this issue in [88]. Users could select more than one interesting region (N) on the query image, and for each of them, the single ROI retrieval strategy is employed. Finally, all the single ROI retrieval results are combined using the following two proposed methods:

- Statistical Method: The basic principle of the statistical method is to average each image rank from the N query results to generate a final position for the image. For example, if P_i is the rank of an image I in the i^{th} of N query results, the final rank I for the image could be calculated as follows:

  $$I' = \frac{\sum_{i=0}^{N} P_i}{\sum_{j=0}^{N} \sum_{k=0}^{M} I_{j,k}} \tag{2.26}$$

  where M is the number of images returned from the query process and $I_{j,k}$ is the occurrence of image I in all the query results from the multiple ROI. According to the experimental result, performance was increased by 48% over the single ROI approach.

- Hierarchical Method: In the hierarchical method, for each ROI, the system will

Please see print copy for Figure2.22

Figure 2.22: The hierarchial searching of the query results [88]
This result is based on the $(i+T)^{th}$ user selected ROI.

return a matching image list which is a subset of all the images in the database. The old result of each ROI is used to generate the new ROI. The process of the hierarchical method could be structured as shown in Figure 2.22. By employing this kind of combination method, retrieval performance is increased by 30% over the single ROI method.

Please see print copy for Figure 2.23

Figure 2.23: Indexing and retrieval of 2D arbitrary shapes [141]

It is interesting to note that neither the single nor multiple ROI query described above supports arbitrarily defined queries. Users had to submit the entire image area as a query. In order to improve retrieval efficiency, users should be permitted to query arbitrarily-shaped images. In other words, one must be able to identify regions of interest that comprised the objects queried. Khanh [141] handled this issue by proposing a sampling-based approach called *SamMatch*. In *SamMatch*, samples of 16×16 pixel blocks were taken at various locations in each image. Through comparing only the sampled blocks falling within the sub-image area, this system can compare arbitrarily-shaped sub-images. The retrieval procedure is illustrated in Figure 2.23. It is supposed that an arbitrarily-shaped object of interest Q is to be saved and retrieved. At the building time, square windows W of various sizes are sliding over the database images. At each sliding location, a fixed-size signature is computed from the blocks enclosed in W. Signatures from windows with the same content in different sizes are considered as virtually identical and set to the same index page. Through detecting a core area on Q, the system can construct sub-images S which contain the same shape as Q, then direct similarity computations can be performed between Q and S.

2.3.4 Multiple Image Query Approach

Although both relevance feedback and region-based image retrieval approaches can improve retrieval performance according to users' expectations, they nevertheless still have drawbacks. For the relevance feedback approach, speed of performance is the major drawback. In some situations, satisfactory results can only be achieved after several iterations. For the region-based approach, neither the single nor multiple ROI method can detect importance between regions. One of the reasons for these drawbacks is that it uses only one relevant sample images in the query. If more than one sample images could be employed in the query, the number of iterations could be decreased for the relevance feedback approach because of a higher precision ratio in each iteration. By comparing common regions between different images, core regions could easily be detected in the region-based approach. Therefore, the multi-image query approach is proposed to eliminate these drawbacks.

In [9], through analysing the feature-to-semantics mapping, Bjoerge claimed that the query-by-one-example cannot realistically lead to scalable, satisfactory query performance. The query-by-one-example is therefore not adequate in order to achieve a higher retrieval performance. Tahaghoghi undertook experiments to illustrate that using multiple examples improved retrieval effectiveness by around 9 percent to 20 percent over single-example queries [129]. The multi-image query approach should thus be employed in some CBIR systems. Depending on the method of query image usage, multi-image query content-based image retrieval can be separated into the single group approach and the multi-group approach. Both of these are reviewed in this section.

Single Group Approach

In the single group approach, all query images are considered equally important in the process of retrieval.

Please see print copy for Figure 2.24

Figure 2.24: Single-image query vs. multi-image query [50]

Iqbal and Aggarwal [50] developed a CBIR system that supported multi-image queries. Users can select more than one sample image as their query, then the distance between the query images and the to-be-judged image can be computed as:

$$D(X_j, S) = \min_k d(X_j, S_k) \tag{2.27}$$

where D represents the distance of the image X_j to the set of images, S and d are the distance of X_j from an image S_k, which is contained in S. The effect of applying the multi-image query is shown in Figure 2.24. Although the retrieval performance is improved - according to the results, the precision increases from 35 percent to 95 percent - the relationship between the query images is nevertheless not analysed. The advantage of this system is that by using the multi-image query to replace the single one, the retrieval performance is increased. However, the disadvantages are also obvious. Without calculating and comparing the similarities and differences between the query images, neither significant features nor components can be detected. Therefore, although the multi-image query approach is employed in the system, the weights for features and components are not modified according to users' expectations. This system's results can therefore only be deemed as combining the results from different single query approaches, and essentially not a multi-image query approach. The users' expectations cannot thus be detected and the retrieval performance cannot be substantially improved.

In [131], Tang also proposed a multiple image query approach, although the concept of the retrieval process is different. As shown in Figure 2.25, different features extracted from different query images are combined so as to construct a new query concept, then further retrieval is processed based on this new query concept. In contrast to the former multi-image query approach, in this approach, features from different images and the linear combination approach are employed to combine the separate features. However, the author did not present a solution on how to modify the weights to balance the importance between features. The relationships between features cannot therefore be detected.

In conclusion, the advantages and disadvantages of the single group approach in multi-image query CBIR systems can be summarised as follows:

Please see print copy for Figure 2.25

Figure 2.25: Component-based image retrieval using multiple query images [131]

- Advantages: (1) retrieval performance is improved greatly over the single query approach; (2) it is easy to implement; and (3) the returned images are similar to most images in the query.

- Disadvantages: (1) it ignores the common and different characteristics between query images, therefore significant features between the query images cannot be detected; (2) retrieval performance is decreased by features that are dispersed between the query images; and (3) since both significant and individual characteristics are not detected, retrieval performance is impacted by irrelevant aspects of the query.

In order to overcome the drawbacks mentioned above, the multi-group approach is employed in multi-image query CBIR systems.

Please see print copy for Figure 2.26

Figure 2.26: Concept of the new feature space transform [86]

Multiple Group Approach

In [86], Nakazato proposed a Query-by-Groups (QBG) approach to multi-image query in database. In this system, users can supply more images as queries and specify them as relevant, irrelevant or neutral. The relevant groups are considered as positive samples, while irrelevant groups are negative samples and neutral groups do not contribute to the search. In the retrieval process, the positive samples are gathered in different groups and retrieval is based on these positive groups, as shown in Figure 2.26. This minimises the scatter of each positive class while maximising the scatter between positive and negative samples. For example, if the query contains both white and red flowers, because the white flowers and the red flowers are very different in their colour feature, so users can separate them into two positive groups as white and red. By employing this multi-group approach, irrelevant samples will not impact the retrieval performance and individual positive requirements are assured to be satisfactory at the same time. However, three disadvantages still exist in this system:

- First, this approach is proposed to improve retrieval results by revising them according to the users' partition. However, one query image sometimes contains both positive and negative features. The sample images cannot therefore be grouped as easily as the user might desire.

- Second, the process of grouping images cannot automatically be implemented by the system. Therefore, images must be manually divided into different groups, which is laborious.

- Although images are divided into positive and negative groups to indicate the level of similarity to users' expectations, significant features and components in the same group can nevertheless still not be decided. For example, if an image of a yellow plane is located in the positive group, the system, however, cannot detect which feature (colour or shape) or which colour (yellow or blue) is the most important and should be used to improve the retrieval results. Therefore, by mixing together features with different levels of importance, the retrieval performance of systems can not be improved based on the importance of features.

Please see print copy for Figure 2.27

Figure 2.27: Semantically related images are scattered in several visual clusters [52]

In [52], Jin et al. pointed out that in the multi-image query CBIR system, semantically related sample images may be very different in terms of their visual features. For example, as in Figure 2.27, images which contain similar semantic meanings are located in different clusters. Through analysing experimental performances, these authors claimed that retrieval by a query centre of multiple queries may achieve different effects when the queries are located in one or more than one cluster. If the queries are located in the same cluster, the query centre can help improve performance, but otherwise, the query centre will degrade performance, as demonstrated in Figure 2.28. In this case, it may achieve different effects when the queries are located in one or more than one cluster. They therefore proposed using multiple representations of the same feature to represent the image for further retrieval. Through combining the multi-query and multi-representations, the precision radio is increased 20 percent in average comparing with the retrieval results of sing representation.

In [11], instead of simply dividing the query image set into positive and negative groups, Brunelli and Mich divided them according to their visual characteristics. In order to achieve this aim, they undertook the following steps:

1. First, by employing the Linde-Buzo-Gray clustering algorithm [71] and silhouette coefficient [54], the query images are classified into several subsets according to their feature vectors' distances between each two query images.

2. Second, a single virtual average image is generated to replace the original images in each subset of the query. By applying this step, the amount of computation is reduced and the speed performance is improved.

However, by analysing the proposed system, four issues are found which could decrease retrieval performance.

1. In order to obtain an efficient classification result on the query image set, the number of images in the query is fixed between six and sixteen in this system. However, from the point of view of convenience and actual usage, it is hard to obtain so many images from the user for each query.

Please see print copy for Figure 2.28

Figure 2.28: Retrieval by query centre of multiple queries [52]

2. According to our studies, the retrieval system uses a classifier to make decision. Therefore, the retrieval performance will be impacted by errors arising from unsuccessful classification results.

3. According to [52], using the virtual average image to replace the original images may decrease retrieval performance if they are not located in the same cluster. Also, the virtual average image may be very different from the users' expectations.

4. Although the query images are divided into several sub-classes according to their low-level features, the system did not, however, propose a strategy to overcome the problem of how to assign suitable weights to each sub-class for the feature distance calculation and feature combination. Therefore, the system still employs the predetermined weights, which would not be suitable for some kinds of query

and would reduce their retrieval performance.

By comparing retrieval performances between the single group and multi-group approaches, retrieval performance is increased by around 20 percent on average through using the multi-group approach. The multi-group approach will therefore be advanced over the single one in the multi-image query CBIR. However, although the multi-group approach divides query images into different groups, such as positive and negative, the significant feature vector components still cannot be detected at present. The problem of how to assign suitable weights among groups and feature components still exists, and needs to be solved.

2.3.5 Some CBIR Systems

Based on these standard feature descriptors or other kinds of feature descriptors, many content-based image retrieval systems have been developed [108][92]. In the following subsections, some CBIR systems will be introduced to highlight their distinct characteristics.

MARS

MARS (Multimedia Analysis and Retrieval System) was developed at the University of Illinois at Urbana-Champaign [111][109][110][112]. MARS is an interdisciplinary research effort covering multiple research communities: such as Computer Vision, Database Management System (DBMS) and Information Retrieval (IR). By employing the relevance feedback technique, users are also involved in the retrieval process. In each iteration of the retrieval, the corresponding weights for features and features' components are updated dynamically according to users' feedback. The main focus of MARS is on how to organise various visual features into a meaningful retrieval architecture which can dynamically adapt to different applications and different users. On-line demonstrations of MARS are at http://jadzia.ifp.uiuc.edu:8080.

QBIC

The Query By Image Content (QBIC) is the first commercial content-based image retrieval system [24][26][87][96]. QBIC supports different kinds of queries, such as example images, user-constructed sketches and drawings, and selected colour and texture patterns, etc. By employing different colour space such as RGB colour space and HSV colour space [83], k element colour histogram [25] is used to represent the colour feature. For the texture feature, by combining the contrast in coarseness and directionality [24], an improved version of the Tamura texture representation is proposed [130]. Its shape feature consists of the shape area, circularity, eccentricity, major axis orientation and a set of algebraic moments invariant [25][115]. In QBIC, a high dimensional feature indexing system is considered. By employing KLT and R^*-tree, the high dimension is first reduced and the multi-dimensional indexing structure is constructed [68]. Furthermore, text-based keyword search strategy can also be combined with a content-based image retrieval engine in QBIC. An on-line QBIC demonstration is at http://www.qbic.almaden.ibm.com/.

VisualSeek

VisualSeek is a highly functional prototype system for searching by visual features in an image database [122]. It was developed at Columbia University [121][122]. The main research features of VisualSeek are that the user forms the queries by diagramming spatial arrangements of colour regions and the visual features are extracted from a compressed domain [142][14][16] [13]. In VisualSeek, the features of colour set and wavelet transform based texture are employed. Also, by utilising an efficient binary tree based indexing technique, the retrieval speed is also increased. The on-line demonstrations of VisualSeek are at http://www.ee.columbia.edu/ sfchang/demos.html

PicToSeek

PicToSeek is an object-based image retrieval system developed at the University of Amsterdam [30][31][32]. In this system, colour models are proposed independently of the object geometry, object pose and illumination. From these proposed colour models, colour invariant edges are derived from which shape invariant features are computed. By employing an efficient computation method, the colour and shape invariants are combined into a unified high-dimensional invariant feature set for discriminatory object retrieval. Experimental results illustrate that object retrieval based on colour invariants have provided a very high retrieval accuracy and the proposed image retrieval scheme is highly robust to partial occlusion, object clutter and a change in the object's pose. The on-line demonstrations of PicToSeek are at http://www.wins.uva.nl/research/isis/pictoseek/.

PhotoBook

PhotoBook is a set of interactive tools for searching and retrieval of images developed at MIT Media Lab [95]. In PhotoBook, three sub-books are employed to extract shape, texture and facial features from images, respectively. For each feature, the retrieval process is generated in the corresponding sub-book. PhotoBook also proposed to include human interaction in the image annotation and retrieval process [99][97][98]. By employing interaction between human and machine, the users' perception is considered in the retrieval process. Experimental results show that this approach is effective in interactive image annotation and retrieval [85][100].

Netra

Netra is a region-based image retrieval system developed in the UCSB Alexandria Digital Library (ADL) project [79]. By employing colour, texture, shape and spatial location information, images are segmented into several non-overlapping regions. During the retrieval process, each region in the query image is used to retrieve similar regions in the image database. The merits of the Netra system are its Gabor filter based texture analysis [4][76][82], neural nets based image thesaurus construction [75] [77][80] and edge flow based region segmentation [78]. The corresponding demonstrations of Netra are at http://vivaldi.ece.ucsb.edu/Netra/.

Virage

Virage is a content-based image search engine developed by Virage Inc [7][42][39]. Virage supports visual queries such as colour, colour layout, texture and structure. An advantage of Virage is that it supports arbitrary combinations of the above four features. Users can adjust the combination weights between these four features according to their own emphasis. The corresponding demonstrations of Virage are at http://www.virage.com/cgi-bin/query-e.

RetrievalWare

RetrievalWare is a content-based image retrieval system developed by Exclibar Technologies [23]. By employing the neural networks in image retrieval, features such as colour, shape and texture are combined together in order to achieve a better retrieval performance. Users are allowed to modify the combination weights according to their expectations. The corresponding demonstrations of RetrievalWare are at http://urw.excalib.com/cgi-bin/sdk/cst/cst2.bat.

PicSOM

PicSOM is a content-based image retrieval system developed at Helsinki University of Technology [65] [62][63] [66][57]. The key techniques employed in this system are pictorial examples, relevance feedback, vector quantisation [37] and self-organising map [62][63][64]. During the retrieval process, by employing TS-SOM [56][55], neural units (images) which possess similar characteristics are located together on the TS-SOM layer surface. Then both the positive and negative units are separated from each other by employing users' interactions and a low-pass filter. Finally, different features' results are combined together according to their corresponding qualification, which depends on the local denseness of positive responses on the SOM map. The MPEG-7 [15][118] visual descriptors are also employed in this system. By combining with the relevance feedback mechanism, this system's retrieval precision exceeds other reference systems [66].

FIRST

Fuzzy Image Retrieval System (FIRST) is an image retrieval system developed at Korea Telecom's Multimedia Technology Research Laboratory [59]. Unlike other image retrieval systems mentioned above, FIRST is based on a fuzzy logical algorithm. FIRST employs Fuzzy Attributed Relational Graphs (FARGs) to represent images, where each node in the graph represents an image region and each edge represents a relationship between two regions. Queries such as exemplar-based, graphical-sketch-based and linguistic with region labels, attributes and spatial relationships can be handled by FIRST. During the retrieval process, the given query is first converted into a FARG and a low-complexity fuzzy graph matching algorithm is used to compare the query graph with the FARGs in the database. The use of an indexing scheme based on a leader clustering algorithm also improves the system's performance.

ImageRover

ImageRover is a search by image content navigation tool for the World Wide Web developed at Boston University [116][132]. By employing techniques such as client-server architecture, optimized k-d tree [5] and relevance feedback, the system is subdivided into several sub-systems which include an image collection sub-system, image analysis sub-system and image query sub-system. For the colour feature, the image colour histograms are computed in the CIE LUV colour space. For texture, the texture direction distribution is calculated using steerable pyramids [28][36]. The corresponding demonstrations of ImageRover are at http://www.cs.bu.edu/groups/ivc/imagerover/.

PicHunter

PicHunter is a prototype content-based image retrieval system developed at NEC Central Laboratories [20] [21][19]. In this system, three pictorial features are employed to retrieve images, which are the HSV 64-element histogram, the HSV 256-element colour autocorrelogram [44] and the RGB 128-element colour-coherence vector [94]. The key research features in PicHunter are: 1) by employing the Bayes's rule [21] and an explicit model of the users' action, the goal images can be predicted; 2) an entropy-minimising display algorithm is proposed to maximise the information obtained from a user of each iteration of the search; 3) a hidden annotation strategy is proposed, so that the user does not need to learn and create queries as in an inaccurate/inconsistent annotation structure; and 4) two experimental paradigms used to quantitatively evaluate the performance of the system are proposed.

iFind

iFind is a web-based image retrieval system developed at Microsoft Research China [17][155][73]. It provides the functionalities of the keyword-based image search, query by image example, category-based image browsing, relevance feedback, and semi-automatic image annotation. The key technique in this system is combining the semantics network with relevance feedback strategy [112]. In each iteration of the retrieval, not only are the low-level features' weights modified, but also the annotation of each image in the database is updated according to users' feedback. According to the experimental results, the updated annotation can further help to improve the image retrieval results of the system for later use [73].

WillHunter

WillHunter is a content-based image retrieval prototype system developed at the National Laboratory of Pattern Recognition in China [146]. Visual features used in this system are a colour histogram in HSV colour space, colour moments, wavelet-based texture and a directionality histogram. The key research feature of this system is that by employing an SVM-based fast learning algorithm [33], a multi-level relevance measurement is proposed. The experimental results on real-world images verify that the proposed relevance-measuring instrument can better identify the users' needs and preferences.

In addition to the image retrieval systems described above, more image retrieval systems can be found in [27], [6], [127], [34], [147], [18], [89] and [126].

2.4 Comparative Analysis

In this section, through analysing the major approaches in image retrieval outlined in the sections above, a comparison between them will be given. The similarities and differences between them or some of them will be summarised in different subsections, respectively.

2.4.1 Similarities Between Image Retrieval Approaches

Content-Based Image Retrieval Approaches

In summary, by examining the major approaches to image retrieval reviewed above, it is easily recognized that apart from the text-based image retrieval approach, all the other methods belong to the content-based image retrieval category. The common characteristics among them are summarised as follows:

- Query by sample image/s is the basic principle of content-based image retrieval. Users or agents provide the retrieval system with only a query image/s which contains some visual content of interest to the user. In the retrieval process, all the images from the database are judged by the selected low-level features. Images that have similar visual characteristics as the query images are found as a result and are returned back to the user or agent.

- Images are retrieved according to the content of the query image/s, such as colours, shapes and textures. All of these visual content characteristics are represented as feature vectors or so-called descriptors, which are extracted from images automatically. Therefore, content-based image retrieval ensures that images are retrieved supposedly according to human visual perception.

- For a given query, the retrieval results will be impacted by the use of descriptors, similarity measures and feature combination approach adopted. Influences on the retrieval results arising from the user bias will therefore be limited.

Single Image Query Approach vs. Multi-Image Query Approach

Besides the similarities mentioned above, there are still some common characteristics or advantages shared by the single image query approach and multi-image query approach.

- The relevance feedback in both the single query approach and the multi-image query approach can improve the retrieval performance in accordance with users' expectations. By estimating users' expectation, feature weights or feature components weights can be appropriately modified.

- The region-based strategy can be applied to both of them. By employing human interaction or other approaches, the impact from irrelevant areas, such as background, can be mostly decreased.

2.4.2 Differences Between Image Retrieval Approaches

After summarising the similarities between retrieval approaches, in the next paragraph, the differences between them are highlighted and additional advantages of the multi-image query content-based image retrieval approach are revisited.

Text-Based Approach vs. Content-Based Approach

By comparing the differences between the text-based image retrieval approach and the content-based image retrieval approach, the advantages of the latter include:

- For the text-based approach, the whole image database needs to be annotated and each image must be manually labeled with suitable caption, descriptions or keywords. Whereas in the content-based approach, the description are automatically extracted by the feature extractor, thus making it more suitable for very large image database retrieval.

- The text-based approach needs to have keywords input as the query for further retrieval, while the content-based approach uses low-level features. Therefore, the results from content-based retrieval systems will be much more in accordance with human visual expectation to the extent that the feature captures relevant visual feature.

- Because of the impact of user bias, the retrieval results will be very different from person to person with the text-based approach, even though they are searching for the same goal. However, since the feature vector is generated from the image, the retrieval result for the given query will therefore be constant, no matter who performs the retrieval process.

As a result of all of the advantages listed above, the content-based image retrieval approach will perform much better than the text-based approach ***by considering the speed, the system accuracy and the time required for annotation.(Yu:04)***

Single Image Query Approach vs. Multi-Image Query Approach

Although relevance feedback in both the single image query approach and multi-image query approach can estimate users' expectations and improve retrieval performance, different principles and techniques will, however, result in differences in both retrieval and speed performances.

- For the relevance feedback approach, in each iteration, the modification is processed according to the relevance level assigned to the images resulting from the former iteration by the user. We can therefore say that the users' original query is somehow changed by the retrieval algorithm in each iteration to make it more suitable for the current image database and ensure that a better retrieval performance is achieved for the current image database. This technique is thus database dependent. On the other hand, for the multi-image query approach, the weights between features and feature components will only be decided by the query images, therefore this technique is database independent.*(According to the reviewer's comments, the un-suitable comparison is removed.)* ***Yu:05***

- In order to improve retrieval performance, the relevance feedback approach needs more iterations, which means that the speed performance will be impacted. Thus the retrieval and the speed performance cannot both be guaranteed in this approach. As shown in [112], the relevant ratio was only 50 percent in the original result, whereas after four iterations, the relevant ratio was increased to 90 percent. So both the retrieval and speed performance cannot be achieved at the same time. On the other hand, for the multi-image query approach proposed in this thesis, since the significant features and components can be calculated before the retrieval process, there is therefore no iteration in this approach and both the retrieval and speed performance can be guaranteed.

In the region-based approach, one or more Region-of-Interest could be used for further retrieval in order to eliminate the impact from irrelevant areas, such as background. However, in most region-based retrieval systems, the user still needs to select the region of interest manually. The multi-image query approach can also detect the users' interest in the query images through analysing the similarities and differences among them. This process can be executed automatically by the computer, therefore, the multi-image query approach has the following advantages over the region-based approach:

- Through analysing the feature vectors among query images, similar components in the feature vectors can be found and assigned with a higher weight. Therefore, with the multi-image query approach, the users' regions of interest among the query images can be automatically found.

- In the region-based approach, the user can select a region-of-interest for further retrieval to eliminate the impact from irrelevant areas. However, this approach cannot eliminate the impact from irrelevant features. The system still does not know which feature should be assigned with a higher weight in the process of retrieval. For example, suppose a user selects a region which contains a red flower as the interesting region of interest. Although the impact from grass and ground can be eliminated, this approach can still nevertheless not decide which features are more important, the red colour or the flower's shape or both of them. The retrieval performance still cannot therefore be improved to satisfy the users' expectations. However, by comparing the similarities and differences between features' vectors among the query images, this issue can be resolved. Using the example mentioned above again, if there is another image which contains a yellow flower and both of them are submitted as queries, with the multi-image query approach, the system will assign a higher weight to the shape features, while decreasing the weight for the colour feature, because of the similar shape and different colours in the query images. Therefore, the retrieval results will be closer to the users' expectations.

2.5 Chapter Summary and Conclusion

Image retrieval is a very important application area for digital image databases. Some of the major approaches towards image retrieval are reviewed in this chapter. Through analysing and summarising each approach and comparing both the similarities and differences between them, both the advantages and disadvantages in each of them are presented and evaluated. The studies first indicate the efficiency of content-based image retrieval. By extracting the feature vector from images, the low-level features of images are employed in the retrieval process. Compared with the text-based approach, content-based approach can therefore decrease the influence from user bias and be sure to obtain the retrieval results more effectively according to human visual perception. Furthermore, the studies survey the development of content-based image retrieval. From the pioneer multi-feature approaches to the latest multi-image query approaches, different strategies are proposed to improve retrieval performance in content-based image retrieval. According to the research results, the latest multi-image query approaches show a superior quality among these major approaches. However, as presented in the studies, some issues are also found in current multi-image query approaches, and these problems need to be solved in order to improve retrieval performance. The research direction in this thesis will therefore focus on approaches using the multi-image query and propose innovative methods to ameliorate retrieval performance.

Chapter 3

Automatic Weight Assignment Scheme for Image Retrieval Systems

3.1 Introduction

Content-based image retrieval (CBIR) continues to be an active research areas as the number images captured and published increases and the need to find a universal method of browsing and retrieving that takes advantage of the rich description implicit in images becomes imperative. Smeulders et al. [119] have given a tool-based review and Rui et al. [105] produced a system-based review. A Query-by-Example (QBE) image retrieval system is the most studied CBIR system, which makes use of low-level structural feature descriptions of example images to retrieve visually similar images from image databases according to a feature-based similarity measure. In such an approach, the performance of a CBIR system largely depends on the features selected and extracted from images and the metric used to measure the similarity. There is an underlying assumption that low-level structural features are able to represent the user query concept. In other words, the user is being asked to select an image whose description in terms of the selected features best matches those of the target image or images. It is well known that this assumption is challenged in many real systems given the current performance of content-based image retrieval systems and, the fact that the quest to fill the semantic gap remains [119].

Relevance feedback, originally developed for information retrieval, is one of the techniques that has been introduced into CBIR to improve retrieval performance. Retrieval systems with relevance feedback iteratively incorporate user intervention to label positive and negative images from previous retrieval outputs. Two main approaches used in relevance feedback systems include weight adjustment and probabilistic approaches [114]. The weight adjustment approach employs two major techniques: query point movement and feature weight adjustment. Zhang et al. [41] presented a brief overview of common relevance feedback techniques used in CBIR from the perspective of a machine learning problem. Briefly, the basic principle of the query point movement method is to update the query from user feedback, while the weight adjustment method assigns higher weights to features found in positive images and lower weights to features found in negative images. The overall distance metric is a summation of weighted feature distances. Rui et al. [109][112][150] proposed assigning weights inversely proportional to standard deviations of features, as well as shifting weights based on relevance feedback scores by the user. Ishikawa et al. [149] combined the query point movement and weight adjustment methods as a minimisation problem and showed that the weight adjustment scheme in [109][112][150] is optimal and is a special case. Rui et al. [106] later proposed a more general formulation of the problem and solutions. Their methods combined all the components of features into one vector and used the generalised Euclidean distance for this combined vector space. However, solution exists only when the number of positive feedback images is greater than the dimension of the combined feature vector. This is in addition to the high computational complexity of the algorithms. In many real retrieval situations, the number of positive relevance images is much smaller than the dimension of the combined feature vector. In addition, feature metrics vary greatly depending on the types of features. For example, the recommended metrics for MPEG-7 visual descriptors [2] differ from feature to feature and connote different physical meanings. The use of the generalised Euclidean distance for the combined feature vector is not appropriate in general.

More recently, in the direction of feed forward retrieval systems, multiple images are being proposed to define QBE retrieval tasks [9][129], in the hope of being able to better capture users' implicit description of the target image or images, at the front-end of the system without recourse to iterative loops required in relevance feedback systems. Currently, some CBIR systems have been developed based on multiple image (or multi-image) query retrieval paradigm. Tang and Acton [131] proposed a component-based the multi-image query method using multi-histogram intersection techniques. Zhu and Zhang [157] presented a variety of result combination strategies used in a geographic data retrieval system. Jin and James [52] proposed a multi-image query CBIR which used more representation signals for retrieval. Furthermore, Bjoerge and Chang [9] showed, through analysing feature-to-semantics mapping, that the query-by-one-example paradigm simply lacks the information to clearly identify the target in a query-concept and hence cannot achieve satisfactory retrieval performance.

In this chapter a method of automatically assigning appropriate weight to feature components and features in accordance to their significance is proposed. The intuition behind this approach is that the query-concept is mutually implied in the components of the feature vectors of multiple example images. The proposed method uses both the value of features and components of features to the query to modulate the corresponding weights and similarity metric.

3.2 Multi-Image Query Model

We motivate the development of the proposed method by considering a multi-image query with M images. Furthermore, let $f_i (i \in [1, F])$ be the i^{th} feature vector that can describe any of the M query images. The feature vector $\mathbf{f_i}$ has dimension K and will be represented, for the m^{th} query image, as row vector,

$$\mathbf{f_i} = [q^m_{i,1}, \ldots, q^m_{i,k}, \ldots, q^m_{i,K}], \tag{3.1}$$

Thus, we define the i^{th} feature matrix of the image query set, $\mathbf{Q_i}$, as the $M \times K$

matrix,

$$\mathbf{Q_i} = \begin{pmatrix} q_{i,1}^1 & ,\ldots, & q_{i,k}^1 & ,\ldots, & q_{i,K}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ q_{i,1}^m & ,\ldots, & q_{i,k}^m & ,\ldots, & q_{i,K}^m \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ q_{i,1}^M & ,\ldots, & q_{i,k}^M & ,\ldots, & q_{i,K}^M \end{pmatrix} \tag{3.2}$$

$$= (\mathbf{q_{i,1}}^T, \ldots, \mathbf{q_{i,k}}^T, \ldots, \mathbf{q_{i,K}}^T) \tag{3.3}$$

$$= (\mathbf{q_i^1}, \ldots, \mathbf{q_i^m}, \ldots, \mathbf{q_i^M})^T \tag{3.4}$$

The vectors, $\mathbf{q_{i,k}}^T$ and $\mathbf{q_i^m}$, are respectively, the k^{th} column and m^{th} row in matrix $\mathbf{Q_i}$ associated with feature f_i. T stands for the matrix transpose.

In a CBIR system, each feature vector is used to search in the database for other images that possess some visual characteristics similar to the query image as defined by the physical meaning of the chosen feature. In order to introduce weight assignment at the feature vector component level, we introduce further notation. Note that weight assignment at the feature vector component level has been referred to as *intra-level* modification model. Let the i^{th} feature vector associated with the n^{th} image in the database be denoted as, $\mathbf{x_i^n}$. It is convenient to denote the distance between a given query image set and an image from the database by d_i^n, for the i^{th} feature. The distance is computed by an appropriately defined distance function, Ψ_i [154], along with the weight vector $\mathbf{w_i}$ of dimension K. Thus,

$$d_i^n = \Psi_i(\mathbf{x_i^n}, \mathbf{Q_i}, \mathbf{w_i}) \tag{3.5}$$

$$\mathbf{w_i} = [w_{i,1}, \ldots, w_{i,k}, \ldots, w_{i,K}] \tag{3.6}$$

$$\sum_{k=1}^{K} w_{i,k} = 1 \tag{3.7}$$

The reasons why to employ the intra feature weight are because images are represented by the feature vector x_i^n with K dimension, in order to calculate the distance between two feature vectors or so called visual dissimilarity between two images, the difference between each two corresponding components from two feature vectors should be calculated and combined together. In order to calculate the difference, the distance function such as l_1 or l_2 can be employed. In order to combine these differences together, both linear and non-linear approaches can be employed. In this thesis, we will employ the linear approach to implement this combination. The reasons are:

1. the linear function is easier to implement than the non-linear one;

2. the linear function's results can be easily controlled by the weight vector w_i.

Each component in the weight vector w_i defines the relative importance of the corresponding component in the feature vector x_i^n. The reasons to employ the intra feature weight vector w_i are:

1. based on the extraction processing of the feature vector x_i^n from the image, the physical meaning of all components may be not similar. Therefore, the component which can represent the major visual feature of the image should be assigned a relatively higher importance level than other components;

2. based on the visual perception of the human-being. For the same image, different people may have different perception and emphases. Therefore, by employing the intra feature weight vector w_i, the personal preference can be imposed in the image retrieval process.

Besides the intra feature weight vector w_i, the intra feature similarity measurement is also very important in the content-based image retrieval process. In order to calculate the similarity between feature vectors, a suitable measurement should be employed. In general, the selection of the similarity measurement depends on the physical meaning of the feature vector. For example, in order to calculate the similarity between two texture feature vector with MPEG-7 Edge Texture Histogram, the L_1 distance is employed. While in order to calculate the similarity between two feature vector represented by the Colour Layout Descriptor of the MPEG-7, the L_2 distance is used. Therefore, the selection of the similarity measurement is decided by the characteristics of the feature vector and not by human perception. Once the measurement is selected in a content-based image retrieval system, it usually will not be changed. In this thesis, we simply employ the intra feature similarity measurement defined in the MPEG-7.

It is necessary to point out that the definition of "intra" in this thesis is different from the one in [112]. In this thesis, the "intra" refers to for the components of each feature vector. For example, the intra of the Color Layout Descriptor contains 13 components. While in [112], the "intra" refers to the feature vectors that represent the same characteristics of the image. For example, the Colour Layout Descriptor, the Color Structure Descriptor and the Scalable Colour Descriptor are called the "intra", because all of them present the characteristics of the colour in the image.

When multiple features are employed in a CBIR system, the results from the individual features are combined to generate the overall distance on the visual similarity between the image from the database and the query image set. Let D^n denote the composite distance between the n^{th} image from the database and the query image set when all the selected features $f_i (i \in [1, F])$ are considered. Furthermore, we denote as $\mathbf{u}$, the weight vector of dimension F, required for the multiple feature combination. If Φ is the distance function for the composite distance calculation , we have

$$D^n = \Phi(\mathbf{d^n}, \mathbf{u}), \tag{3.8}$$

$$\mathbf{d^n} = [d_1^n, \ldots, d_i^n, \ldots, d_F^n], \tag{3.9}$$

$$\mathbf{u} = [u_1, \ldots, u_i, \ldots, u_F], \tag{3.10}$$

$$\sum_{i=1}^{F} u_i = 1. \tag{3.11}$$

Note that weight assignment at the feature vector level has been referred to as *inter-level* modification model. In the next section, by analysing each query feature matrix $\mathbf{Q_i}$ and the inherent relationships of the components, the significant features captured by the image query set are encoded by dynamically modifying the weight vectors for both intra-level and inter-level models.

3.3 Intra-Level Weights Modification Model

3.3.1 Introduction

Content-Based Image Retrieval (CBIR) utilizes content description based on low-level features, such as colour, shape and texture. In the retrieval process, weights are assigned to each component of the low-level feature vector and the distance between images are computed. Therefore, the performance of the retrieval largely depends on the weights assigned to the components of the feature vector. There is the implicit assumption that the weights capture the relative significance of the components.

According to the distance measurement function 3.5, it can be seen that the distance between the query image set and the image in the database is decided by 1) the database image's feature vector (x_i^n) and the query images' feature matrix $\mathbf{Q_i}$. Since the feature vector and matrix reflects images' physical characteristics, it can not be changed according to users' expectation. 2) The distance measurement function $\Psi_i(\bullet)$. The measurement is associated with feature vector's physical meaning much more than users' interest, therefore once it is decided, it usually will not be changed at all. 3) The weights vectors $\vec{W}_i$. The weights assigned to the each component indicate the importance of each component and impact measurement result. Therefore, they can be modified to associate with users' expectation.

In this section, we propose a new method of consistently modifying the intra-level weights used in the computation of distance between images in accordance with the significance of the feature component. The intuition behind this approach is that the query-concept is implied in the components of the feature vector of multiple example images used in the query. For instance, if two images of cars, yellow in colour but with different backgrounds, are provided, the expectation is that the query is to search for yellow cars while playing down the significance of the backgrounds. The problem here is how to identify significant components in the feature vector and adjust the weights accordingly. The proposed method is to use the significance of the components to modulate their weights and similarity metric.

3.3.2 Related Work

There is ample evidence in the literature to justify the use of multi-image query approach to CBIR. In [9], Bjoerge analysied the feature-to-semantics mapping and concluded that the query-by-one-example cannot realistically lead to scalable, satisfactory query performance. Tahaghoghi [129] undertook experiments to illustrate that using multiple examples improved retrieval effectiveness by around $9\% - 20\%$ over single-example queries. This approach requires the analysis of the relationships among query images in order to determine the significant components and assign weights that hopefully capture the users' expectations.

In order to determine weights that will highlight significant components and attenuate insignificant components, among the query feature vectors, Huang and Rui [112] proposed to modify the weights using standard deviation of the components, computed over the query set. Their proposed approach can be summarised in three steps:

1. Standard deviation computation: For each column ($\mathbf{q_{i,k}}$) in matrix Q_i (Equation 3.3), the standard deviation is calculated as:

$$\sigma(\mathbf{q_{i,k}}) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(q_{i,k}^m - \mu(\mathbf{q_{i,k}})^2}, \tag{3.12}$$

where,

$$\mu(\mathbf{q_{i,k}}) = \frac{\sum_{m=1}^{M} q_{i,k}^m}{M}, \tag{3.13}$$

2. Components' weights modification: $\mathbf{w_i}$: The standard deviation obtained from step (1) is used to modify the original components' weights as:

$$\mathbf{w_i}' = [w'_{i,1}, \ldots, w'_{i,k}, \ldots, w'_{i,K}], \tag{3.14}$$

$$w'_{i,k} = \frac{w_{i,k}}{\sigma(\mathbf{q_{i,k}})}, \tag{3.15}$$

where $\mathbf{w_i}'$ is the modified components' weights;

3. Components' weights normalisation: Finally, the modified components' weights is normalised as:

$$\tilde{w}_{i,k} = \frac{w'_{i,k}}{\sum_{k=1}^{K} w'_{i,k}}, \tag{3.16}$$

$$\sum_{k=1}^{K} \tilde{w}'_{i,k} = 1. \tag{3.17}$$

By following these three steps, Huang and Rui proposed increasing the weights for the components whose values are closer to each other within the query set, and decreasing the weights for other components. Nevertheless, a study of the graphs of feature vectors, components' standard deviations and components' weights, revealed two issues in this approach that could decrease the retrieval performance or mislead the retrieval process for some queries.

1. By employing the standard deviation, "similar" components will be separated from "dissimilar" components and highlighted. However, the proposed approach does not consider the value of components. Therefore, both the components whose values are *"similarly large"* and *"similarly small"* will be indiscriminately highlighted. In fact, the physical meaning of the former is *"inclusive search"* and of the latter it is *"exclusive search"*. Sometimes, *"exclusive search"* will change a user's expectations or mislead the retrieval process. For example, if both of two query images contain a yellow flower with different sizes and do not contain any blue colour, then the standard deviation of the component representing "blue" (in a colour based feature vector) will be the smallest. By employing Huang's approach, the "blue" component's weight will be assigned an increased value, and this will imply that the system will be "looking for images without blue". However, we in fact would like to increase the weight for the component representing "yellow", so as to be "looking for images with yellow". Therefore, modifying components' weights based only on the standard deviation is not sufficient;

2. For some components, if the standard deviation is zero, then the modified weight will become infinite, and will swamp other components (in other word, "ignore others components"). A normalisation is therefore required to exclude the situation of infinity.

3.3.3 Proposed Approach

In order to address these two issues, a new approach is therefore proposed to modify components' weights. The two problems are overcome as follows:

1. By considering the value of components, three different cases will be covered in the weights modification:

    (a) For components whose values are close to each other within the query image set and where the numerical values are also large enough (bigger than a threshold), the weights will be greatly increased;

    (b) For components whose values are close to each other within the query image set but where the numerical values are smaller than the threshold, the weights will be increased slightly;

    (c) For components whose values are not close to each other, the weights will be decreased.

    The proposed approach will therefore ensure that the "similarly large" components are greatly emphasised and the retrieval process is mostly guided by "inclusive search".

2. By employing the Gaussian normalisation, the standard deviation of components will be normalised before the weights are normalised, which will:

    (a) Eliminate the situation of infinite weight; and

    (b) Take into account the relationship between components.

We therefore propose modifying components' weights based on both standard deviation and mean, and the process can be summarised into four steps as follows:

1. Mean (μ) and standard deviation (σ)computation: For each column in the matrix Q_i, the standard deviation is calculated as follows:

$$\sigma(\mathbf{q_{i,k}}) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(q_{i,k}^{m} - \mu(\mathbf{q_{i,k}}))^2}, \tag{3.18}$$

where

$$\mu(\mathbf{q_{i,k}}) = \frac{\sum_{m=1}^{M} q_{i,k}^{m}}{M}, \tag{3.19}$$

and

$$\mathbf{q_i^{\mu}} = [\mu(\mathbf{q_{i,1}}), \ldots, \mu(\mathbf{q_{i,k}}), \ldots, \mu(\mathbf{q_{i,K}})], \tag{3.20}$$

$$\mathbf{q_i^{\sigma}} = [\sigma(\mathbf{q_{i,1}}), \ldots, \sigma(\mathbf{q_{i,k}}), \ldots, \sigma(\mathbf{q_{i,K}})]. \tag{3.21}$$

2. Gaussian normalisation of the standard deviation: The standard deviation for all the components will be normalised as follows:

$$\sigma(\mathbf{q_{i,k}})' = \left(\frac{\sigma(\mathbf{q_{i,k}}) - \mu(\mathbf{q_i^{\sigma}})}{3 \times \sigma(\mathbf{q_i^{\sigma}})} + 1\right)/2, \tag{3.22}$$

where

$$\mu(\mathbf{q_i^{\sigma}}) = \frac{\sum_{k=1}^{K} \sigma(\mathbf{q_{i,k}})}{K}, \tag{3.23}$$

and

$$\sigma(\mathbf{q_i^{\sigma}}) = \sqrt{\frac{1}{K}\sum_{k=1}^{K}(\sigma(\mathbf{q_{i,k}}) - \mu(\mathbf{q_i^{\sigma}}))^2}. \tag{3.24}$$

Note that $\sigma(\mathbf{q_{i,k}})'$ is the normalised components' standard deviation;

3. Computation of modified weight of components, $\mathbf{w_i}$: Both the standard deviation and mean obtained from step 1 are used to modify the original components' weights as follows:

$$\mathbf{w_i}' = [w'_{i,1}, \ldots, w'_{i,k}, \ldots, w'_{i,K}], \tag{3.25}$$

$$w'_{i,k} = \frac{w_{i,k}}{\sigma(\mathbf{q_{i,k}})'} \times \Delta(\mu(\mathbf{q_{i,k}})), \tag{3.26}$$

$$\Delta(\mu(\mathbf{q_{i,k}})) = \begin{cases} \alpha, & \mu(\mathbf{q_{i,k}}) \geq \gamma \\ \beta, & \mu(\mathbf{q_{i,k}}) < \gamma \end{cases} \tag{3.27}$$

$$\alpha = \frac{\max(\mathbf{q_i^{\mu}})}{\min(\mathbf{q_i^{\mu}})}, \tag{3.28}$$

$$\beta = 1, \tag{3.29}$$

$$\gamma = \mu(\mathbf{q_i^{\mu}}). \tag{3.30}$$

The parameters α, β and γ are constant values. α refers to when the average of a certain component is bigger than the threshold γ, its corresponding weight will be enlarged in a certain level. The reason why we use the ratio of maximum q_i^u and the minimum q_i^u as α in our system is because we want to enlarge each component's weight by considering all the others. β is set to 1 and means when the average of current component is smaller than the threshold, the corresponding weight will not be changed. γ is the threshold and is set to the average of q_i^u.

4. Normalising components' weights $\mathbf{w_i}'$: Finally, the modified components' weights will be normalised as follows:

$$\tilde{w}'_{i,k} = \frac{w'_{i,k}}{\sum_{k=1}^{K} w'_{i,k}}, \tag{3.31}$$

$$\sum_{k=1}^{K} \tilde{w}'_{i,k} = 1. \tag{3.32}$$

The physical meaning of the proposed solution is that, by calculating the standard deviation and mean, we can estimate the significance of the components of a feature vector for a given query. The smaller the standard deviation and the larger the mean of a given component, the more significant it will be deemed. Therefore, if the significance level for one component is very high, the weight will be increased, otherwise, it will be decreased. In Section 3.5, experiments will be presented to test and validate the proposed approach.

3.4 Inter-Level Weight Modification Model

3.4.1 Introduction

In the previous section (i.e. Section 3.3) the basis of Query-by-Example (QBE) using intra-level weight modification model was presented. It was clear that a system based on the model does not account for variation in the significance of the features used in describing images in a query set. The performance of the system largely depends on the feature selected and the distance metric used to measure the similarity. The choice of features used for visual similarity retrieval should capture the underlying user expectation in the query. In order to improve the performance of retrieval systems, multiple features are used and individual results are combined to achieve a final result. In the sequel, we present the inter-level weight modification model that allows for the assignment of weights in accordance with estimated significance of each feature. It can be conjectured that a careful choice of features can mimic human visual perception through combining results from multiple features. Thus these two-level models could lead to a higher accuracy in CBIR performance [106].

As shown in Equation 3.8, the final retrieval result is decided by both the individual feature's results and the combination weights. The former will indicate how similar the database images and the query images are, by considering only one feature. The latter will decide how to combine these individual features together. The more important a feature is, the higher the assigned combination weight will be.

The intuition behind the approach presented here is that the query-concept is implied in the mutual features of multiple example images. For instance, if two images of cars with different colours are provided, the query is meant to retrieve cars regardless of colours. Furthermore, if two images of cars with a similar colour in distinct backgrounds are provided, the query can be interpreted as "retrieve cars with a similar colour in any background". The problem here is how to identify significant mutual features in example images and use them to query the database. Our method is to use feature significance to modulate the similarity metric and feature combination.

3.4.2 Related Work

As introduced in the previous section, the weights used in the combination of the features have significant impact on the final retrieval results, and much work has been carried out on weights estimation.

In [50] Iqbal and Aggarwal undertook a series of experiments on their database (10,221 images) and claimed that by combining feature structure (S), colour (C) and texture (T), the retrieval performance was improved by around 18% over the use of single feature. They also pointed out that by applying different weights to the three features, retrieval performance could be greatly enhanced. For example, when using a flower and vegetation as the query, by applying equal weight (S=0.33, C=0.33, T=0.33), the precision ratio is only 15%. However, by increasing the weight of the texture (S=0.05, C=0.05, T=0.9), the precision ratio is increased to 80%. This indicates therefore, that the weight of features can often be used advantageously on the final retrieval results. Although these authors also employed the multi-image query approach to improve retrieval performance, they did not modify the weights according to the physical characteristics of query images and only used fixed weights for all queries. It is easy to see that an improved performance could be obtained by using different weights for different queries, since a single fixed weight could not possibly capture the significance of each feature for different queries. For instance, if the query contains three images and each of them is a yellow flower but different in size, then the weight for colour should be increased, because the query shares a similar colour and colour is the significant feature. If each of the query images is a flower similar in size but different in colour, then texture should be considered as the significant feature and assigned a higher weight. An adaptive modification of this nature is more likely to adjust the weight according to the users' interest.

In order to overcome the problem of fixed weights and make the retrieval algorithm more robust and flexible, Vadivel et al. conducted a series of experiments on their own image database [138] to try to find the relationships between features. A detailed study of the performance of different combinations of weights assigned to colour (w_c) and texture (w_t) was conducted on a large image database ($28,168$ images). Their experimental results showed that when the weight of texture feature vector, (w_t), was in the range of $w_c \pm 0.1$ to $w_c \pm 0.2$, the retrieval performance was much better than in other combinations. Although the weighting factors used in [138] were not fixed, their work did not, however, present an analytical method of determining the weights. We also note that the results quoted in the work are database dependent.

In order to automatically assign suitable weights to different features, Shao et al. [117] proposed an automatic feature weight assignment approach based on a genetic algorithm. First, the problem of weight assignment is formulated as an optimisation problem. In order to obtain a maximum recall and precision performance, the best performance feature weights in each generation was kept and new weight was regenerated for all the other features by using a crossover method. Finally, the optimal weight was generated for each feature. According to the result presented in the work, this approach had the ability to assign suitable weight to different features when combining the retrieval results. However, there were still some obvious drawbacks to this method, which could be improved as follows:

1. Although suitable weight could be assigned in most cases to the features, the genetic algorithm require several iterations to find the optimal solution. Therefore, in order to economize on the number of iterations, the algorithm should in practice be automatically terminated after a certain number of passes. This does not guarantee an optimal solution.

2. The speed performance could also be increased by initialising the population according to the characteristics of the features or the users' requirements.

The relevance feedback technique is another popular method that has been applied to solve the problem of weights determination. The concept of using this kind of approach to improve retrieval performance in CBIR systems was first formally proposed in [46] by Huang et al. in 1996. Retrieval systems with relevance feedback (RF) incorporate user interaction by providing positive and negative examples from previous retrieval outputs in an iterative process and adjust the weight of features based on users' feedback. Once satisfactory results are generated, or after a certain number of iterations, the feedback system will be terminated and will output the modified and merged results.

In their classic paper, Rui et al. [112] employed users in the retrieval process. The feedback from users was ranked into five different levels according to the judgement of users (from highly relevant to highly non-relevant). By comparing the initial results with the users' relevance feedback in each iteration, the features' weights were modified according to the relevant level. The weighting of the features in which users are interested was increased, otherwise the weighting was decreased. Because the users' perceptions were considered in this system, the modified results were much more in accordance with users' expectations.

Rui and Huang [106] therefore proposed another relevance approach to overcome the drawback of degree of relevance. In [106][107][109], by assuming the constraints of weights matrix to one, they formulated the relevance problem as an optimisation problem and utilized the Lagrange multipliers method to generate optimal weighting for the features. The optimal solution indicated that if the total distance of feature f_i were smaller, this feature should receive a high weight, and vice versa. However, there are still some limitations that will impact the wider application of the proposed approach, which are:

- Depending on the size of the image database and the speed performance, giving feedback on all the images in the results is laborious and difficult;
- The combination scheme assumes that the features can be linearly combined. and the metric for the similarity measure is quadratic. In this case features that use $l_1 - norm$ are not suitable.

Although much work has been undertaken on assigning the weights of features, as outlined in all the approaches mentioned above, it is nevertheless observed that work on estimating users' expectations and assigning features' weights by a query image's physical characteristics has not yet been considered. Work to date has only focused on how to solve the optimal performance equation or update the weights with users' interactions in each iteration. In the next section, therefore, we propose an approach to articulate users' expectations and assign the weights of features in the multi-image query approach using significant features.

3.4.3 Proposed Approach

In order to modify the inter-level weights, we first develop the feature distance set, which consists of the distance between each two sample images in the query. The distance set D_i, for feature f_i, could be written as,

$$D_i = \{d_i^{p,t}\}, p, t \in [1, M], p \neq t, \tag{3.33}$$

$$d_i^{p,t} = \Psi_i(\mathbf{q_i^p}, \mathbf{q_i^t}, \mathbf{w_i}'), \tag{3.34}$$

where $\mathbf{q_i^p}$ and $\mathbf{q_i^t}$ are the p^{th} and t^{th} feature vectors in the Q_i matrix. The vector, $\mathbf{w_i}'$, is the modified *intra-level* weight proposed in the previous section. In order to eliminate the error caused by different amplitudes in different features, we normalise $d_i^{p,t}$ by the Gaussian normalisation as follows:

$$d_i^{p,t'} = (\frac{d_i^{p,t} - \mu(G_i)}{3 \times \sigma(G_i)} + 1)/2, \tag{3.35}$$

$$G_i = \{d_i^n\}, n \in [1, N], \tag{3.36}$$

After the normalisation, we obtain a normalised distance set D_i', and $D_i' = \{d_i^{p,t'}\}$. The combination weight for feature f_i can be calculated as follows:

$$u_i' = \frac{u_i}{\sqrt{\mu(D_i') \times \sigma(D_i')}}, \tag{3.37}$$

where the u_i' is the modified *inter-level* weight for feature f_i. In our development, attributing the same importance to all features implies that the u_i is initialised with a value of $\frac{1}{F}$.

The normalised and weighted distance between the n^{th} database image and the query images set, is given by,

$$D^{n'} = \Phi(\mathbf{d^{n'}}, \mathbf{u}'), \tag{3.38}$$

$$\mathbf{d^{n'}} = [d_1^{n'}, \ldots, d_i^{n'}, \ldots, d_I^{n'}], \tag{3.39}$$

$$d_i^{n'} = (\frac{d_i^n - \mu(G_i)}{3 \times \sigma(G_i)} + 1)/2, \tag{3.40}$$

$$\mathbf{u}' = [u_1', \ldots, u_i', \ldots, u_I'], \tag{3.41}$$

$$\sum_{i=1}^{I} u_i' = 1. \tag{3.42}$$

The physical meaning of the proposed solution is that by calculating the standard deviation and mean of each feature vector's components, we can find which features are significant for the query. The lower the standard deviation and mean of the distance for a given feature f_i, the more significant it will be deemed. Therefore, if the significance level of one feature is very high, the weight will be increased, otherwise the weight is decreased. In the next section, experiments will be carried out to test, validate and evaluate the proposed approach.

3.5 Experiments

First, the setup of the experiments and the evaluation method are introduced. Second, the single image query approach is compared with the multi-image query approach. Third, the proposed intra-level modification model is employed to make a comparison with equal weights and the method proposed by Huang et al. Fourth, the proposed inter-level modification model is applied and compared with equal weights. Finally, both the proposed intra- and inter-level modification models are employed and compared with equal weights.

3.5.1 Setup

In this subsection, the configuration of the experiments is introduced, including image database, query, ground truth and evaluation method. Three different methods are compared. Method A is the equal weights. Method B is Rui and Huang proposed [112] weights. Method C is the proposed weight assignment model. It is necessary to mention that we are not going to compare the relevance feedback approach with the multi-image query approach, since they are proposed to solve different issues. For example, in a relevance feedback system, the multi-image query approach can also be employed in order to improve the system performance. Method B is one of the approaches in the relevance feedback system which is used to solve the problem of the weights updating. Meanwhile, in the multi-image query approach, we also need some ways to update the weights dynamically. Therefore Method B [112] has been adapted for use in a multi-image query system. Method C provides improvement on Method B. The comparison provided accounts for the weight update methods only.

Image Database

The image database consists of 5210 images from a "real-world" collection. The content of the images comprises vehicles, buildings, plants, animals, landscapes, artworks and other images. The size of the images range from 170×128 pixels to 3721×3086 pixels. All the images are stored in JPEG format. Before the experiments are carried out, the feature vectors of each image for all the selected descriptors are extracted and stored in an XML file in advance. The actual retrieval process works on the XML file database.

Query

In order to test the retrieval performance, 87 different query sets are selected from the image database. The principles of designing of the queries are:

1. making sure the queries are comprehensive enough for our database. Therefore, for each category of image in the database, some queries are designed for it;

2. in order to ensure the reality of the experiments, some of the query images ar directly selected out from the database while others are not;

3. In order to ensure the accuracy of the experiments, the number of image in the query is ranged from only one image to five images;

4. In order to test the proposed schemes, images in the same query contain some common characteristics, such as similar colour or nearly colour layout or closed texture.

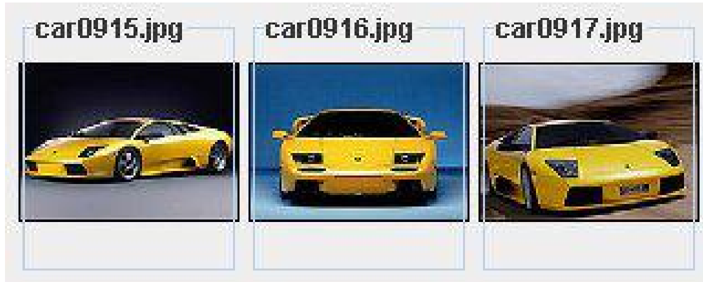


Figure 3.1: Sample query

Figure 3.2: Sample ground truth

For each query, the ground truth is also selected by users (21 people). An example of the query set and corresponding ground truth are displayed in Figures 3.1 and 3.2 respectively. Therefore, by comparing the retrieval results with the ground truth, the retrieval performance can be evaluated in the form of precision and recall ratio. In the experiments, all the ground truth images are highlighted by a red background if they appear in the retrieval results.

Descriptors

In the experiments, five different descriptors, from MPEG-7 Visual Descriptors, are employed as features to test the performance of the proposed methods. They include the Colour Layout Descriptor (CLD), the Colour Structure Descriptor (CSD), the Scalable Colour Descriptor (SCD), the Edge Histogram Descriptor (EHD) and the Homogeneous Texture Descriptor (HTD). These descriptors have been presented in Chapter 2.

Performance Evaluation

In order to evaluate the retrieval performance of CBIR systems, the effectiveness of retrieving target images in rank order for given queries are measured. A common practice in information retrieval for evaluating retrieval effectiveness is as follows [120]: a benchmark query is submitted to the system, the system retrieves images in rank order, then for each cutoff value k, the following decision theoretic values are computed, where $V_n \in [0, 1]$ is the relevance of the document with rank n, and where n, $k \in [1, \ldots, N]$ range over N images:

- $A_k = \sum_{n=1}^{k} V_n$, is the total relevance of the top k results;
- $B_k = \sum_{n=1}^{k} (1 - V_n)$, is the total irrelevance of the top k results; and
- $C_k = \sum_{n=k+1}^{N} V_n$, is the total relevance of the items not in the result set.

The following quantitative retrieval effectiveness measures, Precision, Equation 3.43, and Recall, Equation 3.44, are easily computed:

$$P_k = \frac{A_k}{A_k + B_k}, \tag{3.43}$$

$$R_k = \frac{A_k}{A_k + C_k}. \tag{3.44}$$

Precision is the ratio of relevant retrieved images to the total retrieved images and is an indication of the efficiency of the retrieval. Recall is the proportion of desired results retrieved within the first k results.

3.5.2 Single Image Query compared with Multi-Image Query

In this subsection, the retrieval performance between single image query and multi-image query is compared. 87 different images query set is employed to test the performance. For the multi-image query, the number of query images range between 2 and 5. Each selected descriptor is tested separately and a combination of two descriptors is also tested. Both the intra- and inter-level weights are not modified, therefore equal

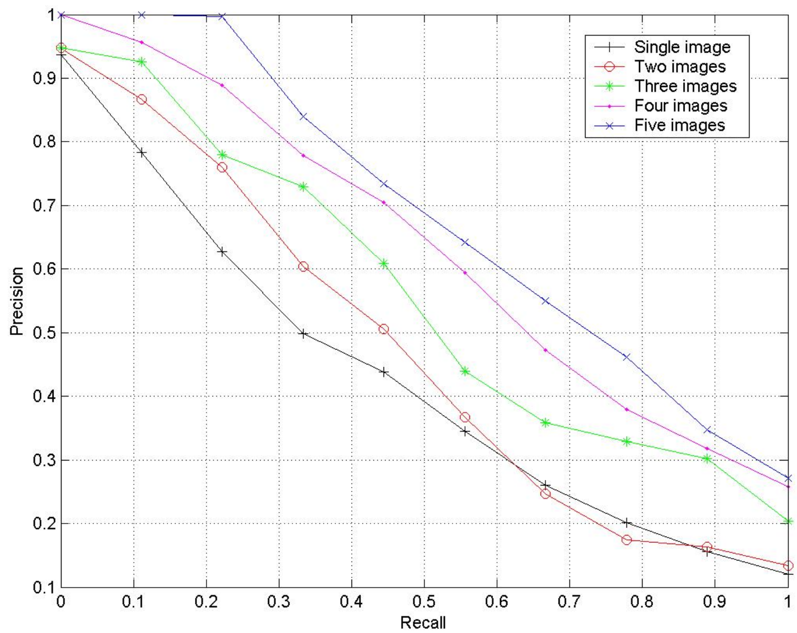


Figure 3.3: Comparison of CSD results

weights are employed in this experiment. The average retrieval performances of 87 queries for CLD, CSD, SCD, HTD and EHD are displayed from Figure 3.3 to Figure 3.7. It can be seen that the multi-image query approach performs much better than the single image query approach (20 percent improvement mostly), and as the number of query images increases, the retrieval performance is improved by 5 percent in average. In Figure 3.8, all the single descriptors' retrieval performances are compared. For the colour descriptors, the CSD performs the best and the EHD performs better among the texture descriptors. We therefore simply select these two descriptors as the multi-features for combination. In Figure 3.9, it can also be seen that for multi-feature retrieval, the multi-image query is also superior to the single image query approach.

3.5.3 Intra-level Weights Modification

In order to test and evaluate the proposed intra-level weights modification model, we compare the retrieval results with those obtained using method A and B. First, we display the retrieval performance for all the selected single descriptors. Second, the

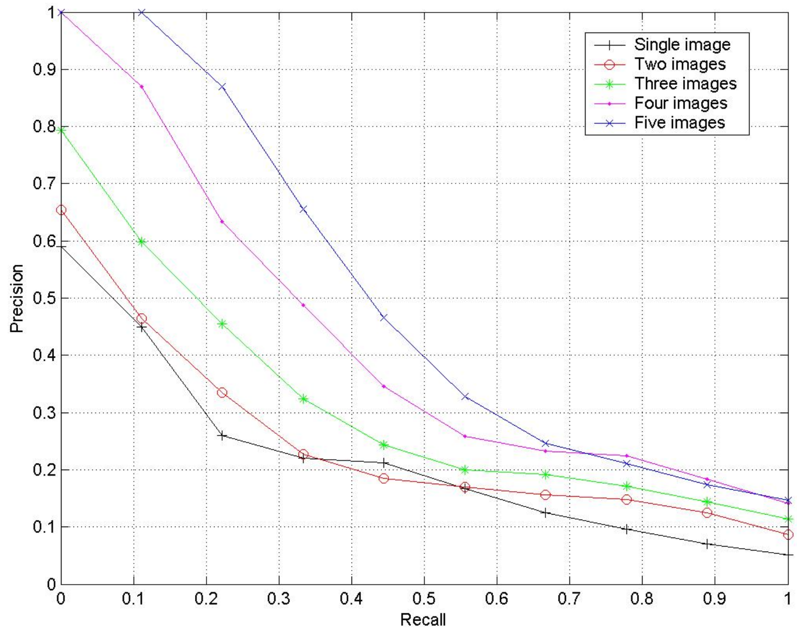


Figure 3.4: Comparison of CLD results

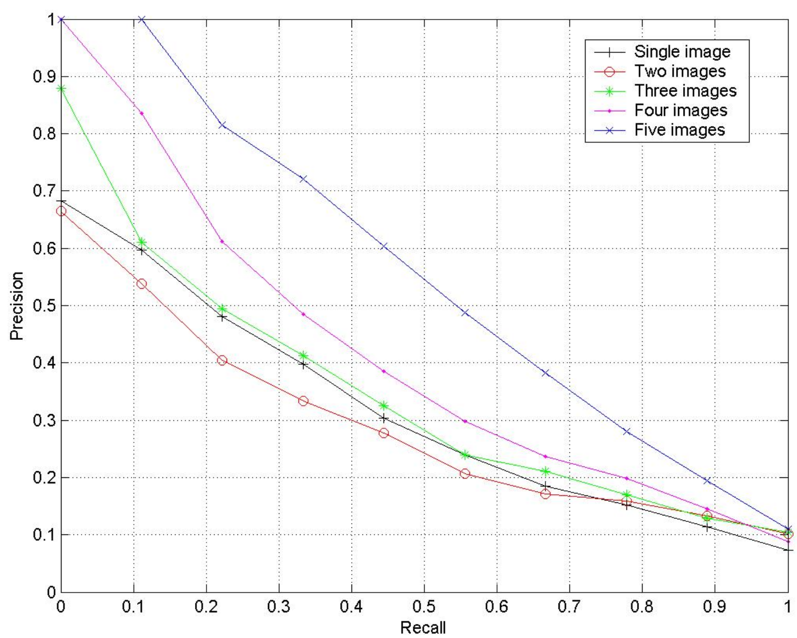


Figure 3.5: Comparison of SCD results

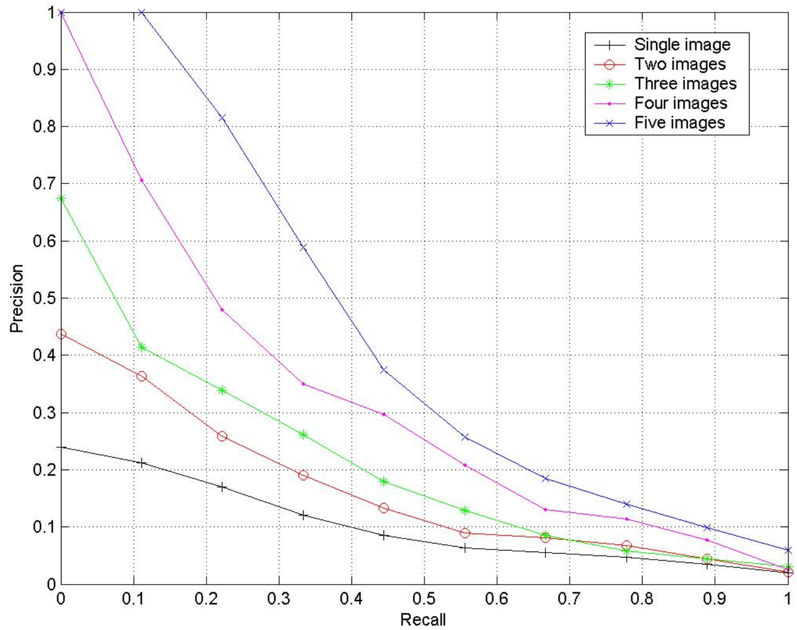


Figure 3.6: Comparison of HTD results

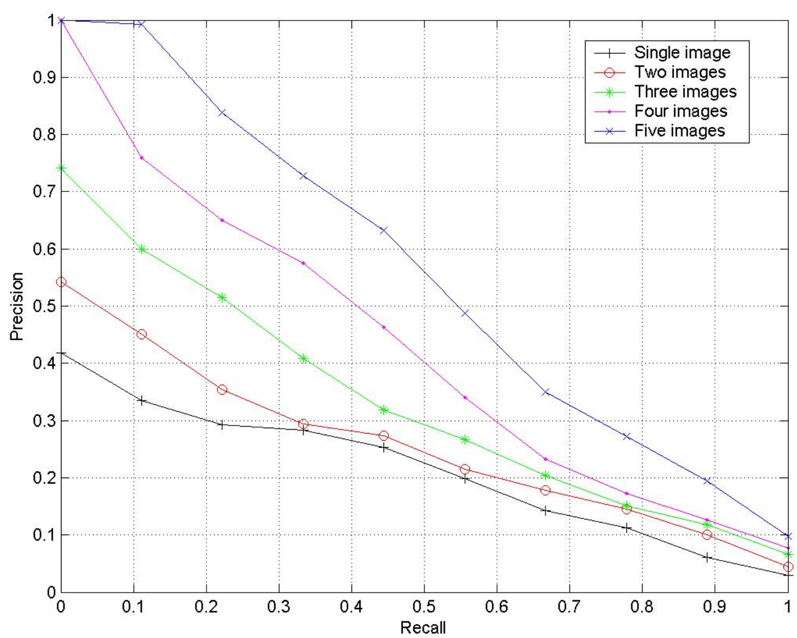


Figure 3.7: Comparison of EHD results

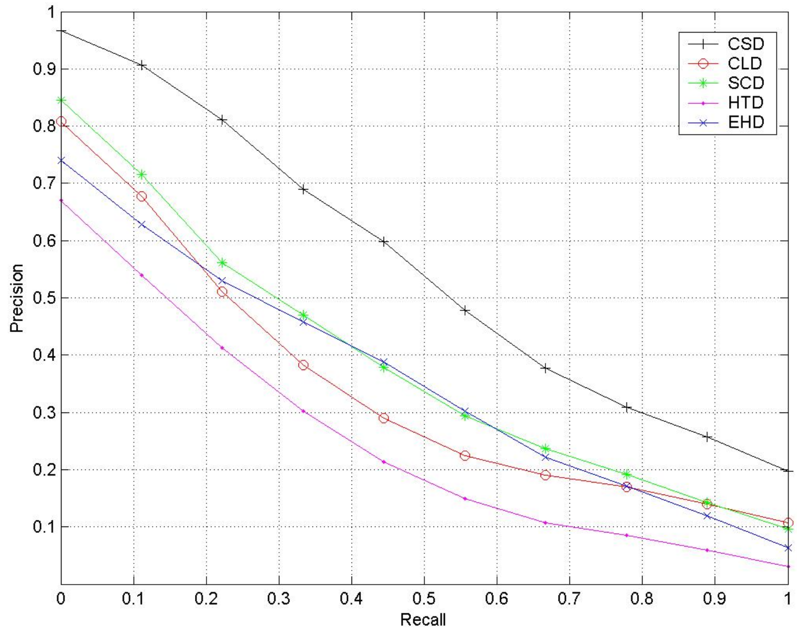


Figure 3.8: Comparison of all descriptors' results

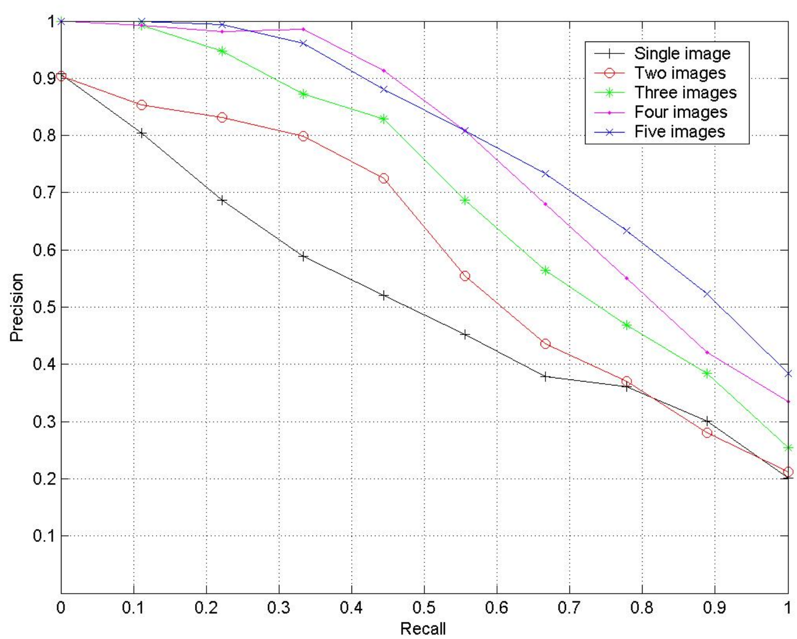


Figure 3.9: Comparison of multi-feature (CSD & EHD) results

retrieval results of the two-descriptor case, CSD and EHD, are illustrated. Finally, an example retrieval is demonstrated to show the effectiveness of the proposed approach.

Single Descriptor

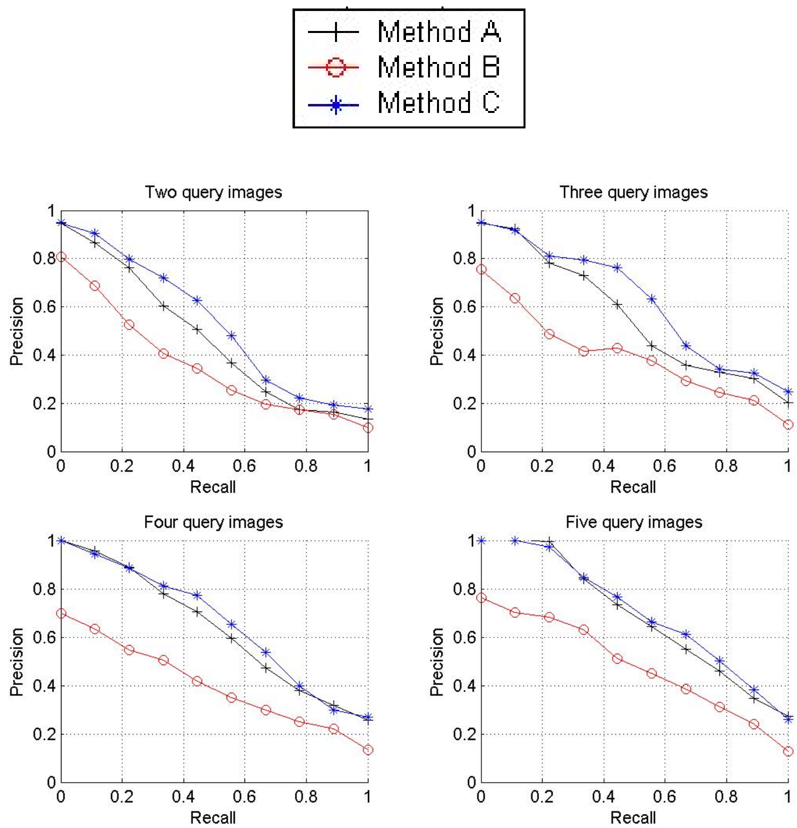
The proposed intra-level weights modification model is applied on the CSD. In Figure 3.10(a), it can be seen that when the number of query images equals three, the proposed approach can generally improve retrieval, making it around 15% better than method B, with same number of query images. On average over the 87 queries (Figure 3.10(b)), the newly proposed approach also performs the best amongst the three different approaches. For the CLD, when the number of query images equals four, our proposed approach can improve the retrieval performance by around 10% compared with method B. The average diagram also illustrates the effectiveness of the proposed approach. For the SCD descriptor, although the improvement is not as great as that of former two, it is still not worse than method A and B. From Figure 3.13 and Figure 3.14, it can be seen that our proposed approach can also improve the retrieval performance for texture descriptors by around 10% on average.

Multi-feature Descriptors

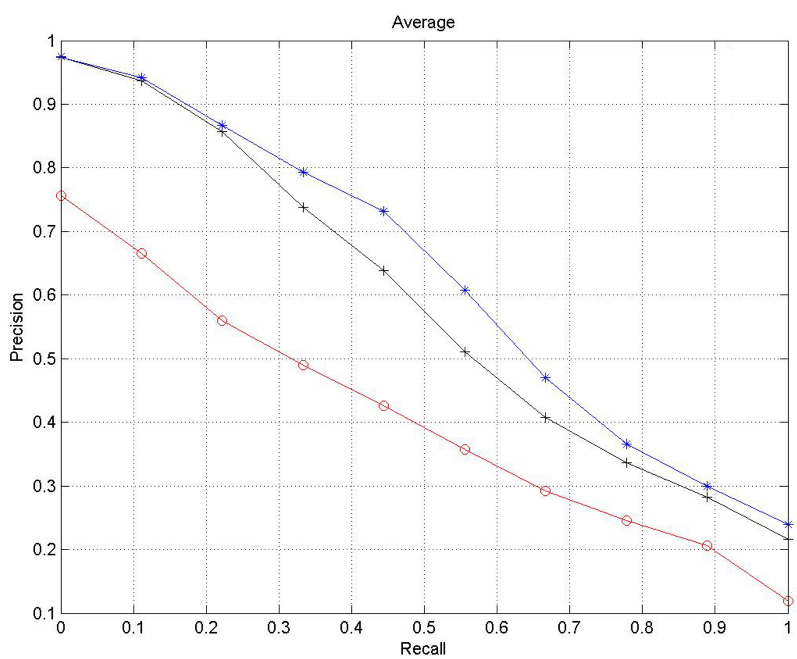
Besides testing the retrieval performance of single descriptors, multi-feature descriptors are also tested and evaluated to demonstrate the effectiveness of the proposed approach. Nevertheless, we still choose the CSD and EHD descriptors as the combination features. In Figure 3.15, it can be seen that by employing our proposed intra-level weights, retrieval performance is improved by around 7% over method and by around 15% over method B on average.

Examples

In Figure 3.16, we display a set of query images as an example: three yellow butterflies (query A). Since yellow is the common colour among the query images, we therefore estimate that the user expects to be shown yellow butterflies. In figure 3.17, the

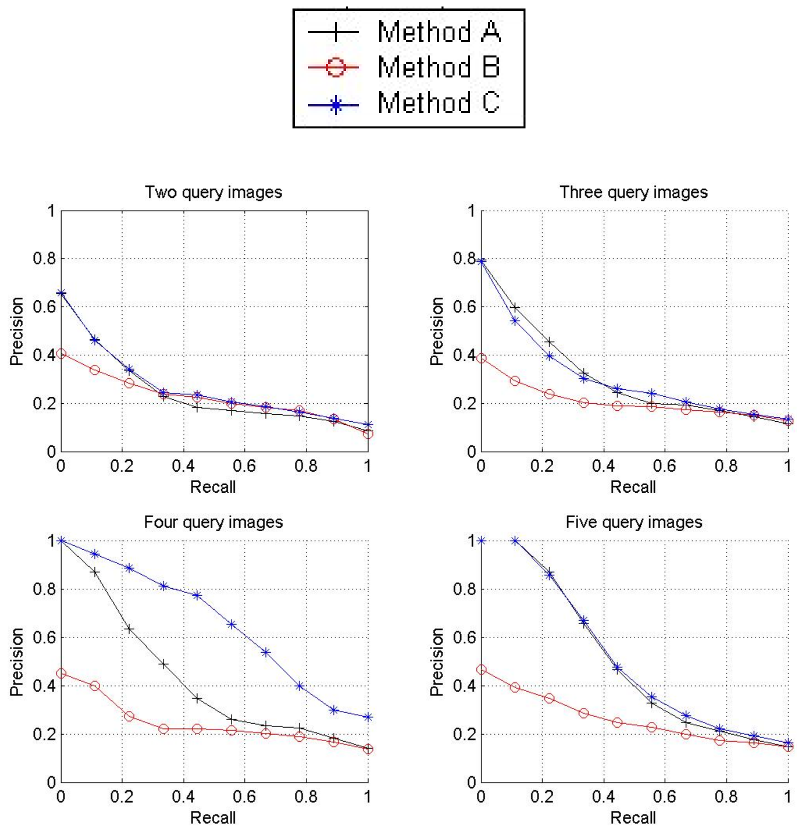


(a) Different number of query images

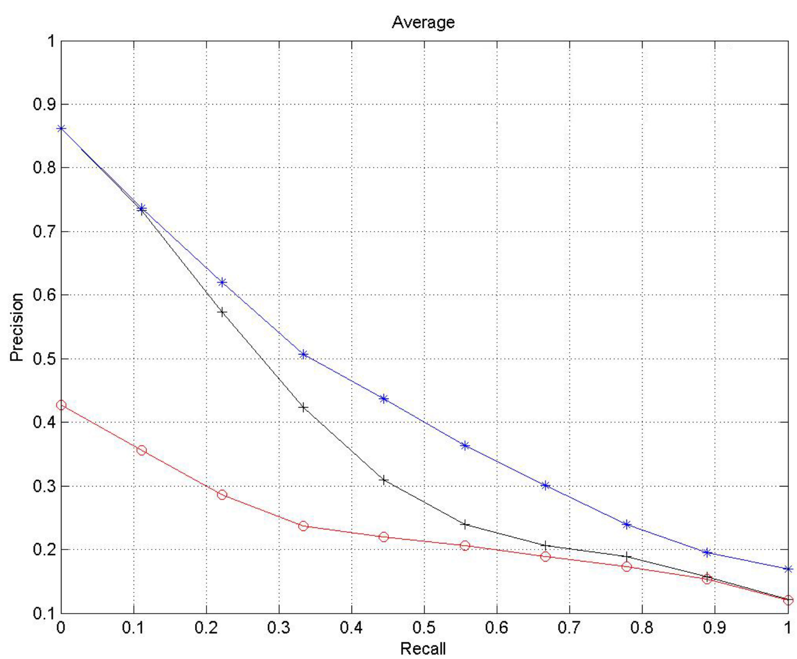


(b) Average

Figure 3.10: Comparison of CSD results

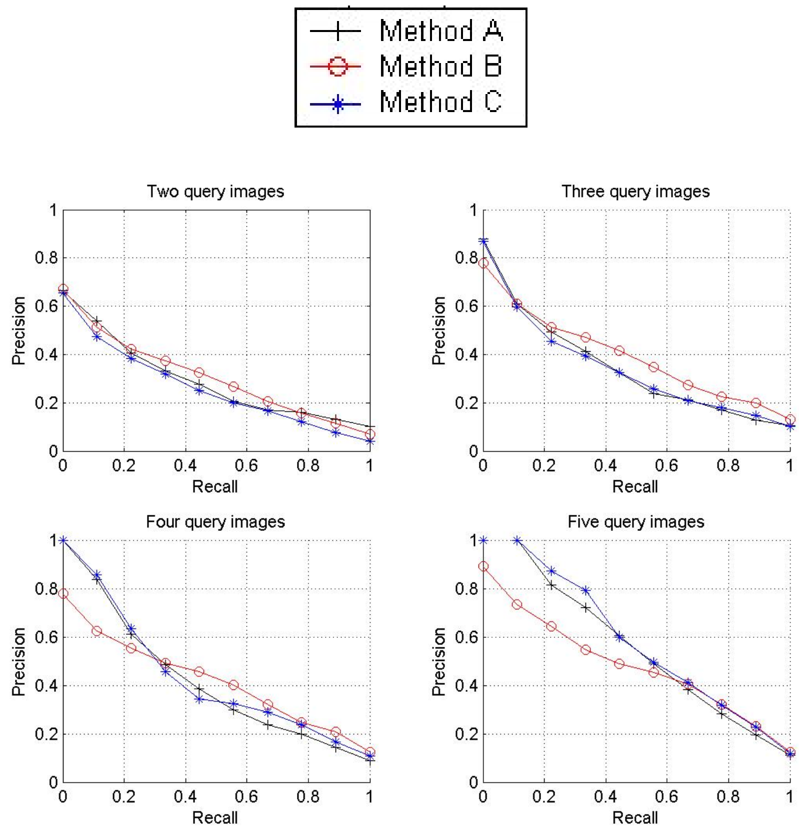


(a) Different number of query images

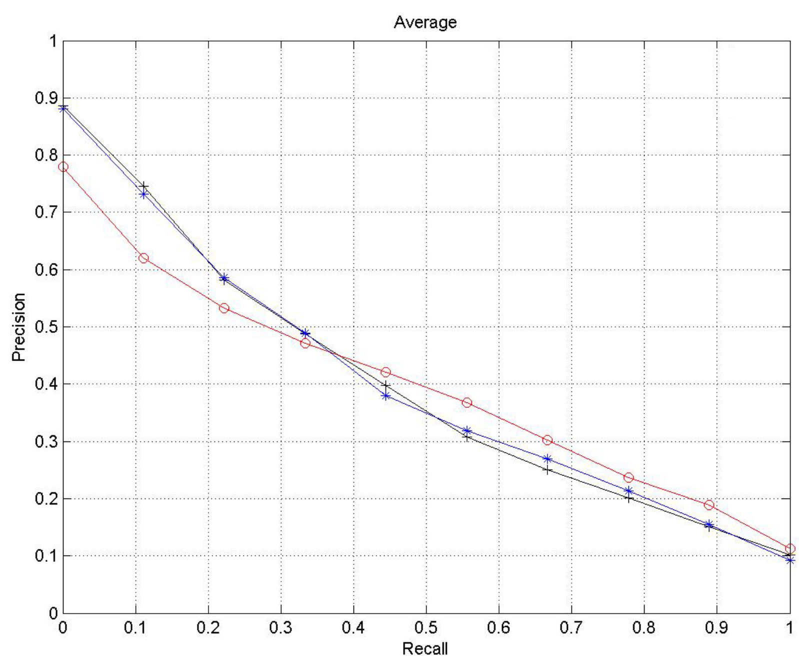


(b) Average

Figure 3.11: Comparison of CLD results

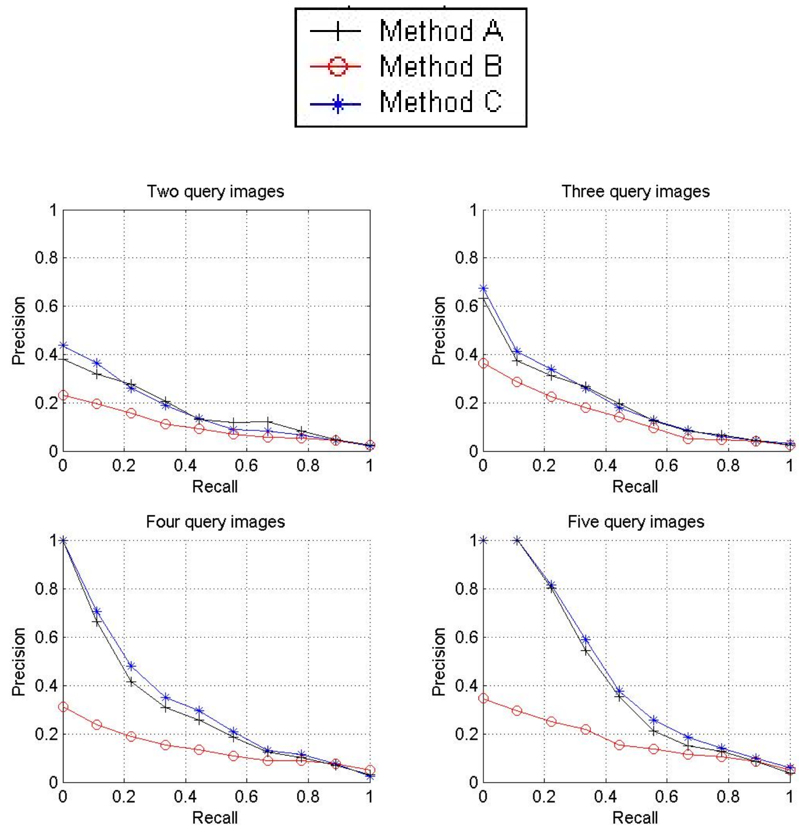


(a) Different number of query images

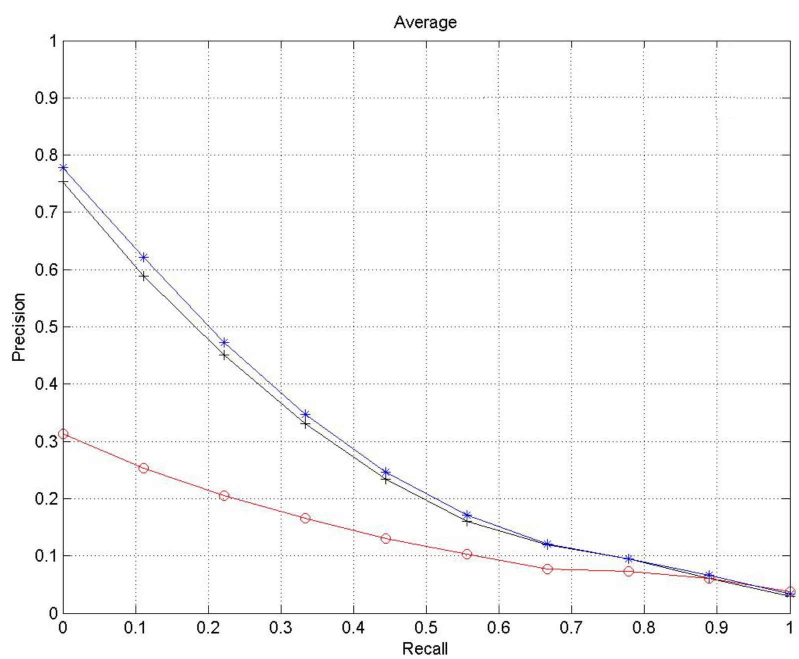


(b) Average

Figure 3.12: Comparison of SCD results

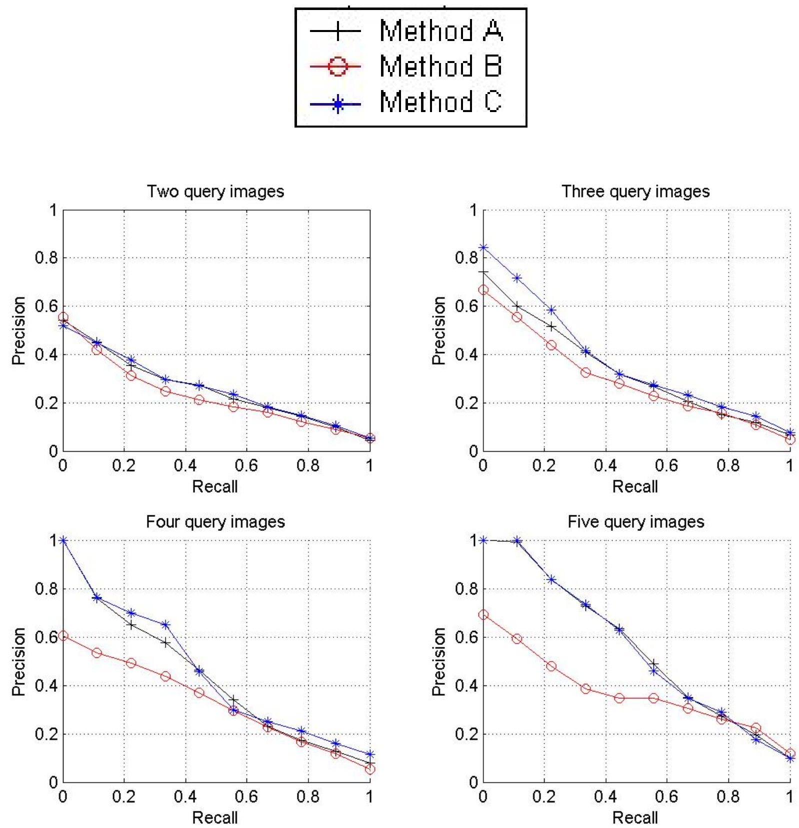


(a) Different number of query images

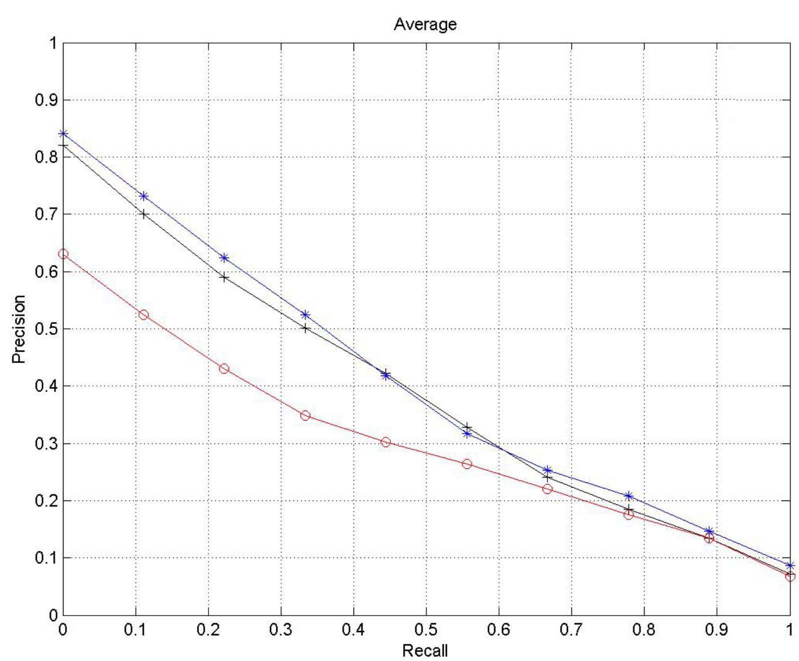


(b) Average

Figure 3.13: Comparison of HTD results

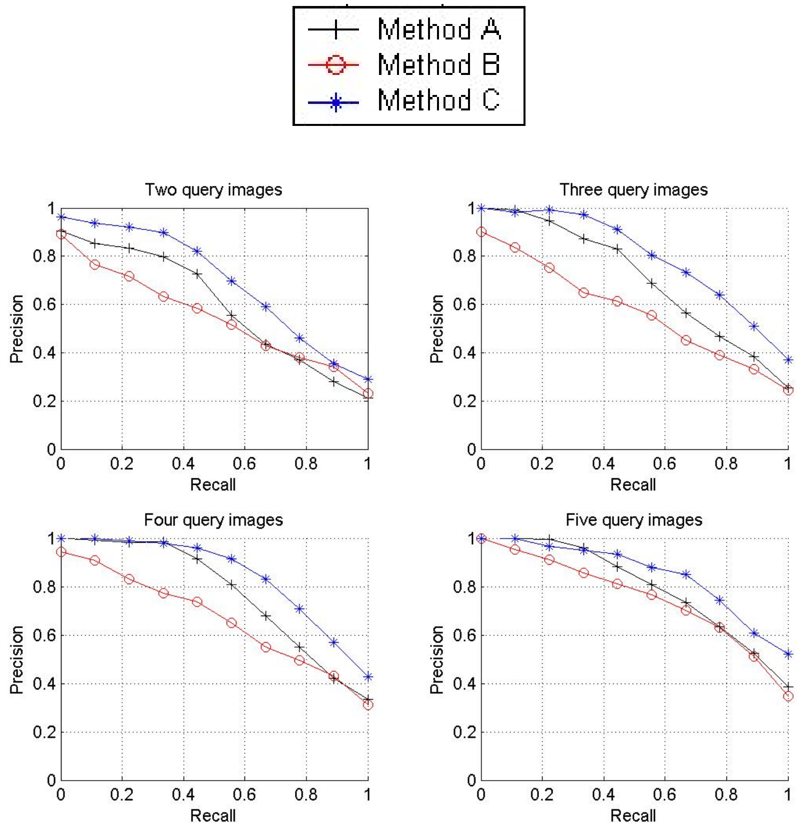


(a) Different number of query images

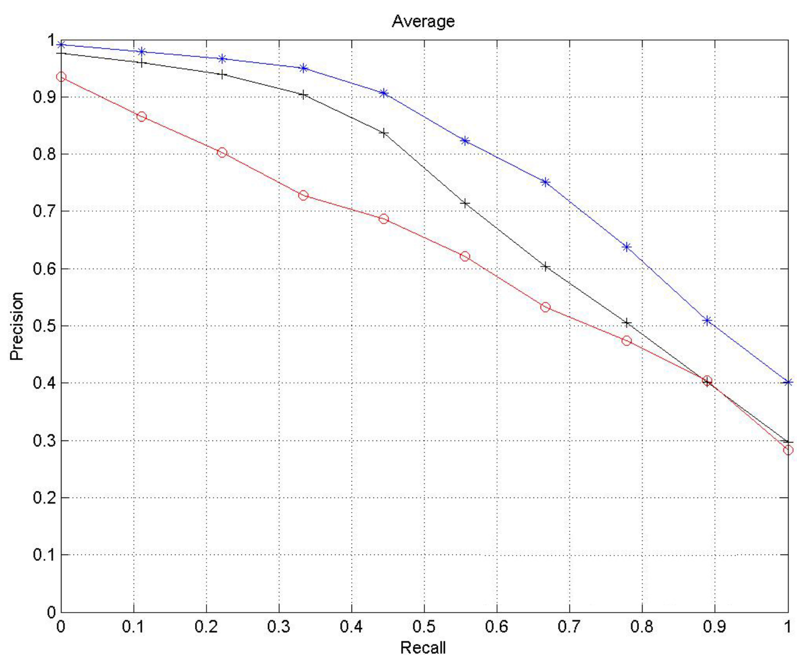


(b) Average

Figure 3.14: Comparison of EHD results



(a) Different number of query images



(b) Average

Figure 3.15: Comparison of Multi-feature (CSD & EHD) results

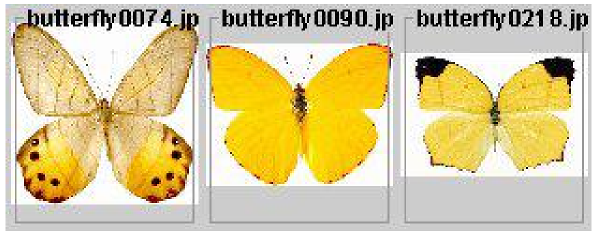


Figure 3.16: Query A: three yellow butterflies

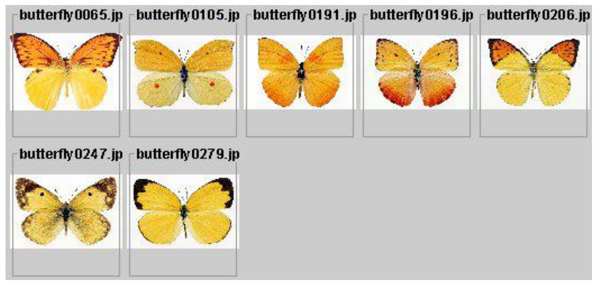
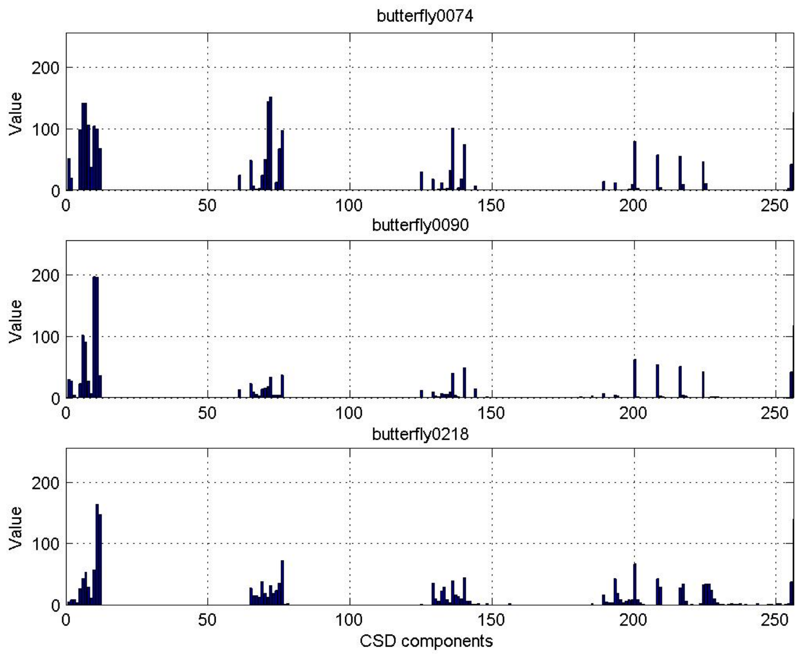


Figure 3.17: Query A: ground truth

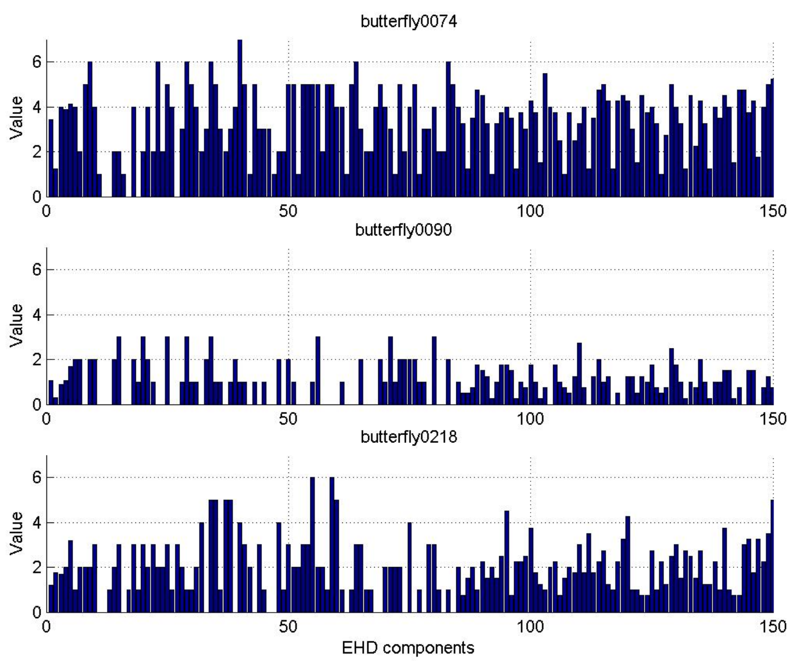
ground truth is displayed. In order to illustrate the improvement by using our proposed approach, we display the retrieval results step by step.

It is necessary to mention that the reason why we only select the same query (one dominant object with almost plain background) to be demonstrated is because image with only one dominant object on almost plain background can simplify the problem. It is convenient for the reader to find the common feature among the query images. The they can easily notice the expectation of the query and understand the physical meaning of the proposed method. The choice of query also make the experimental results comparable. Since we will demonstrate several experimental results with different approach and settings, by employing the same query in all of these experiments, the results are more comparable and the reader can see the difference among several approaches and the improvement achievable by the proposed approaches.

First, in Figure 3.18, the feature vectors of both the CSD and EHD are displayed. It is obvious that some components in the CSD and EHD are very similar among the feature vectors of the query images. Second, as shown in Figure 3.19, the mean and standard deviation of each component are calculated. It can be seen that the standard deviation of some components is zero, which will generate infinite weight in

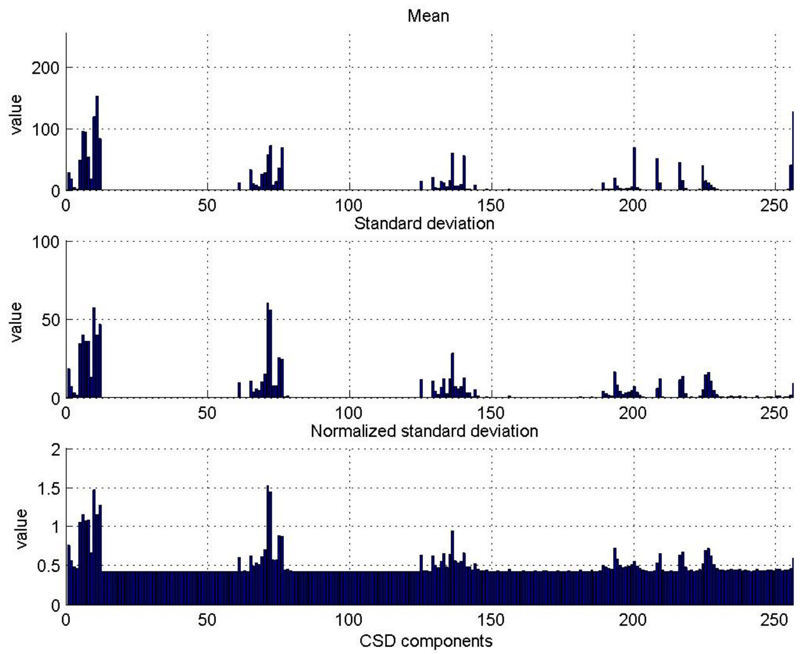


(a) CSD

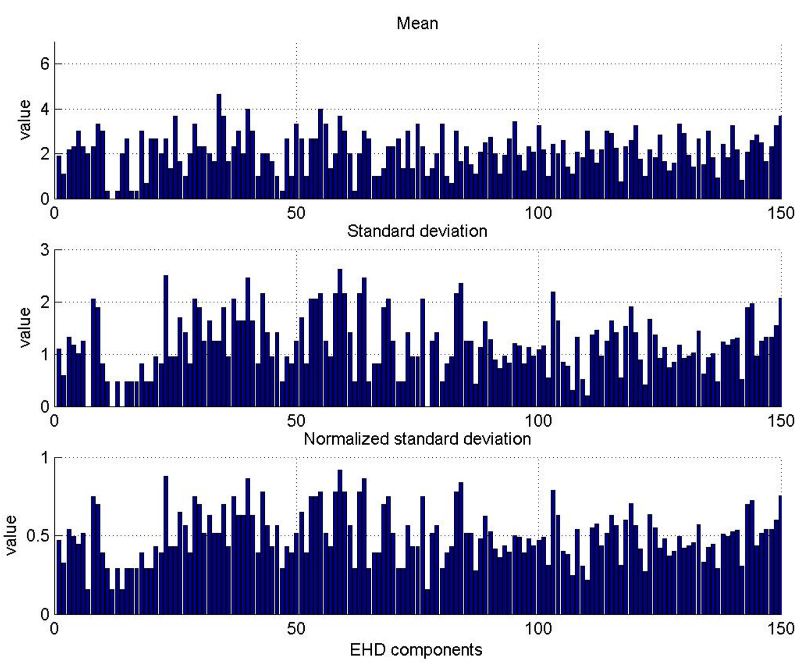


(b) EHD

Figure 3.18: Query A: feature vector

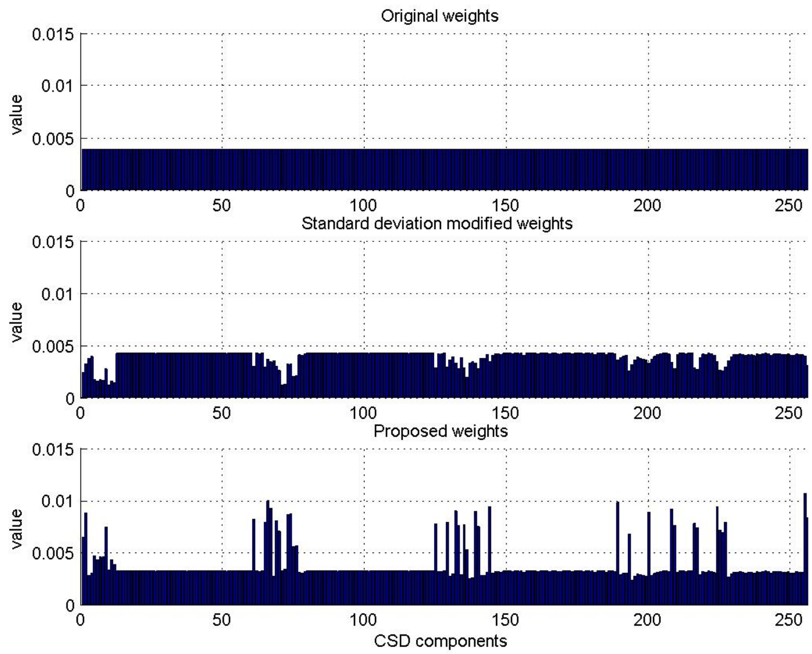


(a) CSD

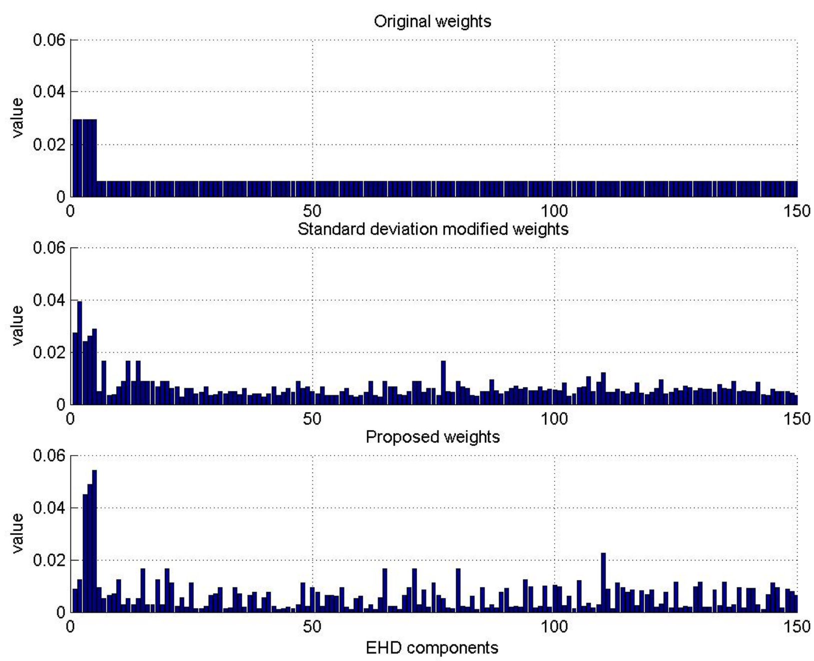


(b) EHD

Figure 3.19: Query A: feature vector's mean and std

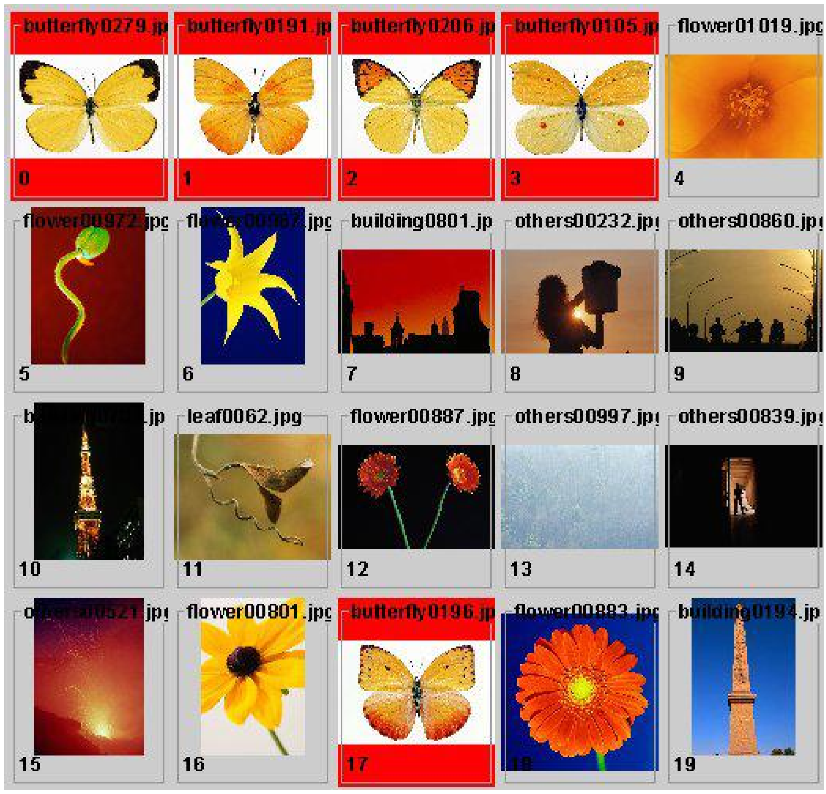


(a) CSD

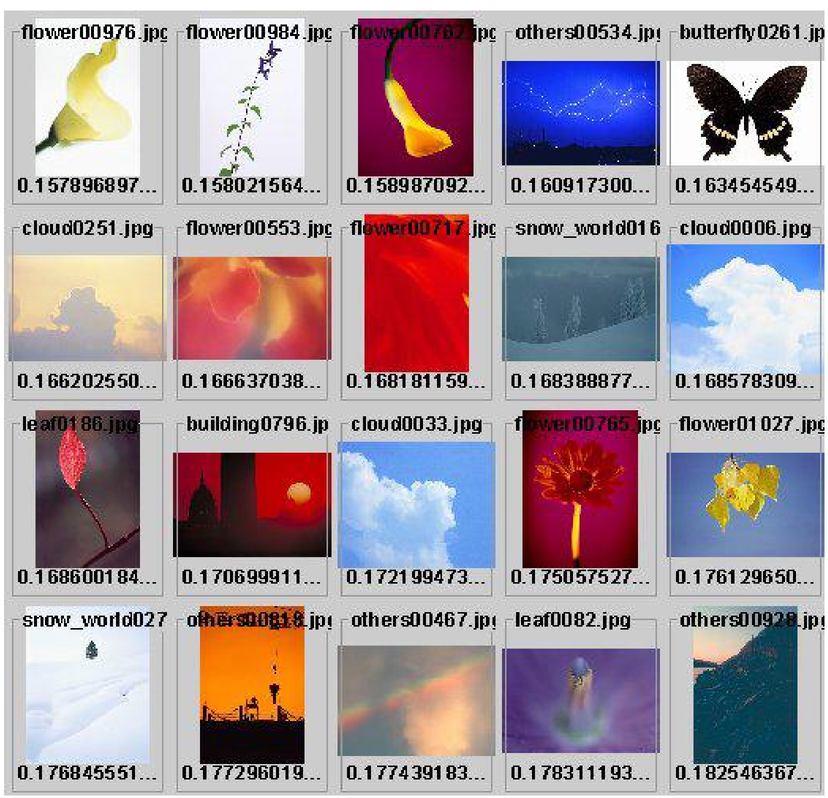


(b) EHD

Figure 3.20: Query A: feature vector's weights

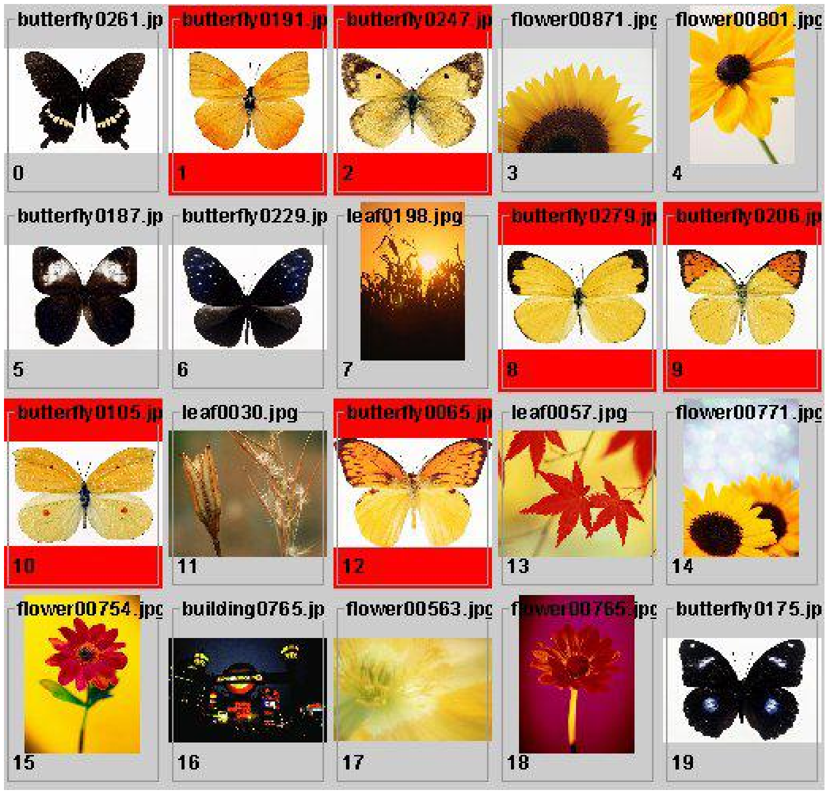


(a) Top 1 - 20

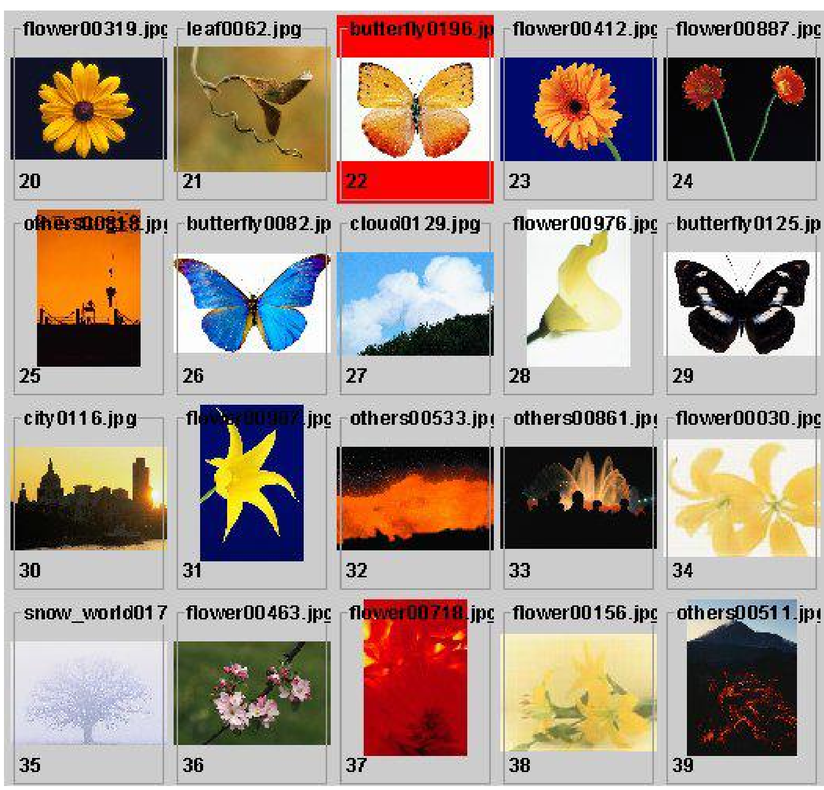


(b) Top 21 - 40

Figure 3.21: Query A: method A results

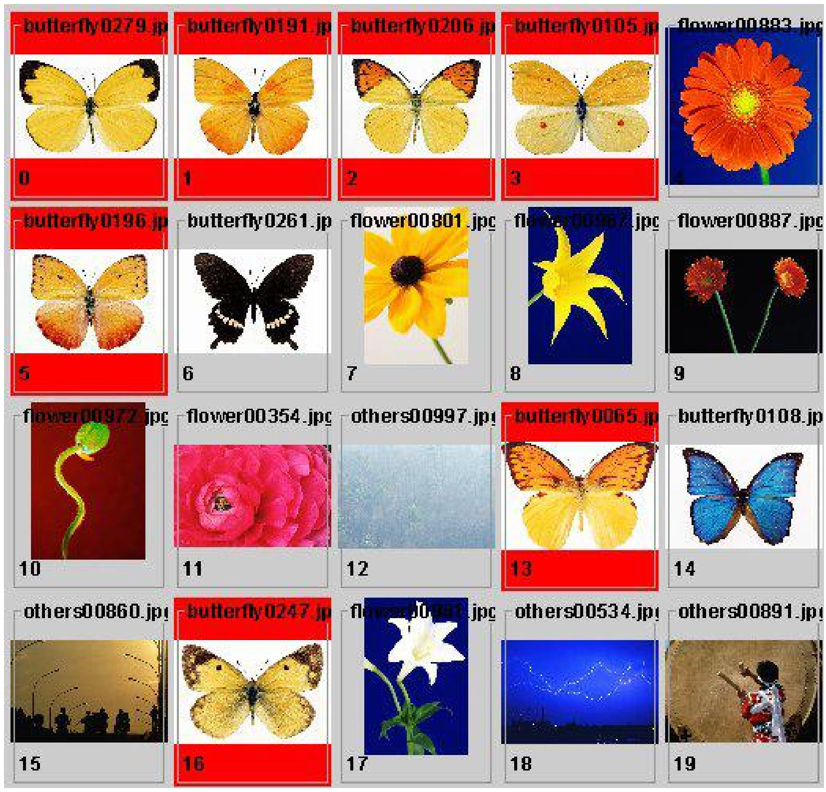


(a) Top 1 - 20



(b) Top 21 - 40

Figure 3.22: Query A: method B results



(a) Top 1 - 20

(b) Top 21 - 40

Figure 3.23: Query A: method C results

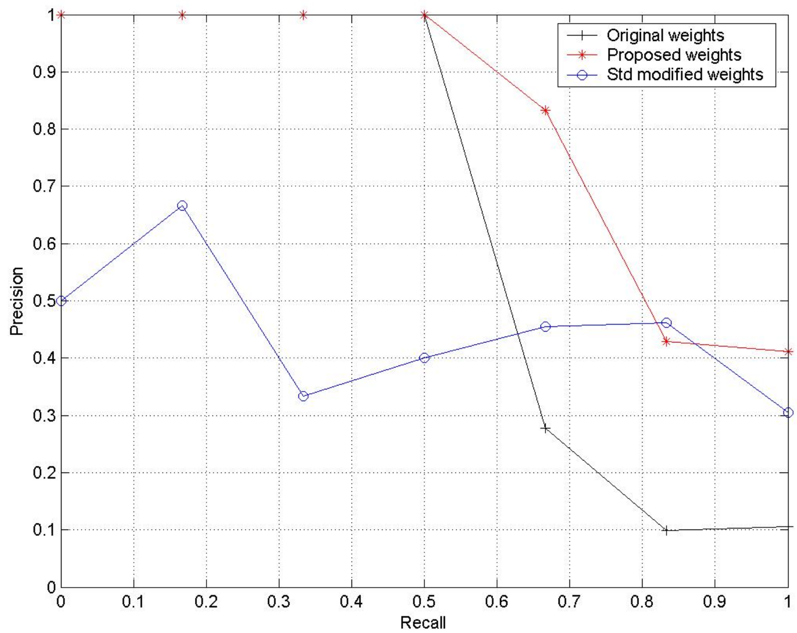


Figure 3.24: Query A: performance comparison

method B. The standard deviation of all components is normalised with the Gaussian normalisation. As shown in the three graphs in Figure 3.19, the problem is solved after normalisation. In Figure 3.20, all the weights obtained from the different approaches for both CSD and EHD are displayed. Comparing this with Figure 3.18, it can be seen that the weights for the components which are considered as significant among the queries are in fact decreased by only employing the standard deviation to modify them. On the other hand, by applying the approach proposed in this thesis, the weights for the significant components (”yellow colour” for the CSD and ”global texture” for the EHD) are increased, which accords with the users' expectations. The retrieval results with different weights are also displayed in Figures 3.21, 3.22 and 3.23, respectively.

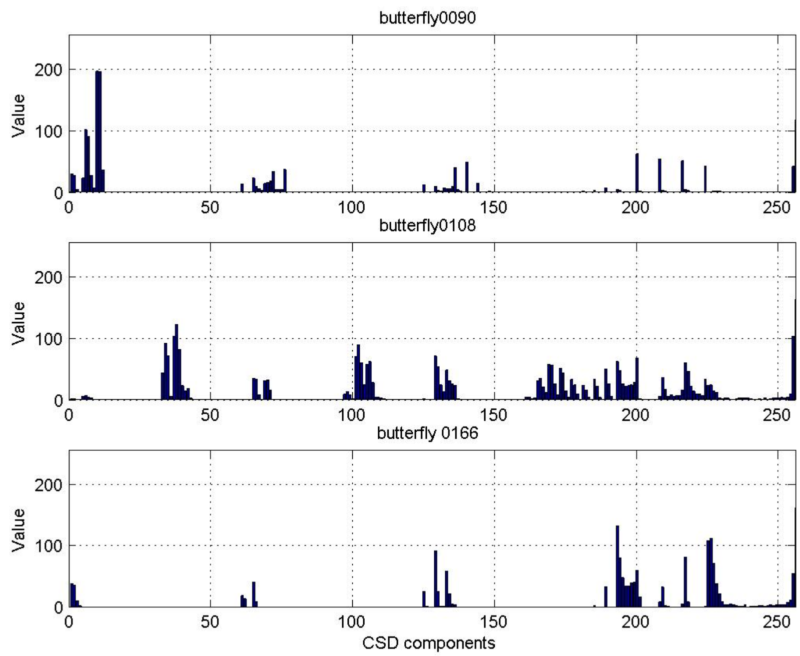
In Figure 3.21, we use the method A to retrieve and the retrieval results (top 40) are displayed. We can see that with the method A, the significant feature (yellow colour) cannot be detected, therefore only five ground truth images are found. In Figure 3.22, the standard deviation modified retrieval results are displayed. It can be seen that, since the weights for the significant components (yellow colour) within the query are

Figure 3.25: Query B: three yellow butterflies

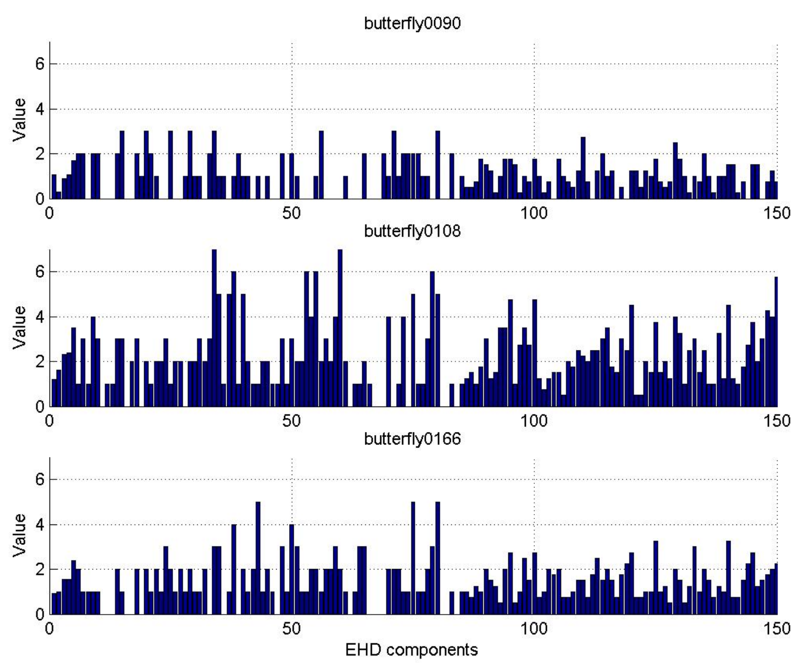
decreased, as shown in Figure 3.20, and although six out of seven ground truth images are found in the top 20, the first image in the results is nevertheless dissimilar to the query. This has negative impact on the performance in the precision-recall graphics shown in Figure 3.24. In Figure 3.23, the retrieval results in the proposed weights modification model are displayed. It is obvious that all the ground truth images are found in the top 20 and also that the ranking order is greatly improved, as shown in Figure 3.24.

In Figure 3.25, another query is displayed (query B). In this query, three butterflies with different colours are selected as the query images. We assume, therefore, that the user is more interested in both the butterfly's texture and contour shape. Butterflies with different colours are thus selected in the ground truth shown in Figure 3.26. In the ground truth, it can be seen that as well as yellow, blue and black butterflies, green, orange and grey butterflies also exist. According to the users' expectations, therefore, it is proposed that for the CSD, components representing "yellow", "blue" and "black" colours should be highlighted together; for the EHD, components representing "global texture of butterfly" should also be highlighted. In Figure 3.27(a), it is obvious that the CSD feature vectors are very different within the query, while the global texture of the query is similar, as shown in Figure 3.27(b). By employing the mean and normalised standard deviation for each component shown in Figure 3.28, the proposed weights for both the CSD and EHD are displayed in Figure 3.29. From Figure 3.29(a), it can be seen that with the proposed approach, the weights for the components (three different colours) are greatly increased, whereas in the previous approach, they were

Figure 3.26: Query B: ground truth

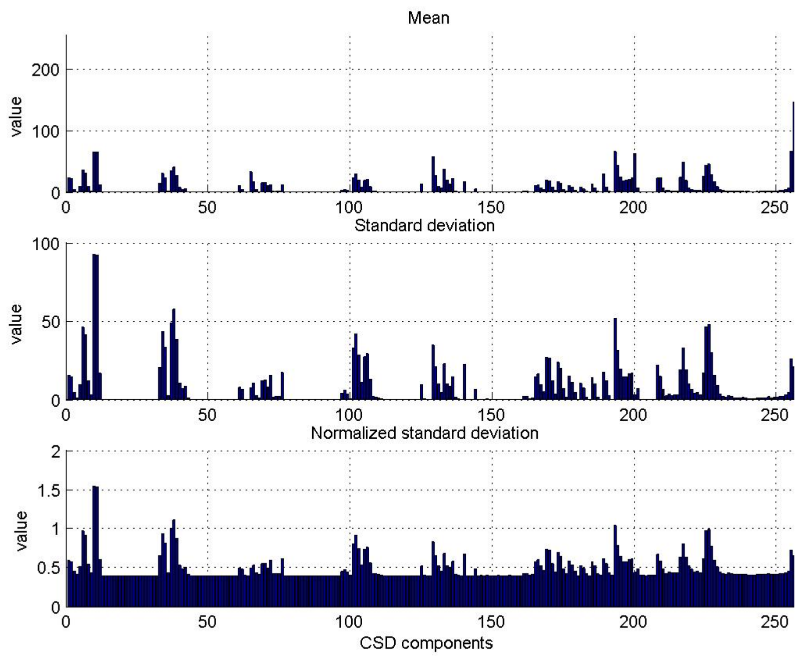


(a) CSD

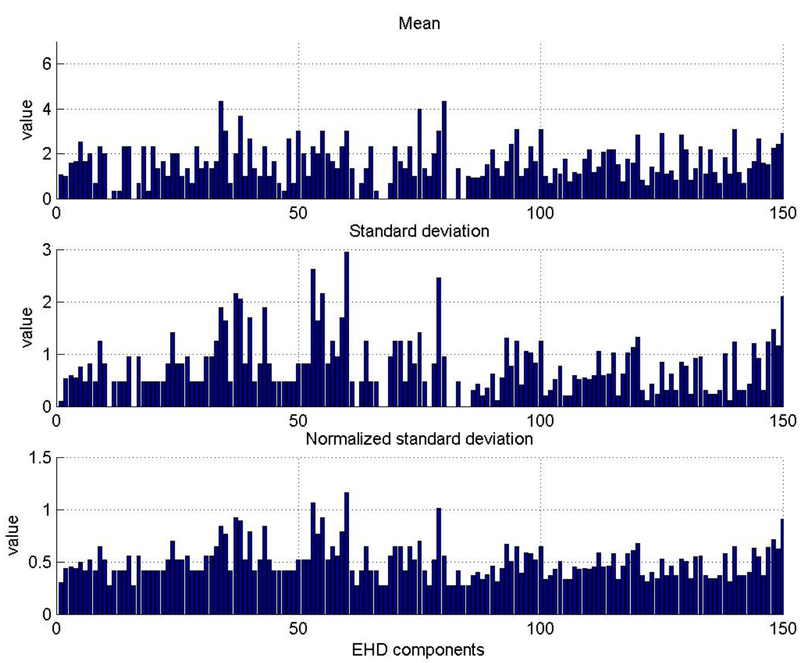


(b) EHD

Figure 3.27: Query B: feature vector

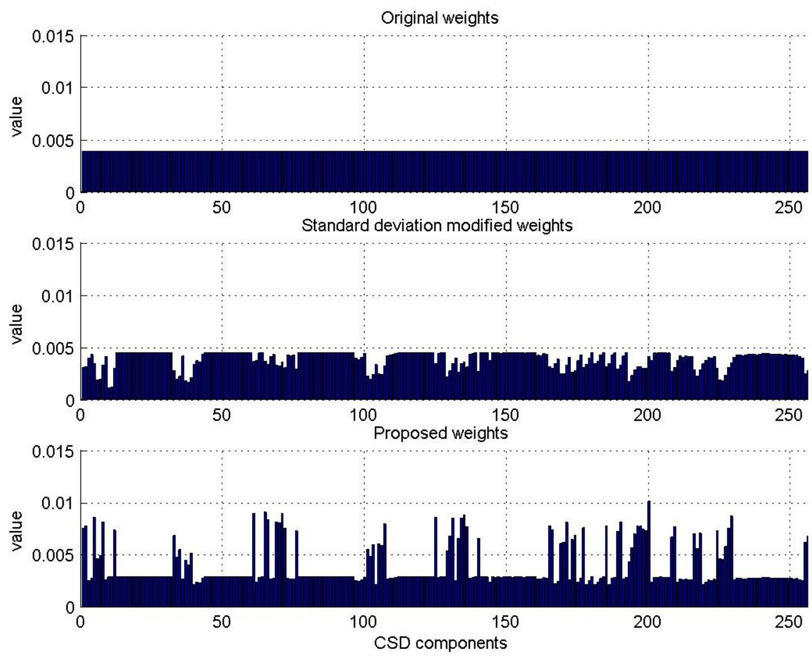


(a) CSD

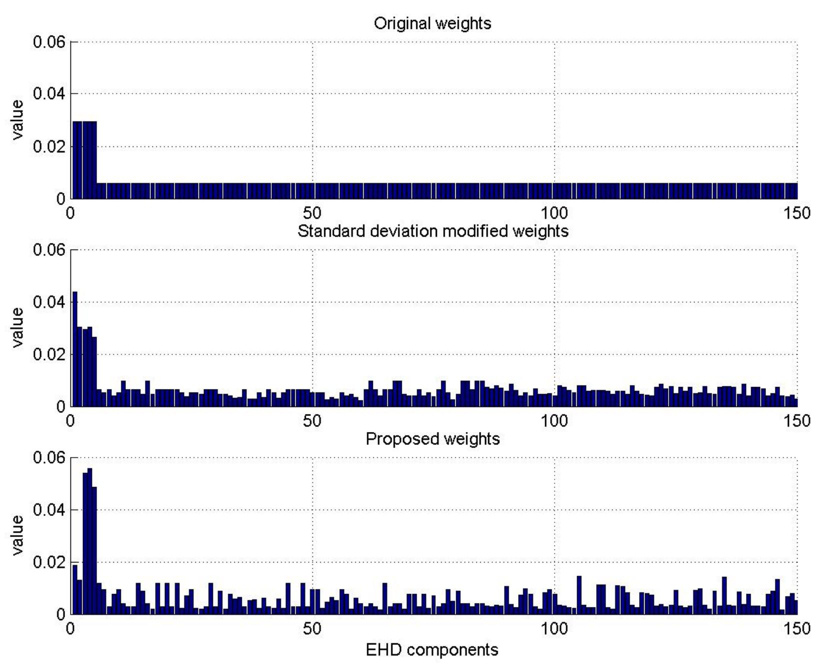


(b) EHD

Figure 3.28: Query B: feature vector's mean and std

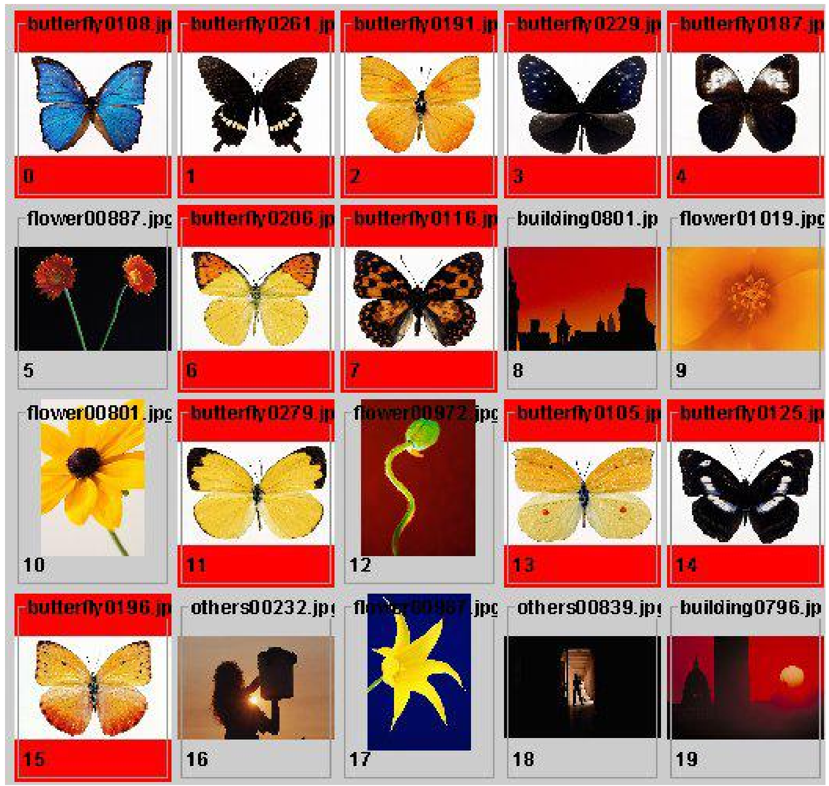


(a) CSD



(b) EHD

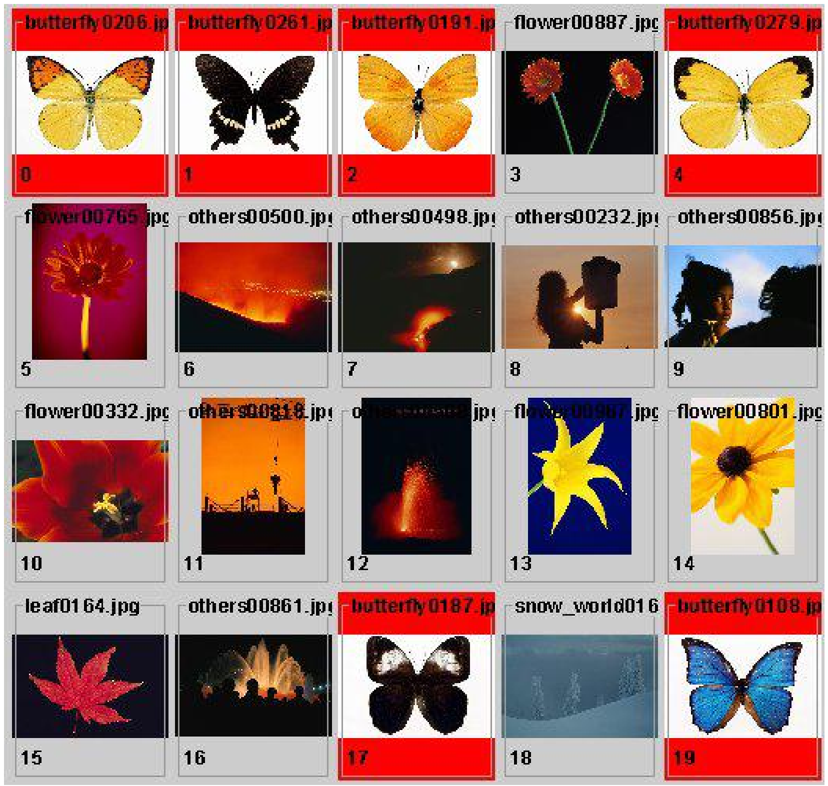
Figure 3.29: Query B: feature vector's weights



(a) Top 1 - 20

(b) Top 21 - 40

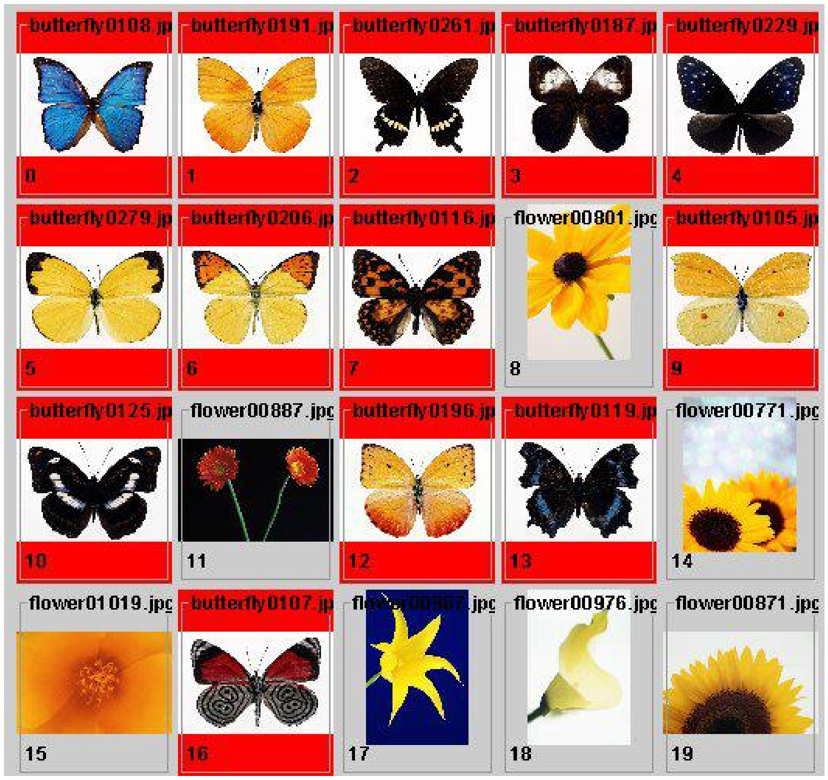
Figure 3.30: Query B: method A results



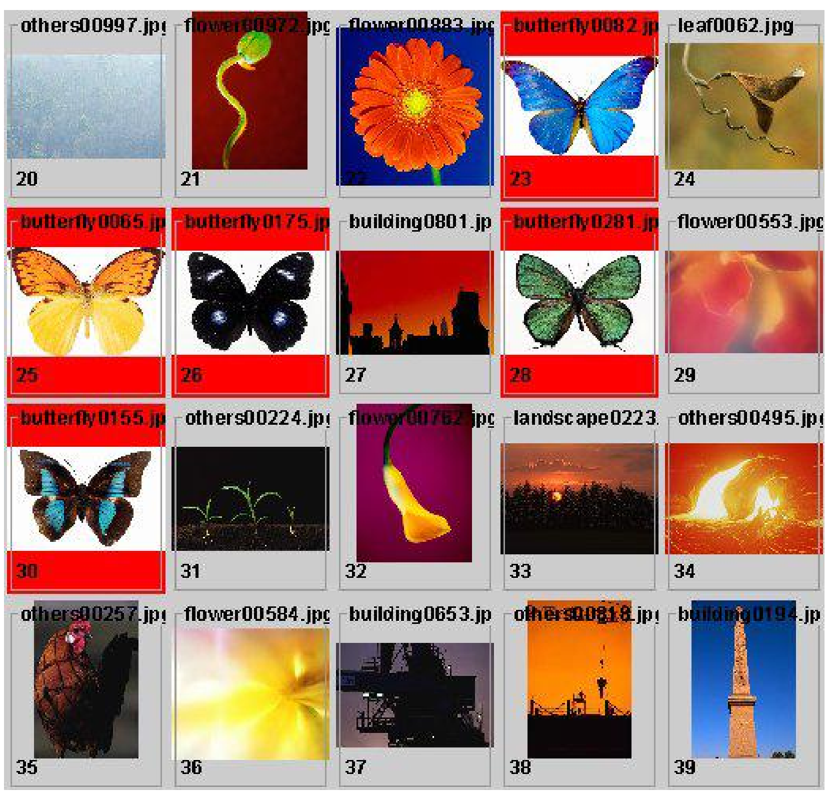
(a) Top 1 - 20

(b) Top 21 - 40

Figure 3.31: Query B: method B results



(a) Top 1 - 20



(b) Top 21 - 40

Figure 3.32: Query B: method C results

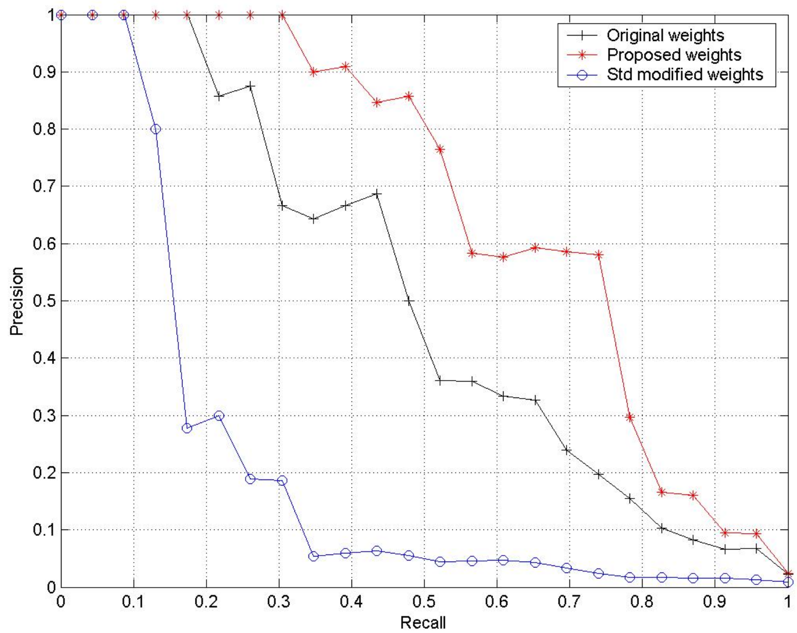


Figure 3.33: Query B: performance comparison

all decreased because of a larger standard deviation. For the EHD, it can also be seen that the global components' weights are twice as large as in the other two approaches. It can therefore be said that the proposed approach has the ability to estimate the users' expectations, and the computed weights can highlight significant components in the feature vectors.

The retrieval results with different weights for query B are also displayed. In Figure 3.31, it can be seen that because the weights for significant components are decreased, retrieval performance with the method B is even worse than the method A's. (Figure 3.30). In contrast with this, by employing the proposed method, shown in Figure 3.32, the retrieval results are better than those in Figure 3.30 and 3.31. Not only are the butterflies in yellow, blue and black found, but also the butterflies in green and grey are discovered. Therefore, the retrieval performance is greatly improved, as shown in Figure 3.33.

	CSD	EHD
Equal weights	0.5	0.5
Proposed weights	0.871	0.129

Table 3.1: Query A: feature weights

3.5.4 Inter-level Weights Modification

The Combination of Two Descriptors

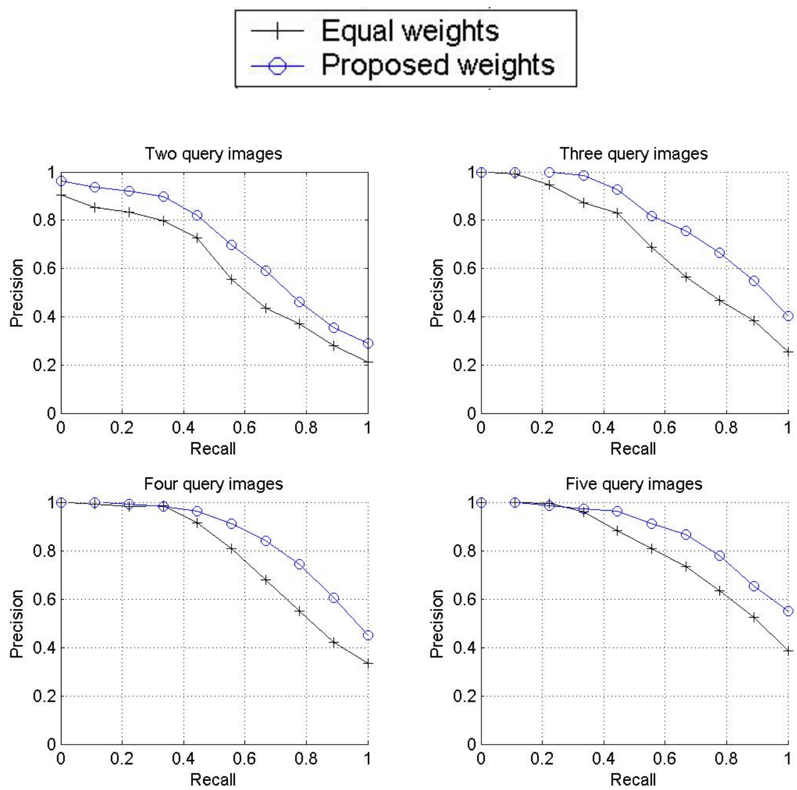
In order to test the performance of the proposed Inter-level weights, different descriptors are combined together. We tested the combination of two descriptors (one for colour and one for texture) and that of four descriptors (two for colour and two for texture). For two descriptors, the combinations of CSD & EHD and CSD & HTD are tested. The intra-level weights already been modified according to the proposed approach introduced in former section. The retrieval performances for each combination with a different number of query images are displayed in Figure 3.34 and Figure 3.35. From these experimental results, it can be seen that, with the proposed approach, retrieval performance can be improved by around 10% compared with equal weights for tested combinations of two descriptors.

The Combination of Four Descriptors

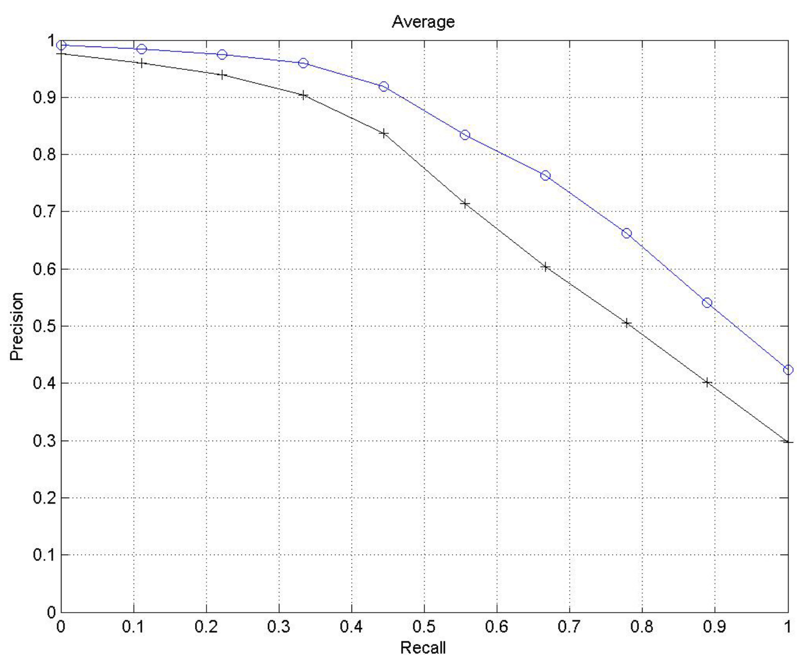
Two sets of descriptors, each containing four descriptors, the combination of SCD , CLD , EHD & HTD and CSD , SCD , EHD & HTD are tested. The retrieval performances for each combination with a different number of query images are displayed in Figure 3.36 and Figure 3.37. From the experimental results, it can be seen that, with the proposed approach, retrieval performance can be improved by around 5% compared with equal weights for tested combinations of four descriptors.

Examples

The query A is also employed to test the retrieval performance. First, we use the equal weighting to retrieve and the retrieval results (top 20) are displayed in Figure 3.38.

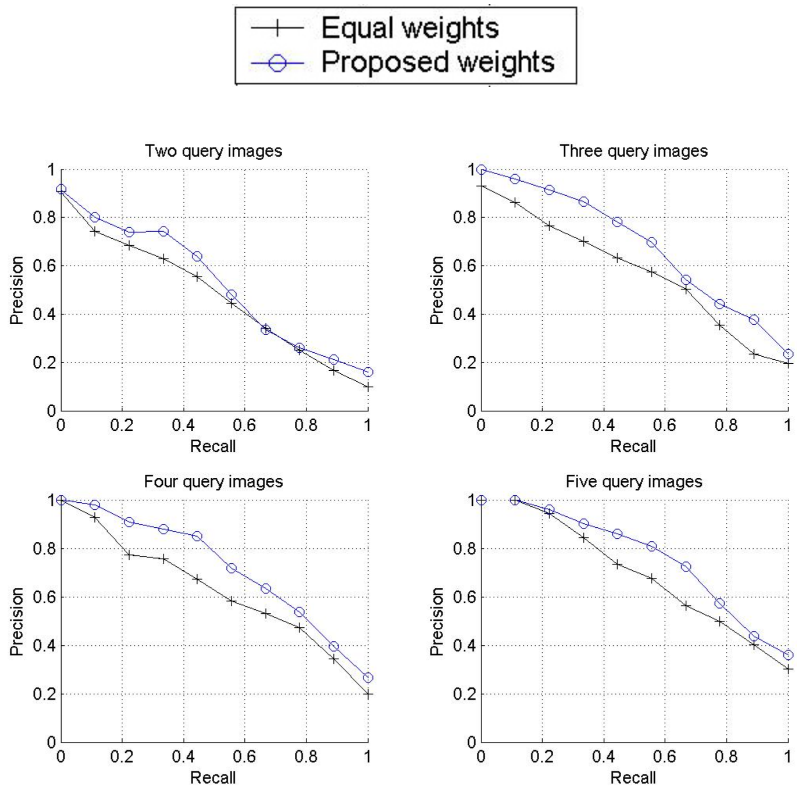


(a) Different number of query images

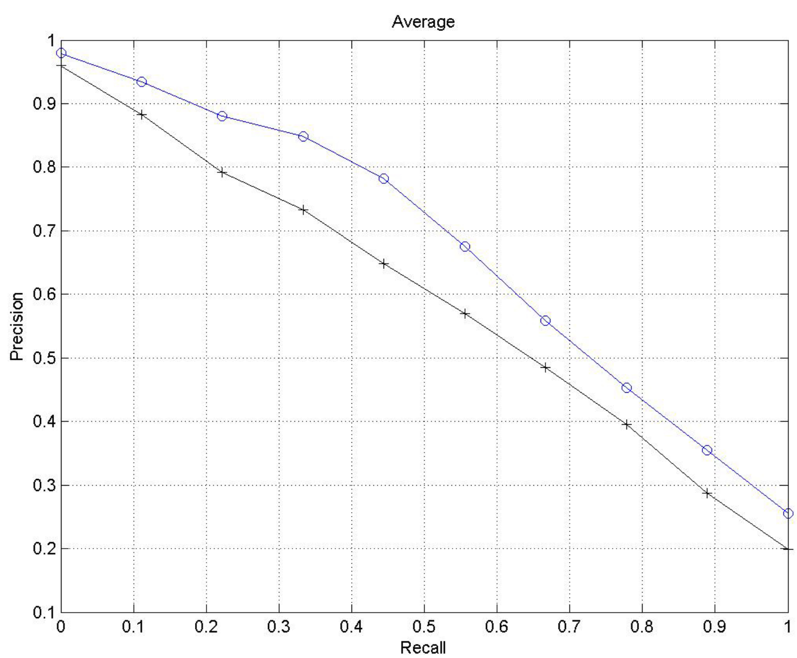


(b) Average

Figure 3.34: Comparison of CSD & EHD results

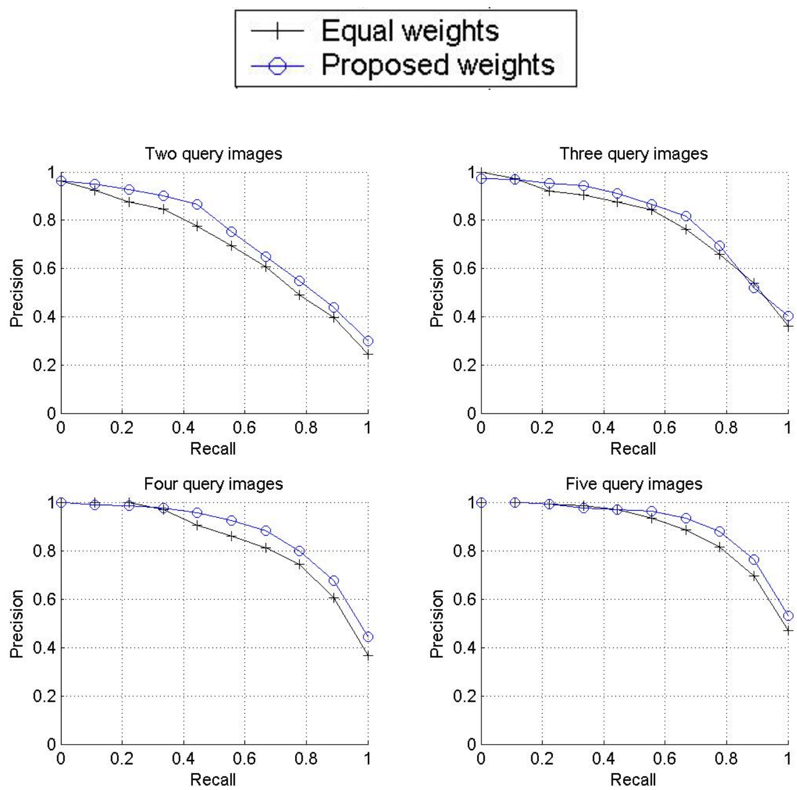


(a) Different number of query images

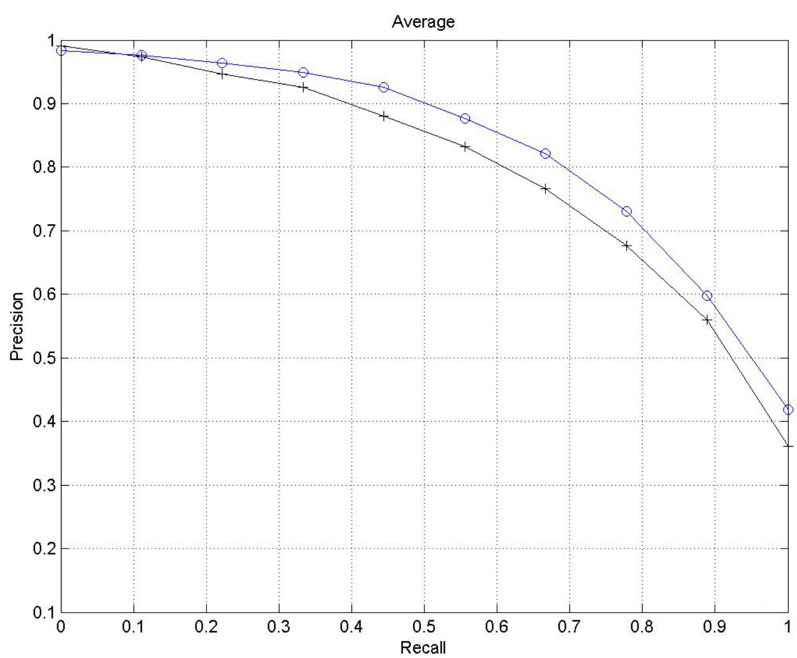


(b) Average

Figure 3.35: Comparison of CSD & HTD results

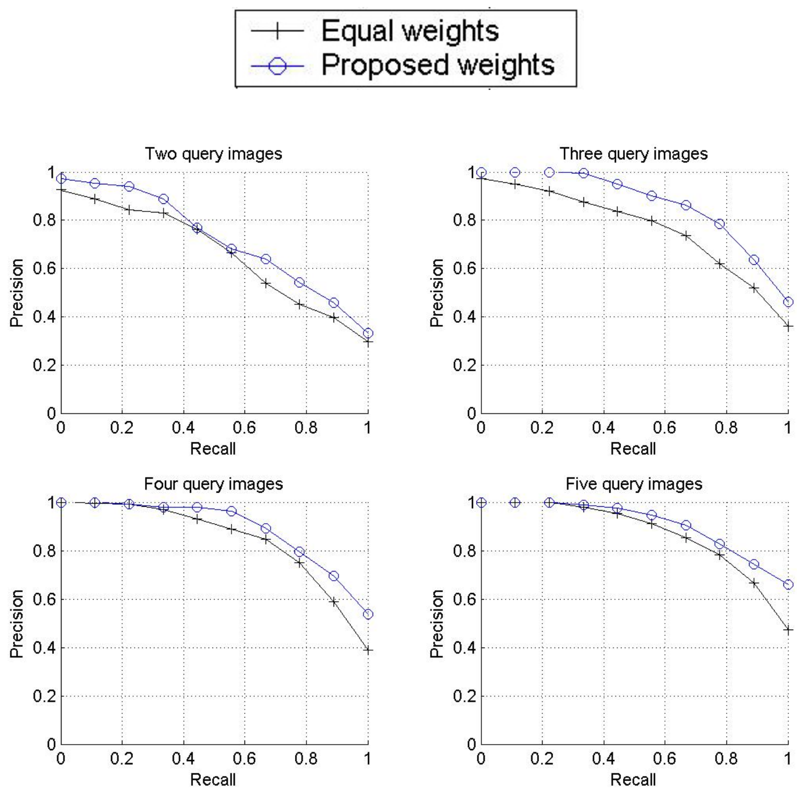


(a) Different number of query images

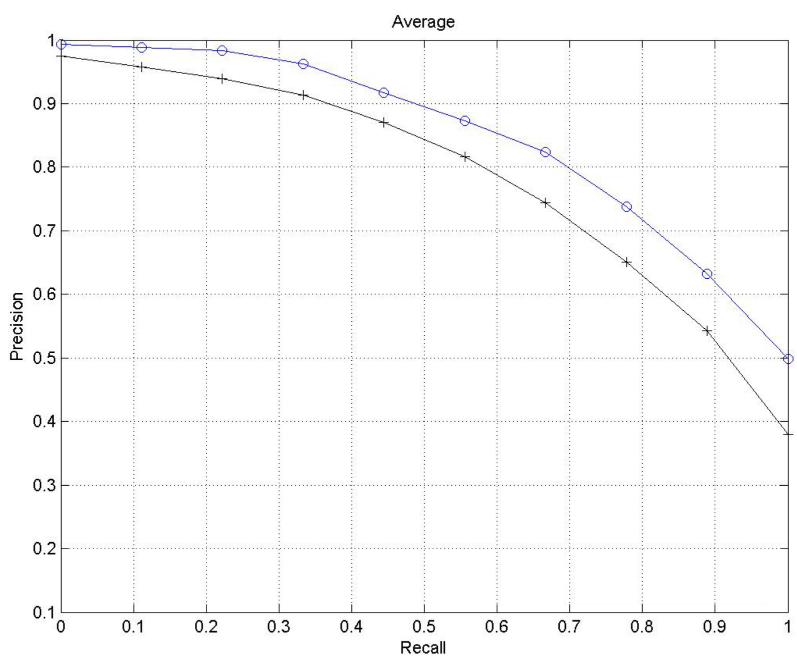


(b) Average

Figure 3.36: Comparison of CSD , SCD , EHD & HTD results



(a) Different number of query images



(b) Average

Figure 3.37: Comparison of SCD , CLD , EHD & HTD results

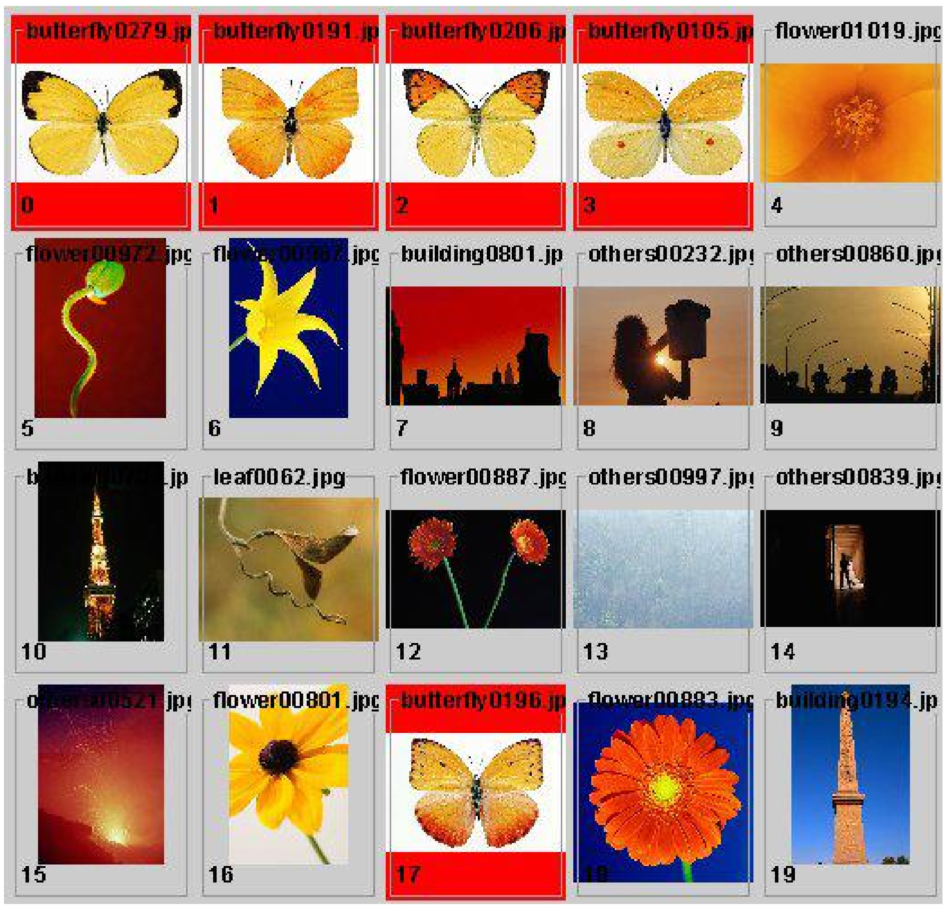


Figure 3.38: Query A: equal weights results (top 20)

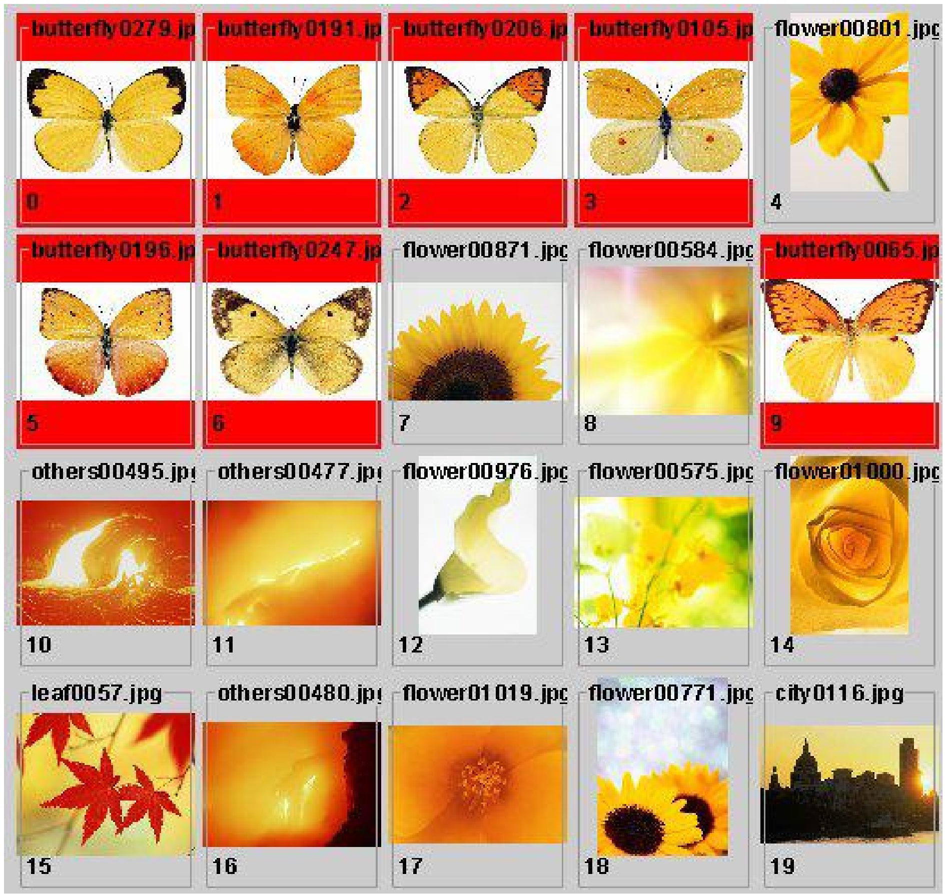


Figure 3.39: Query A: proposed weights results (top 20)

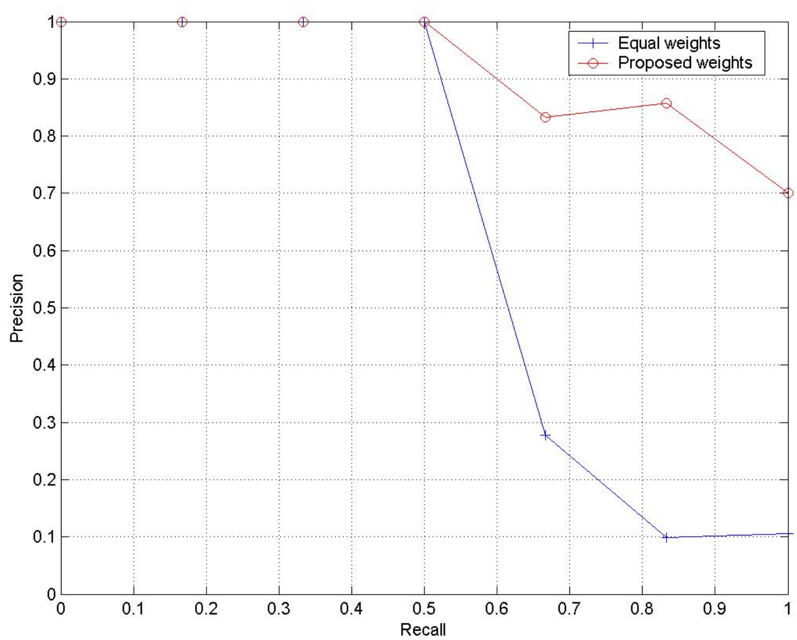


Figure 3.40: Query A: performance comparison

	CSD	EHD
Equal weights	0.5	0.5
Proposed weights	0.434	0.566

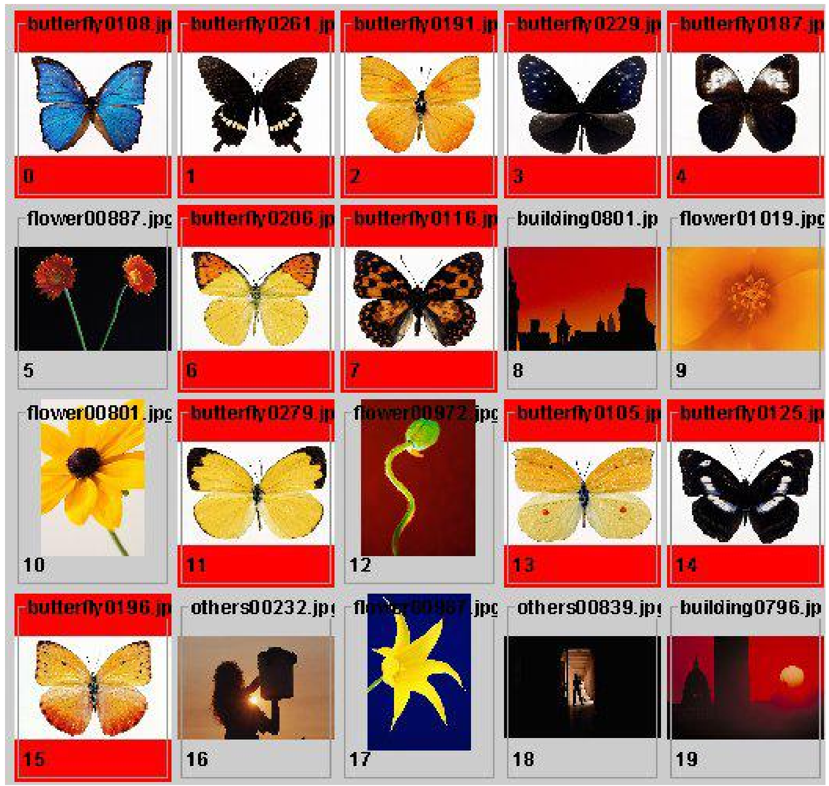
Table 3.2: Query B: feature weights

We can see that with equal weights, the significant feature (yellow butterfly) cannot be detected, therefore only five ground truth images are found. Second, by employing our proposed modification, the weights of the features are modified. As shown in Table 3.1, the weight for the CSD is greatly increased, which follows the expectation of the proposed approach. In Figure 3.39, the retrieval results with the proposed weights are displayed. It is obvious that all the ground truth images are found in the top 20. The precision-recall graphs for query A are also displayed in Figure 3.40. It can be seen that after applying the proposed weights, the retrieval performance is greatly improved.

The query B is also employed to test the retrieval performance. First, equal weights are applied in the retrieval process and the retrieval results are displayed in Figure 3.41. From the retrieval results it can be seen that, with equal weights, ground truth butterflies whose colour is different from the query images cannot be found in the top 40. The reason is that both the colour and texture are considered as having the same importance, therefore butterflies with different colour cannot be found. In order to solve this problem, the proposed combination weights are applied as shown in Table 3.2. Because the weight for texture is increased, ground truth butterflies with different colours from the query can therefore also be found and the retrieval performance is also improved, as displayed in Figure 3.43.

3.6 Chapter Summary and Conclusion

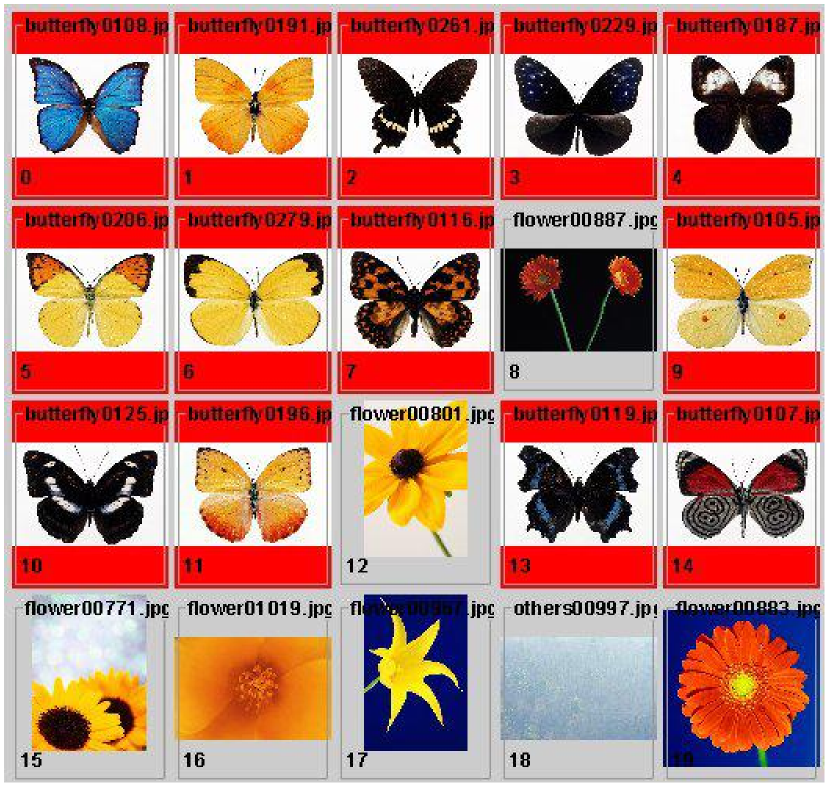
In this chapter, a new approach to adaptively modify the weights of the component of feature vectors and the weights assigned to feature vectors in a multi-image query CBIR is presented. Although no formal proof was given, the proposed models provide avenue to articulate the query-concept in the multi-image query. The method modulates both



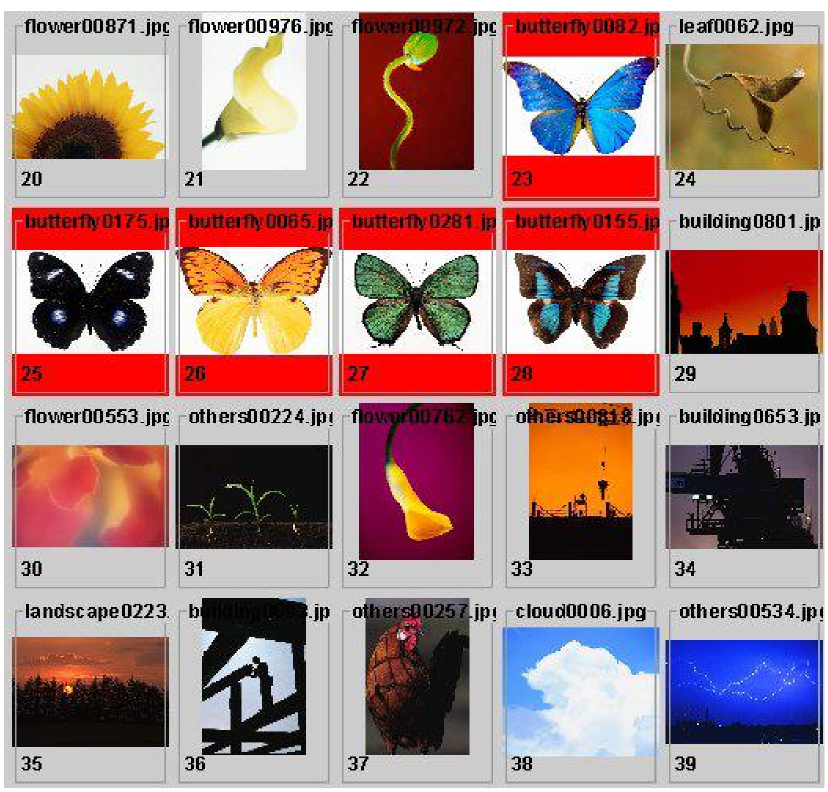
(a) Top 1 - 20

(b) Top 21 - 40

Figure 3.41: Query B: equal weights results



(a) Top 1 - 20



(b) Top 21 - 40

Figure 3.42: Query B: proposed weights results

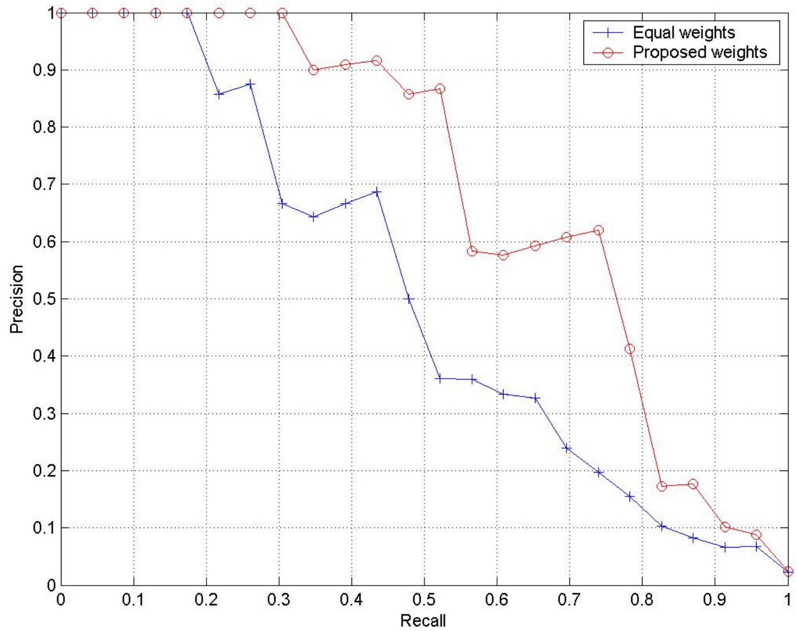


Figure 3.43: Query B: performance comparison

intra and inter feature weights of a dissimilarity metric for each individual query. By considering both the distribution and numerical value of each component of the feature vectors in the image query set, significant components are detected according to the expectations impled by the query image set, and the corresponding weight for each component is modified. Furthermore, by analysing the relationship among selected features in the query image set, significant features are captured by considering the perceptual significance as implied by the image query set, and the combination weights are modified to guide the retrieval process by an inclusive search. Experimental results show that the proposed method significantly improves the retrieval performance over previous methods.

Chapter 4

Applications of Multi-Image Query CBIR

4.1 Introduction

In this chapter, a CBIRS (Content-Based Image Retrieval System) developed as an application of the proposed approach will be introduced. Some examples are given to demonstrate the system performance.

4.2 An Application of Multi-Image Query CBIR

4.2.1 System Overview

CBIRS (Content-Based Image Retrieval System) is an image search engine which supports low-level feature based image retrieval. CBIRS employs both colour and texture features to retrieve images in the database. Six descriptors are employed to describe images. For colour features, the Colour Structure Descriptor (CSD), Colour Layout Descriptor (CLD), Scalable Colour Descriptor (SCD) and HSV Histogram Descriptor (HHD) are employed. For texture features, the Edge Histogram Descriptor (EHD) and Homogeneous Texture Descriptor (HTD) are used. These feature descriptors were introduced and described in Chapter 3. CBIRS supports both single and multiple image queries. For the multi-image query, by employing the scheme proposed in Chapter 3, both the intra and inter-level weights are modified by analysing the query images presented by the user. In the following paragraphs, the structure of the system and merits will be introduced.

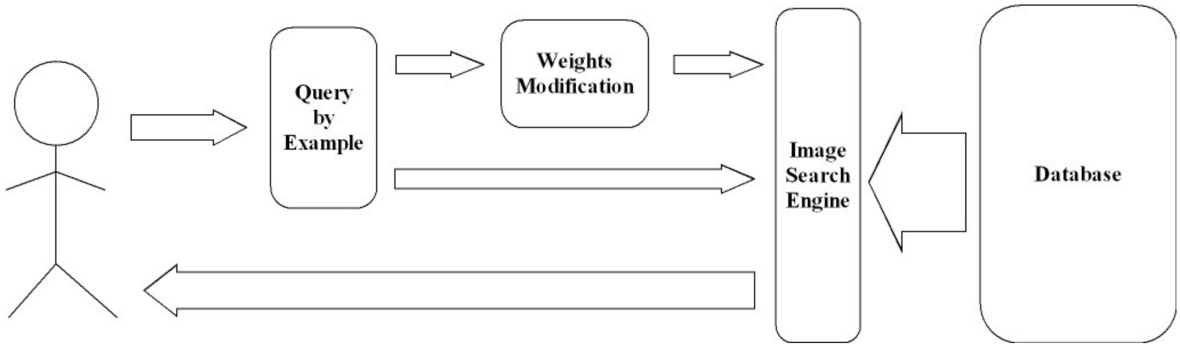


Figure 4.1: CBIRS: system overview

4.2.2 System Structure

As shown in Figure 4.1, the structure of CBIRS contains four main blocks: query-by-example block, weights modification block, image search engine block and database block.

Query-by-Example Block

In CBIRS, examples are used to search for images in the database. The user can select single or multiple images as the example query and submit them to retrieve similar images from the image database. In this block, each query image's feature vectors, for all selected feature descriptors, are extracted and forwarded to both the weights modification block and image search block for further analysis and usage.

Weights Modification Block

When feature vectors of the query images are transferred to the weights modification block, both of the intra and inter-level weights are generated dynamically by the system. The greatest advantage of this CBIRS is that by analysing the features of the query images, the query concept implied by the query images and the selected features can be used in the retrieval process. The weights for both the intra and inter-level searches are modified according to the feature analysis, which guide the retrieval process to be "inclusive". The weight modification block implements these functions. When the user submits the query, the weight modification block will operate in two modes depending on the number of query images. In the case of the single image query, equal weights will be used in both the intra and inter-level search. However, in the case of the multi-image query, according to the approaches introduced in Chapter 3, the modified weights are generated and sent to the image search engine block for the single features' similarity calculation and features' combination. Furthermore, if the user is not satisfied with the retrieval result, other kinds of combinations can also be employed and the new combination weights are sent to the Image Search Block once again for an inter-level search.

Database Block

In this CBIRS, all the feature descriptors of images in the database are extracted in advance. Each image is described by six feature descriptor vectors. In the retrieval process, these vectors, not the original images, are used to calculate the visual distance between the query image/s. Then, according to the distance value, images from the database which are similar to the query will be displayed to the user. By extracting the feature vectors in advance, the system can obtain each image's representations and generate the retrieval results quickly.

Image Search Block

In CBIRS, the process of similarity matching and image retrieval is carried out by the image search engine block. Once the query is submitted to the system, the feature extractor will represent all the query images in the form of feature vector. Based on the predefined feature similarity measurement and the proposed weight modification schemes presented in chapter 3, the similarity between the database images and the query are calculated. Finally, the similarity value are normalized by the Gaussian Normalisation, then they are ranked and sent back to the user.

4.2.3 Demonstration

In this section, four examples are given to illustrate the retrieval performance of the CBIRS. The database contains 14,613 images and the size of the images range from 170 × 128 pixels to 3721 × 3086 pixels. The images in the database are collected in two ways. 1) the real-world collection images (9084 images), which are classified into 21 categories, such as animal, beach, building, flower, car, waterfall etc. 2) the MPEG-7 standard image database, which contains 5529 images from video frames and classified into categories such as human, building and landscape. Both the CSD and EHD descriptors are employed in this experiment. The proposed weights modification schemes 3 are employed to adjust intra and inter-level weights according to query images. The retrieval results illustrate the efficiency of CBIRS.

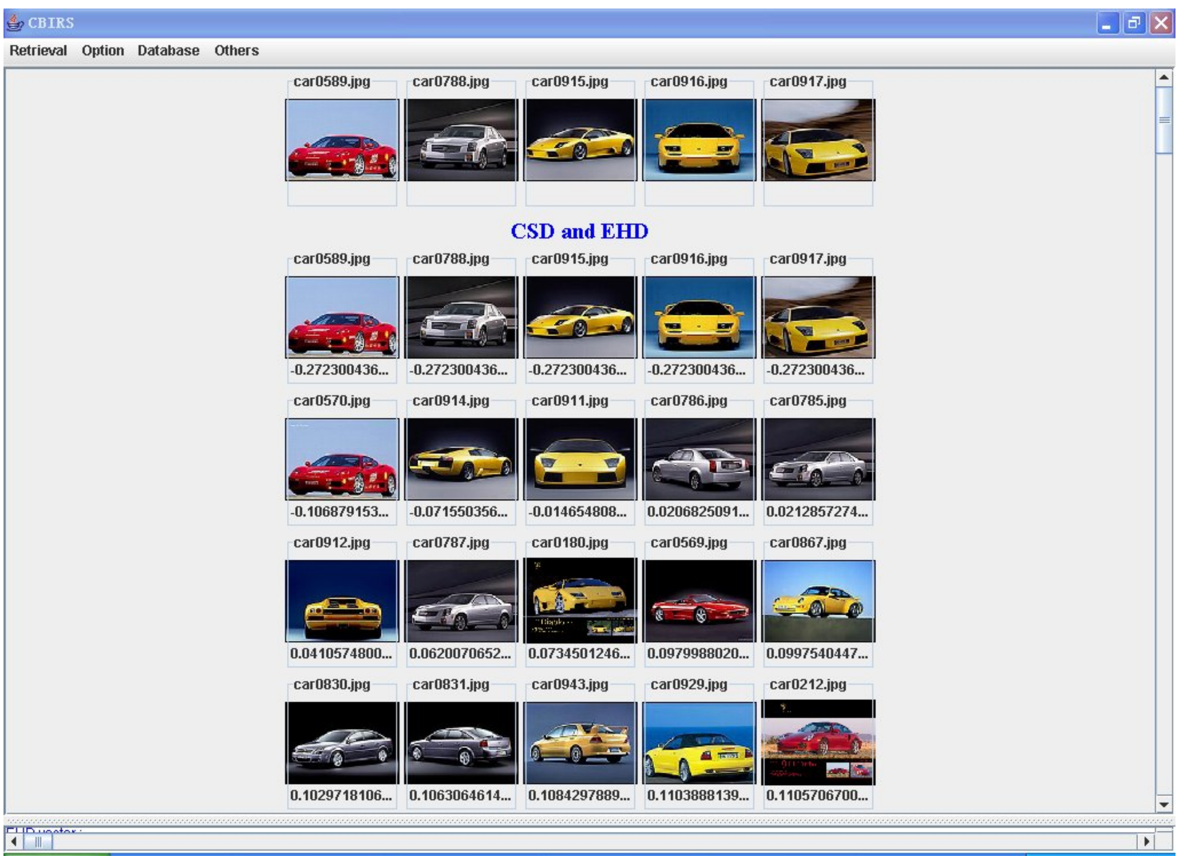


Figure 4.2: Five cars

Firstly, an example query of five cars in three different color (red, white and yellow) is demonstrated in Figure 4.2. In the result, it can be seen that cars in these three colors

	CSD	EHD
Weights	0.347	0.653

Table 4.1: Five cars' combination weights

are found and ranked on the top. Since the query images' color feature characteristics is different and texture feature characteristics is relatively similar, a higher combination weight is assigned to the texture as shown in table 4.1. In the next example, two query images will be removed from the query set, and the proposed system is able to capture this change and modify the combination weights accordingly.

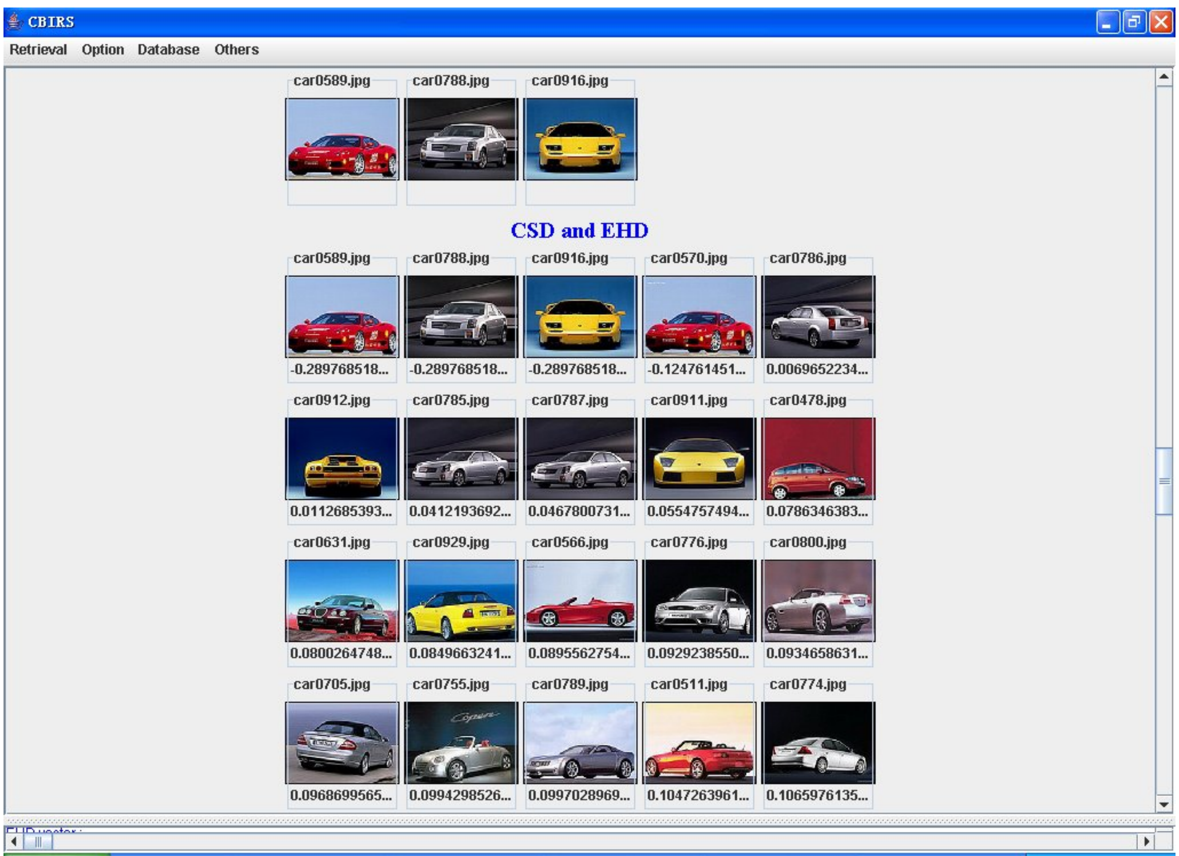


Figure 4.3: Three cars in different color

	CSD	EHD
Weights	0.224	0.776

Table 4.2: Three different color cars' combination weights

In the first case, two images of yellow car ($car915.jpg$ and $car917.jpg$) are removed from the query set. Compared to the original query set, the color feature characteristics of new query images set are relatively different. So it is expected that the weight assigned to color feature should decrease. In Figure 4.3, the retrieval result with new query is displayed. Also, it can be seen that cars in different colors are found and the combination weight for color feature, as shown in table 4.2, is decreased as expected (around 10 percent decrease).

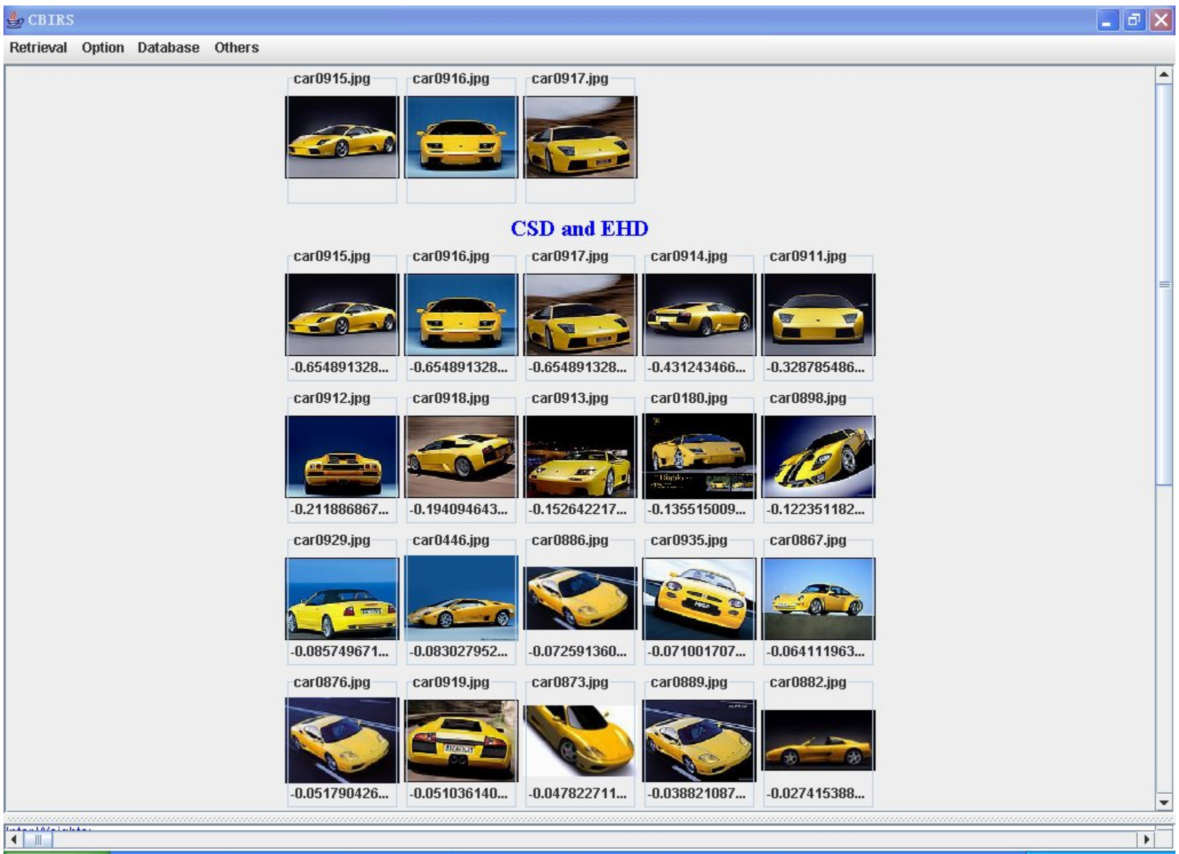


Figure 4.4: Three cars in similar color

	CSD	EHD
Weights	0.907	0.093

Table 4.3: Three similar color cars' combination weights

In the second case, the first two query images ($car0589.jpg$ and $car788.jpg$) are removed from the original query set. The remaining three query images share a similar color (yellow). Therefore, it is expected that the combination weight for color feature should be increased. In Figure 4.4, the retrieval result is illustrated. It can be seen that cars in yellow color are ranked at the top. The reason is because the proposed system dynamically modifies the combination weights and the weight for CSD is increased to more than 0.9 as shown in table 4.3.

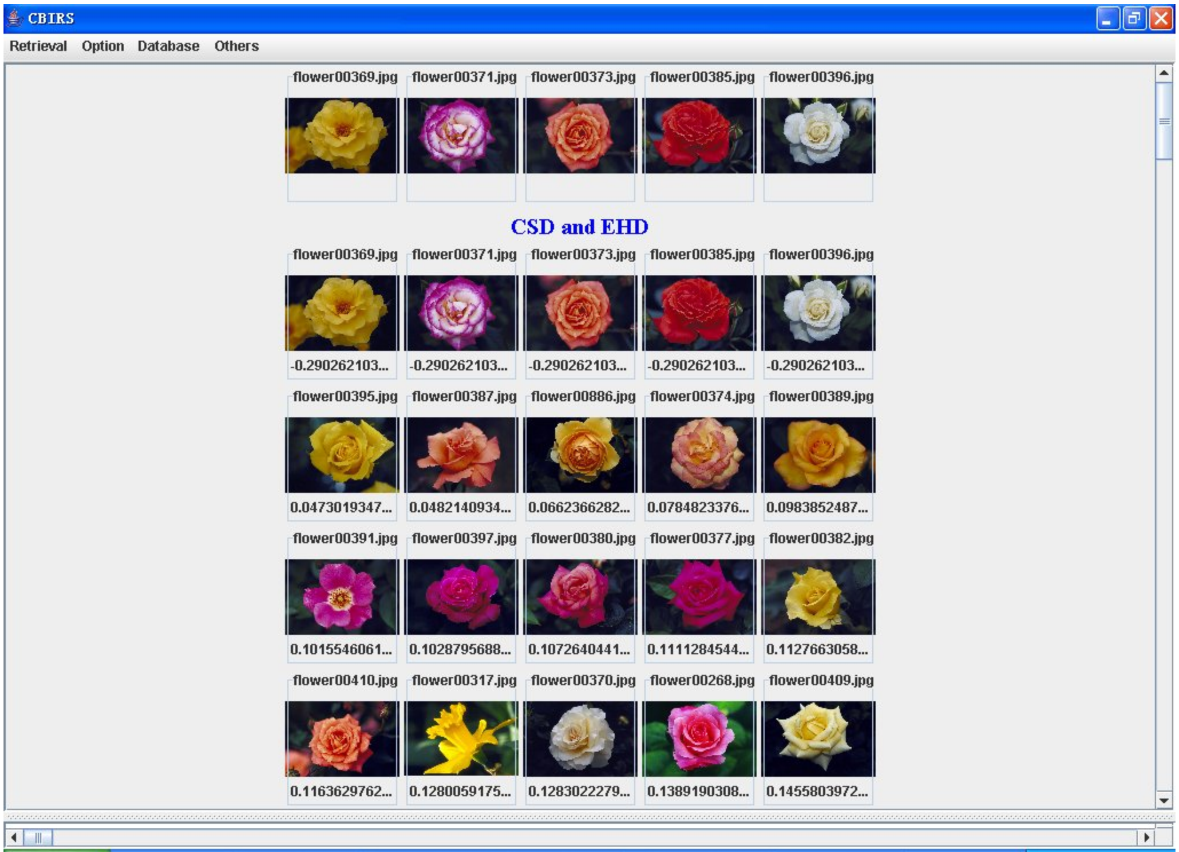


Figure 4.5: Five flowers

	CSD	EHD
Weights	0.361	0.632

Table 4.4: Five flower's combination weights

The five query images of differently coloured flowers shown Figure 4.5, are submitted to the system. It is expected that the combination weight of texture feature should be higher than that assigned to colour feature. By employing the proposed weights modification schemes, the combination weights for this query are displayed in Table 4.4. The EHD's weight is 0.632, which is almost twice the value of CSD's. The retrieval result is also displayed in Figure 4.5. It can be seen that flowers with different color are found as the result. In the following steps, the query set is varied a little bit so as to test whether the proposed system can capture the different kinds of changes.

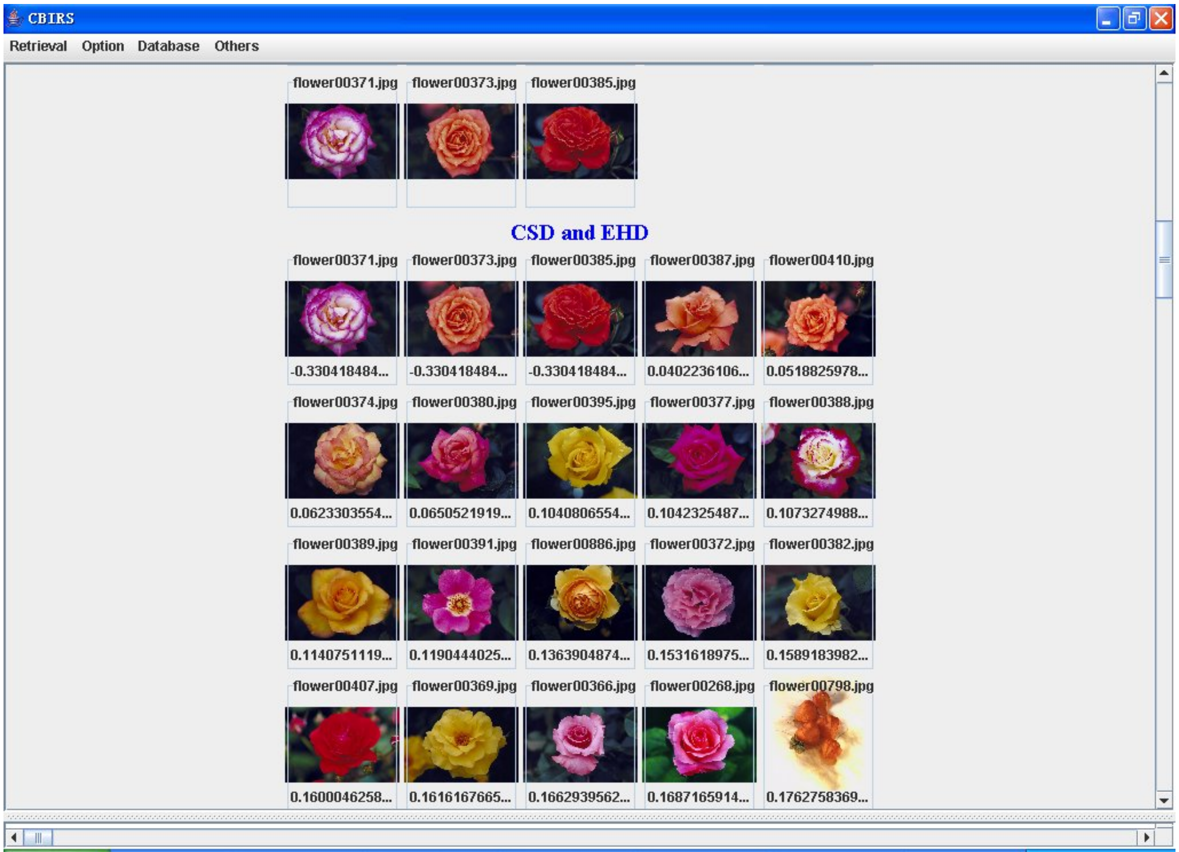


Figure 4.6: Three similar color flowers

	CSD	EHD
Weights	0.473	0.527

Table 4.5: Three similar color flowers' combination weights

In the first case, two flower images with very different colors, yellow (image $flower00369.jpg$) and white (image $flower00396$), are removed from the query set. The remaining three flowers share relatively similar color features than the original query images as shown in Figure 4.6. Therefore, it is expected that the combination weight for colour feature should be increased relative to the original weight. In table 4.5, the combination weights for the new query is displayed. It can be seen that the CSD's weight is increased.

	CSD	EHD
Weights	0.210	0.790

Table 4.6: Three different color flowers' combination weights

In the second case, by removing image $flower00373.jpg$ and $flower00385.jpg$, three

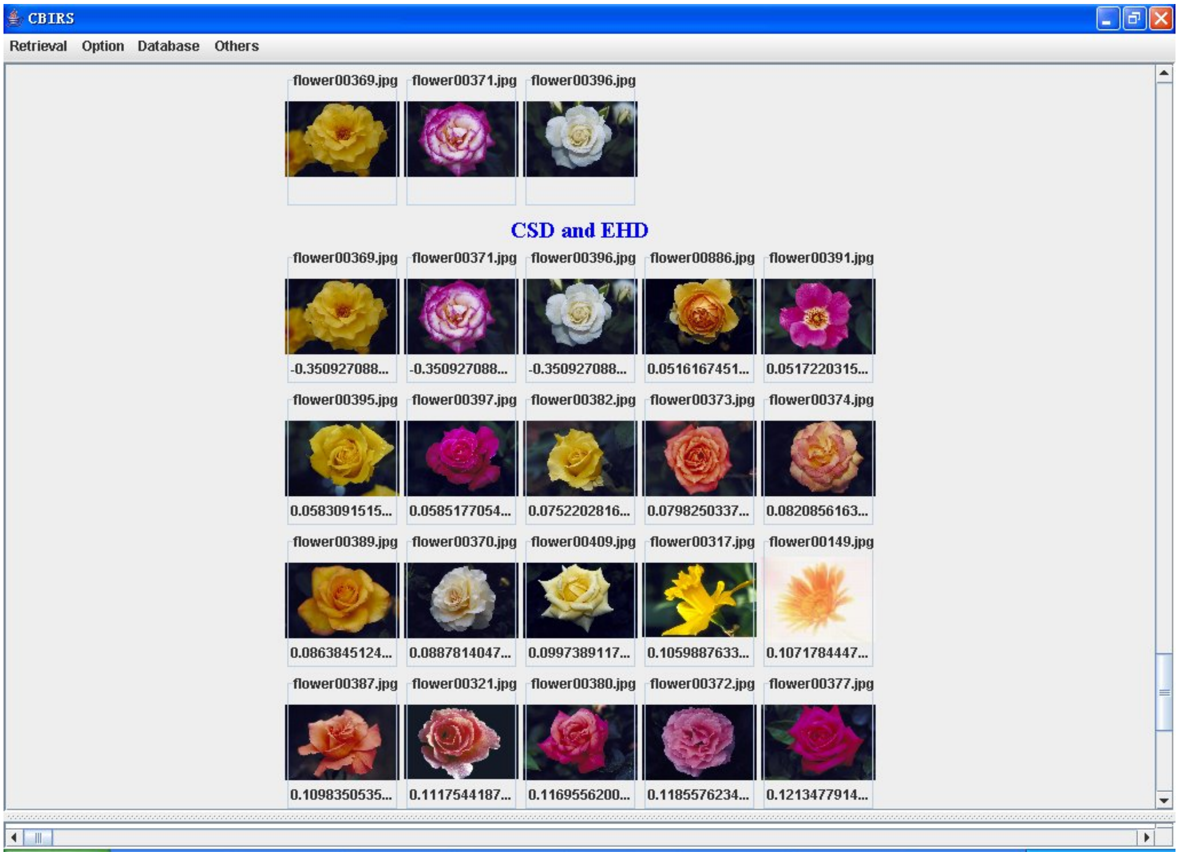


Figure 4.7: Three different color flowers

flowers with very different color but similar texture are submitted as query (Figure 4.7. Therefore, it is expected that the weight assigned to color feature should be decrease relative to those of the original query set. By analysing the query images' feature characteristics, the proposed system decreases CSD's combination weight to 0.21, which is nearly 60 percent of the original weight.

The proposed system can also be employed in a scenario where it is required to retrieve all key frames from similar shots in a video sequence. It is assumed that there are no editing effects that will merge, fade or dissolve one frame into another. This could be case in a surveillance video footage acquired with a pan-tilt-zoom camera. In Figure 4.8, three images from the same video shot are submitted as the query. It can be seen that the proposed system can retrieve frames of similar shot from the image database. In Figure 4.9, three images in the same scene but different shots are submitted as the query. It can be seen that the proposed system can also retrieve different shots in the same scene from the image database.

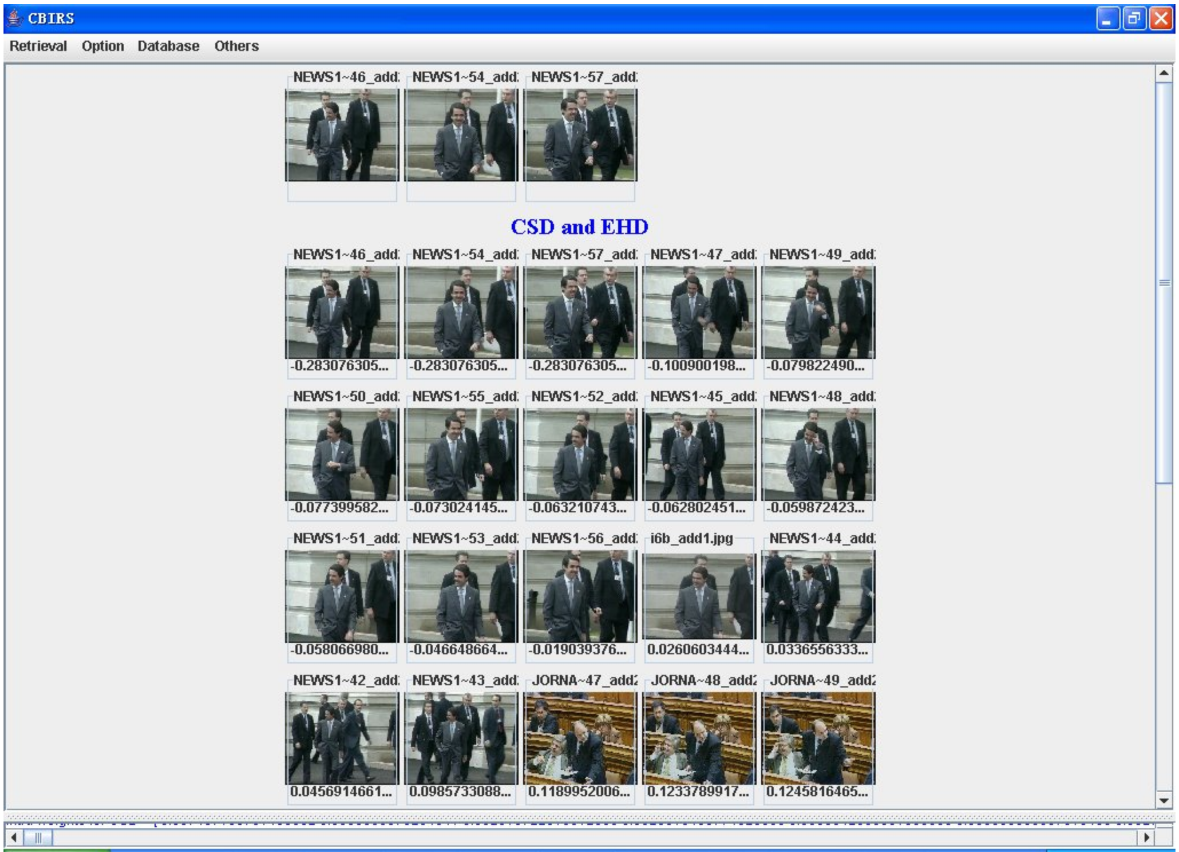


Figure 4.8: Sample E: walking people

In conclusion, the CBIRS based on the proposed weight estimation and assignment model can modify the weights dynamically according to the significant features of the query images and result in desirable results.

4.3 Chapter Summary and Conclusion

In this Chapter, by employing the weight modification approach introduced in Chapter 3, a multi-image query CBIR system is developed. The experimental results demonstrate how the weight modification dynamically adjusts weights effectively.

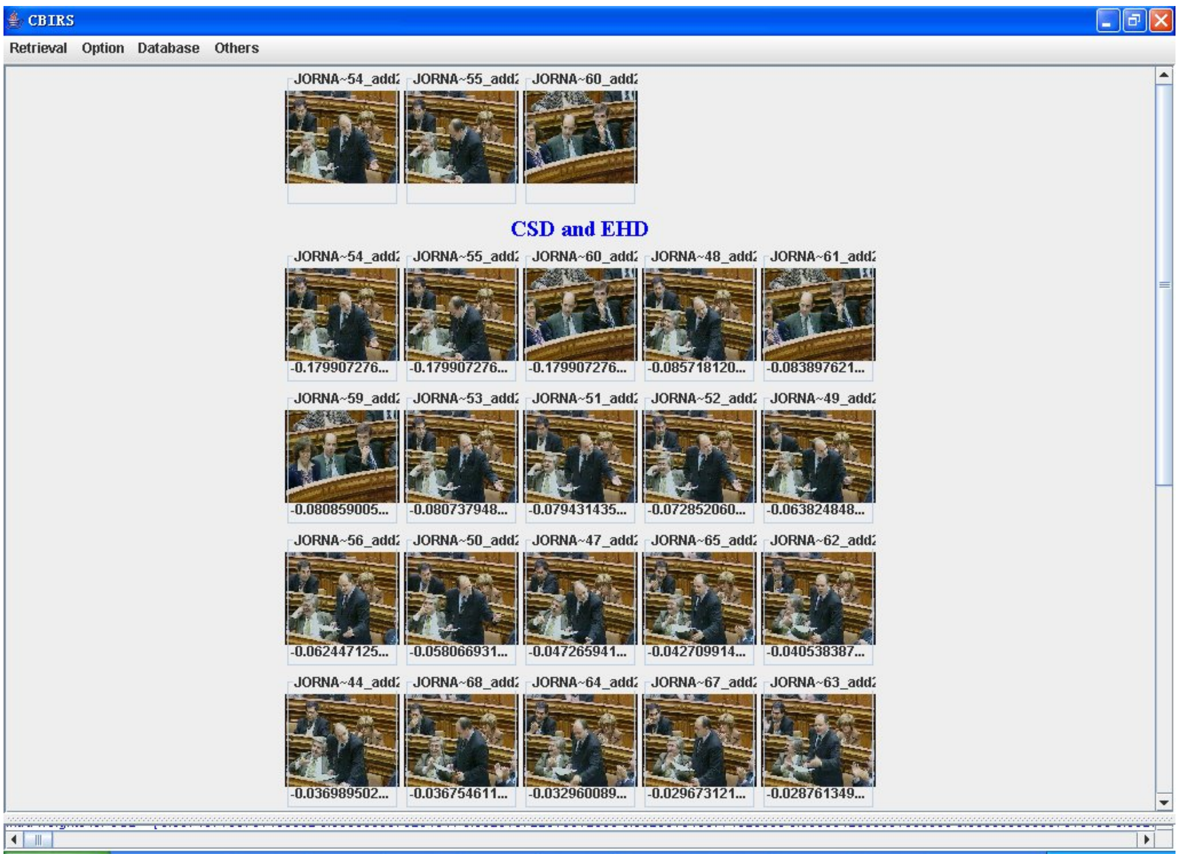


Figure 4.9: Sample F: baldheaded man representing

Chapter 5

Summary, Conclusion and Further Work

5.1 Introduction

In this chapter summary and conclusion of the thesis are presented, and some suggestions for new directions and improvements are given.

The thesis has considered content-based image retrieval from image databases using a multi-image query approach. The main objective of the multi-image query approach is to improve the ability of users to describe target images and by doing so to improve the quality of the query and retrieved images. In essence multi-image query offers the possibility to capture the query concept underlying the image set the user selected as examples.

In the retrieval process, the underlying query concept is captured by the proposed scheme and both the intra and inter-level weights are dynamically modified according to the significance of the selected features. For the intra-level search, by employing both the standard deviation and mean of each component of the features describing the query set, significant components are detected and assigned a high weight in the feature distance calculation. Experimental results show that the proposed scheme can correctly detect significant components of the feature, indirectly capture the underlying query concept and improve retrieval performance. For the inter-level search, by calculating the distance between pairs of images within the query image set, the significance of each feature in the chosen query image set is estimated. For features that have the distances distributed close to each other, large combination weights are assigned, whilst others are allocated low combination weights. By employing both intra and inter-level weight modifications, the overall underlying query concept is captured from the low-level feature vectors. The scheme has helped narrow the so-called "Semantic Gap" and thus achieve the aim and objectives of this thesis.

5.2 Thesis Summary and Conclusion

In this thesis, efforts has been focused on establishing a multi-image query content-based image retrieval system. The achievements can be divided into four main sections:

- Database Establishment

  With the aim of collecting many image samples, an image database for image retrieval was developed during the period of this research. All the images in the database are "real-world" images collected from several sources, such as the Internet, companies and standard image databases. According to the perceived meaning of each image, the database is further sub-divided into 22 categories, such as flower, car, tree, waterfall, sunset, animal, landscape, etc.

- Feature Selection

  The feature employed in any content-based retrieval system is critical in achieving a respectable performance. During this study, the Colour Layout Descriptor (LCD), Colour Structure Descriptor (CSD), Scalable Colour Descriptor (SCD), HSV Histogram Descriptor (HHD), Edge Histogram Descriptor (EHD) and Homogeneous Texture Descriptor (HTD) are extracted and compared. It was found that the combination of the Colour Structure Descriptor and Edge Histogram Descriptor can generate a better retrieval performance than other kinds of combinations.

- Performance Improvement

  The methods outlined below are developed and evaluated in terms of increasing the performance of the content-based image retrieval system .

Intra-level weight modification: The multi-image query approach is introduced in this thesis to better capture underlying query concept by modifying the intra-level feature weights. The first and second order moments of the columns of a feature matrix are employed in a significance computation for the components in each feature vector and weights are appropriately assigned to encode the query concept at the individual feature level.

Inter-level weights modification: Since the inter-level weights indicate the importance between selected features in the content-based image retrieval system, the assignment of the weights should therefore in accord with the underlying query concept from the inter feature perspective. In the multi-image query approach, the users' intended query concept can be reflected in the selection of query images, so, by analysing the relationships between the features of the query images, the significance of the features can be revealed. In this thesis, the visual distances between pairs of query images for all the selected features is used to estimate the significance of the features. The features are assigned weight in proportion to their significance.

- Image Search Engine

  During the period of this research, an image search engine was developed to demonstrate the use of the proposed weight modification model. The proposed models are applied to an off-line content-based image retrieval system. By employing both colour and texture descriptors this CBIRS can search images in the database according to their low-level features. The CBIRS supports both single and multi-image query.

5.3 Further Work

Some important issues related to the CBIR have been addressed in this thesis. However, there are still a number of possible improvements and new directions that require further investigations.

Possible improvements and further studies on the proposed methods include:

- The multi-image query approach can be combined with classification techniques, such as Bayesian classification, to improve retrieval performance. The database images can be divided in advance into several classes according to their visual characteristics. For each of the classes, a histogram approximating some underlying density can be generated. During the retrieval process, another histogram approximating the density of the query images is generated. By comparing these probability densities, one or more classes in the image database are first selected as the potential classes. After this class-level search, most of the irrelevant classes are excluded from consideration. Furthermore, an image-level retrieval can be generated based only on those potential classes. By dividing the normal retrieval process into classes and image-level searches, 1) the speed of performance can be improved since only one subset of the database is searched, instead of the whole database; and 2) the precision of performance will also be improved, since most irrelevant images are excluded during the class-level search. The key challenge of the proposed improvement is how to classify the database images according to their feature characteristics and usage of the probability density functions.

- The Region-Based image retrieval techniques can be combined with the proposed approach to search the Regions-of-Interest in the query image.

- Text-based annotation can be combined with the content-based approach to capture users' expectation.

Bibliography

[1] ISO/IEC 15938-3:2001. Multimedia content description interface - part 3: Visual. volume 1, 2001.

[2] ISO/IEC 15938-8:2002. Multimedia content description interface - part 8: Extraction and use of mpeg-7 descriptions. volume 1, 2002.

[3] M. Abdel-Mottaleb and S. Krishnamachari. Multimedia descriptions based on MPEG-7: Extraction and application. *IEEE Transactions on Multimedia*, 6(3):459 – 468, June 2004.

[4] A. Alexandrov, W. Y. Ma, A. E. Abbadi, and B. S. Manjunath. Adaptive filtering and indexing for image databases. *Signal Processing, Sensor Fusion, and Target Recognition*, 2420:12–23, 1995.

[5] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Association for Computing Machinery (ACM)*, 45(6):891–923, Nov. 1998.

[6] V. Atitsos, M. J. Swain, and C. Frankel. Distinguishing photographs and graphics on the world wide web. In *Workshop on Content-Based Access of Image and Video Libraries (CBAIVL)*, page 10, June 1997.

[7] J. R. Bach, C. Fuller, A. Gupta, A. Hampapur, B. Horowitz, R. Humphrey, R. Jain, and C. Shu. The Virage image search engine: An open framework for

image management. In *Proceedings of 4th Conference on Storage and Retrieval for Still Image and Video Databases*, pages 76–87, 1996.

[8] J. A. Benediktsson, J. R. Sveinsson, and O.K. Ersoy. Optimized combination of neural networks. In *IEEE International Symposium on Circuits and Systems*, pages 535–538, May 1996.

[9] T. E. Bjoerge and E. Y. Chang. Why one example is not enough for an image query. In *IEEE International Conference on Multimedia Expo (ICME)*, pages 253–256, 2004.

[10] M. Bober, W. Price, and J. Atkinson. The contour shape descriptor for MPEG-7 and its applications. In *Digest of Technical Papers. International Conference on Consumer Electronics (ICCE)*, pages 286–287, June 2000.

[11] R. Brunelli and O. Mich. Image retrieval by examples. *IEEE Transactions on Multimedia*, 2(3):164–171, 2000.

[12] A. Chalechale. *Content-Based Retrieval from Image Database Using Sketched Queries*. PhD thesis, University of Wollongong, March 2005.

[13] S. F. Chang. Compressed-domain content-based image and video retrieval. In *Processing Symposium on Multimedia Communications and Video Coding*, pages 915–920, 1995.

[14] S. F. Chang. Compressed-domain techniques for image/video indexing and manipulation. In *The IEEE International Conference on Image Processing (ICIP)*, pages 314–317, 1995.

[15] S. F. Chang, T. S., and A. Puri. Overview of the MPEG-7 standard. *IEEE Trans. Circuits Syst. Video Techn*, 11(6):688–695, 2001.

[16] S. F. Chang, J. R. Smith, H. J. Meng, H. W., and Di Zhong. Finding images/video in large archives: Columbia's content-based visual query project. Technical report, D-Lib Magazine, Feb. 1997.

[17] Z. Chen, L. Wenyin, C. Hu, M. Li, and H. J. Zhang. ifind: A web image search engine. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 450, 2001.

[18] B. Cheng. *Approaches to Image Retrieval Based on Compressed Data for Multimedia Database Systems*. PhD thesis, University of New York at Buffalo, 1996.

[19] I. J. Cox, M. L. Miller, T. P. Minka, and T. V. Papathomas. The Bayesian image retrieval system, PicHunter: Theory, implementation, and psychophysical experiments. *IEEE Trans. Image Processing*, 9(1):20–37, Jan. 2000.

[20] I. J. Cox, M. L. Miller, S. M. Omohundro, and P. N. Yianilos. Pichunter: Bayesian relevance feedback for image retrieval. In *International Conference on Pattern Recognition*, volume 3, pages 361–369, 1996.

[21] I. J. Cox, M. L. Miller, S. M. Omohundro, and P. N. Yianilos. Target testing and the PicHunter Bayesian multimedia retrieval system. In *Advances in Digital Libraries (ADL)*, pages 66–75, 1996.

[22] A. D. Doulamis and N. D. Doulamis. Generalized nonlinear relevance feedback for interactive content-based retrieval and organization. *IEEE Trans. Circuits Syst. Video Techn*, 14(5):656–671, 2004.

[23] J. Dowe. Content-based retrieval in multimedia imaging. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 164–167, 1993.

[24] W. Equitz and W. Niblack. Retrieving images from a database using texture - algorithm from the QBIC system. Technical Report RJ 9805, Computer Science, IBM Research Report, May 1994.

[25] C. Faloutsos, M. Flickner, W. Niblack, D. Petkovic, W. Equitz, and R. Barber. Efficient and effective querying by image content. Technical report, IBM Research Report, 1993.

[26] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, Sep. 1995.

[27] C. Frankel, M. J Swain, and V. Athitsos. WebSeer: An image search engine for the world wide web. Technical Report TR-96-14, Department of Computer Science, University of Chicago, July 1996.

[28] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Anal. Mach. Intell*, 13(9):891–906, 1991.

[29] H. Frigui and S. Salem. Fuzzy clustering and subset feature weighting. In *The 12th IEEE International Conference on Fuzzy Systems*, volume 2, pages 857–862, 2003.

[30] T. Gevers and A. W. M. Smeulders. PicToSeek: A content-based image search system for the world wide web. In *International Conference on Visual Information Systems (ICVIS)*, pages 93–100, 1997.

[31] T. Gevers and A. W. M. Smeulders. The PicToSeek WWW image search system. In *International Conference on Multimedia Communications Systems (ICMCS)*, volume 1, pages 264–269, 1999.

[32] T. Gevers and A. W. M. Smeulders. PicToSeek: Combining color and shape invariant features for image retrieval. *IEEE Trans. Image Processing*, 9(1):102–119, Jan. 2000.

[33] I. Gondra, D. R. Heisterkamp, and J. Peng. Improving image retrieval performance by inter-query learning with one-class support vector machines. *Neural Computing and Applications*, 13(2):130–139, 2004.

[34] Y. Gong, H. J. Zhang, H. C. Chuan, and M. Sakauchi. An image database system with contents capturing and fast image indexing abilities. In *International*

Conference on Multimedia Communications Systems (ICMCS), pages 121–130, 1994.

[35] Abby A. Goodrum. Image information retrieval: An overview of current research. In *Informing Science*, volume 3, pages 63–66, Jan. 2000.

[36] M. M. Gorkani and R. W. Picard. Texture orientation for sorting photos at a glance. In *International Conference on Pattern Recognition*, pages 459–464, 1994.

[37] R. M. Gray. Vector Quantization. *IEEE ASSP Magazine*, Apr. 1984.

[38] Eugene J. Guglielmo and Neil C. Rowe. Natural-language retrieval of images based on descriptive captions. *ACM Transactions on Information Systems*, 14(3):237–267, July 1996.

[39] Amarnath Gupta and Ramesh Jain. Visual information retrieval. *Communications of the ACM*, 40(5):71–79, May 1997.

[40] Keister L. H. User types and queries: Impact on image access systems. *In Challenges in Indexing Electronic Text and Images*, 1:7–22, 1994.

[41] M. Li H. Zhang, Z. Chen and Z. Su. Relevance feedback and learning in content-based image search. In *World Wide Web: Internet and Web Information Systems*, volume 6, pages 131–155, 2003.

[42] A. Hampapur, A. Gupta, B. Horowitz, and C. F. Shu. Virage video engine. *Signal Processing, Sensor Fusion, and Target Recognition*, 3022:188–198, 1997.

[43] C. B. Huang, S. S. Yu, J. L. Zhou, and H. W. Lu. Image retrieval using both color and local spatial feature histograms. In *2004 International Conference on Communications, Circuits and Systems*, pages 927 – 931, June 2004.

[44] J. Huang, R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih. Image indexing using color correlograms. In *Computer Vision and Pattern Recognition (CVPR)*, pages 762–768, 1997.

[45] Q. Huang, A. Puri, and Z. Liu. Multimedia search and retrieval: New concepts, system implementation, and application. *IEEE Trans. Circuits Syst. Video Techn*, 10(5):679–692, 2000.

[46] T. S. Huang, S. Mehrotra, and K. Ramchandran. Multimedia analysis and retrieval system (MARS) project. In *33rd Annual Clinic on Library Application of Data Processing - Digital Image Access and Retrieval*, pages 1–4, 1996.

[47] T. S. Huang and X. Zhou. Image retrieval with relevance feedback: from heuristic weight adjustment to optimal learning methods. In *The IEEE International Conference on Image Processing (ICIP)*, volume 3, pages 2–5, 2001.

[48] X. Y. Huang, Y. J. Zhong, and D. Hu. Image retrieval based on weighted texture features using DCT coefficients of JPEG images. In *Proceedings of the 2003 Joint Conference of the Fourth International Conference on Information, Communications and Signal Processing*, pages 1571–1575, Dec. 2003.

[49] Q. Iqbal and J. K. Aggarwal. Combining structure, color and texture for image retrieval: A performance evaluation. In *International Conference on Pattern Recognition (ICPR)*, pages 438–443, 2002.

[50] Q. Iqbal and J. K. Aggarwal. Feature integration multi-image queries and relevance feedback in image retrieval. In *6th International Conference on Visual Information Systems*, pages 467–474, Sep. 2003.

[51] S. Jeong, K. Kim, B. Chun, J. Lee, and Y. J Bae. An effective method for combining multiple features of image retrieval. In *Proceedings of the IEEE Region 10 Conference TENCON*, volume 2, pages 982–985, Sept. 1999.

[52] X. Jin and J. C. French. Improving image retrieval effectiveness via multiple queries. In Shu-Ching Chen and Mei-Ling Shyu, editors, *Proceedings of the First ACM International Workshop on Multimedia Databases*, pages 86–93, Nov. 2003.

[53] E. Kasutani and A. Yamada. The MPEG-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval. In *International Conference on Image Processing (ICIP)*, pages 674–677, Oct. 2001.

[54] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data. An Introduction to Cluster Analysis*. Wiley, New York, 1990.

[55] P. Koikkalainen. Progress with the tree-structured self-organizing map. In *Proceedings of the Eleventh European Conference on Artificial Intelligence*, pages 211–215, Aug. 1994.

[56] P. Koikkalainen and E. Oja. Self-organizing hierarchical feature maps. In *IEEE International Joint Conference on Neural Networks (IJCNN)*, volume 2, pages 279–284, 1990.

[57] M. Koskela, J. Laaksonen, and E. Oja. MPEG-7 descriptors in content-based image retrieval with PicSOM system. In *Application Areas of Visual Information Systems (VISUAL)*, pages 247–258, 2002.

[58] R. Krishnapuram, S. Medasani, S. H. Jung, Y. Choi, and R. Balasubramaniam. Content-based image retrieval based on a fuzzy approach. *IEEE Trans. Knowl. Data Eng*, 16(10):1185–1199, 2004.

[59] R. Krishnapuram, S. Medasani, S.H. Jung, Y. S. Choi, and R. Balasubramaniam. Content-based image retrieval based on a fuzzy approach. *IEEE Transactions on Knowledge and Data Engineering*, 18(10):1185 – 1199, Oct. 2004.

[60] A. Kushki, P. Androutsos, K. N. Plataniotis, and A. N. Venetsanopoulos. Retrieval of images from artistic repositories using a decision fusion framework. *IEEE Transactions on Image Processing*, 13(3):277–292, 2004.

[61] A. Kushki, A. N. Venetsanopoulos, K. N. Plataniotis, and P. Androutsos. Fuzzy aggregation of palette colors for hybrid querying of fine art image databases. In *14th International Conference on Digital Signal Processing (DSP)*, March 2002.

[62] J. Laaksonen, M. Koskela, S. L., and E. Oja. PicSOM—content-based image retrieval with self-organizing maps. *Pattern Recognition Letters*, 21(13–14):1199–1207, Dec. 2000.

[63] J. Laaksonen, M. Koskela, S. Laakso, and E. Oja. Self-organising maps as a relevance feedback technique in content-based image retrieval. *Pattern Analysis and Applications*, 4(2–3):140–152, 2001.

[64] J. Laaksonen, M. Koskela, and E. Oja. PicSOM—A framework for content-based image database retrieval using self-organizing maps. In *Processing of 11th Scandinavian Conference on Image Analysis (SCIA)*, pages 151–156, June 1999.

[65] J. Laaksonen, M. Koskela, and E. Oja. PicSOM: Self-organizing maps for content-based image retrieval. In *IEEE International Conference on Neural Networks (IJCNN)*, volume 4, pages 2470–2473, July 1999.

[66] J. Laaksonen, M. Koskela, and E. Oja. PicSOM-self-organizing image retrieval with MPEG-7 content descriptors. volume 13, pages 841–853, July 2002.

[67] M. K. Lahmari, A. E. Kamel, and P. Borne. Multimodel control using fuzzy fusion. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, volume 1, pages 38–43, Oct. 1999.

[68] D. Lee, R. Barber, W. Niblack, M. D. Flickner, J. Hafner, and D. Petkovic. Indexing for complex queries on a query-by-content image database. In *International Conference on Pattern Recognition (ICPR)*, volume A, 1994.

[69] H. K. Lee and S. I. Yoo. A neural network-based image retrieval using nonlinear combination of heterogeneous features. In *Proceedings of the 2000 Congress on Evolutionary Computation (CEC)*, pages 667–674, July 2000.

[70] B. Li and S. Yuan. A novel relevance feedback method in content-based image retrieval. In *International Conference on Information Techonlogy (ITCC)*, pages 120–123, 2004.

[71] Y. Linde, A. Buzo, and R. M. Gray. An algorithm for vector quantizer design. *IEEE Transactions on Communications*, 28(1):84–95, Jan 1980.

[72] G. Lu and A. Sajjanhar. Region-based shape representation and similarity measure suitable for content-based image retrieval. *Multimedia Systems*, 7(2):165–174, 1999.

[73] Y. Lu, C. Hu, X. Zhu, H. Zhang, and Q. Yang. A unified framework for semantics and feature based relevance feedback in image retrieval systems. In *Proceedings of the 8th International ACM Conference on Multimedia*, pages 31–38, Oct.–Nov. 2000.

[74] M. Ghanbari M. Grgic and S. Grgic. Texture-based image retrieval in MPEG-7 multimedia system. In *International Conference on EUROCON*, pages 365–368, July 2001.

[75] W. Y. Ma and B. S. Manjunath. Image indexing using a texture dictionary. *Signal Processing, Sensor Fusion, and Target Recognition*, 260:288–296, Oct. 1985.

[76] W. Y. Ma and B. S. Manjunath. A comparison of wavelet transform features for texture image annotation. In *The IEEE International Conference on Image Processing (ICIP)*, pages 2256–2259, 1995.

[77] W. Y. Ma and B. S. Manjunath. Texture features and learning similarity. In *Computer Vision and Pattern Recognition (CVPR)*, pages 425–430, 1996.

[78] W. Y. Ma and B. S. Manjunath. Edge flow: A framework of boundary detection and image segmentation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 744–749, 1997.

[79] W. Y. Ma and B. S. Manjunath. NeTra: A toolbox for navigating large image databases. In *The IEEE International Conference on Image Processing (ICIP)*, volume 1, pages 568–571, 1997.

[80] W. Y. Ma and B. S. Manjunath. A texture thesaurus for browsing large aerial photographs. *Journal of the American Society for Information Science (JASIS)*, 49(7):633–648, 1998.

[81] X. Ma and D. Wang. Learning similarity for image retrieval with locally spatial information feedback. In *Proceedings of International Conference on Intelligent Sensing and Information Processing*, pages 133–138, 2004.

[82] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Trans. Pattern Anal. Mach. Intell*, 18(8):837–842, 1996.

[83] B. S. Manjunath, J. R. Ohm, V. V. Vasudevan, and A. Yamada. Color and texture descriptors. *IEEE Trans. Circuits Syst. Video Techn*, 11(6):703–715, 2001.

[84] B. S. Manjunath, P. Salembier, and T. Sikora. *Introduction to MPEG-7: Multimedia Content Description Standard*. Wiley, New York, 2001.

[85] T. P. Minka and R. W. Picard. Interactive learning with a society of models. In *Computer Vision and Pattern Recognition*, pages 447–452, June 1996.

[86] M. Nakazato and T. S. Huang. Extending image retrieval with group-oriented interface. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 201–204, Aug. 2002.

[87] W. Niblack, R. Barber, W. Equitz, M. D. Flickner, D. Glasman, D. Petkovic, and P. Yanker. The QBIC project: Querying image by content using color, texture, and shape. *Signal Processing, Sensor Fusion, and Target Recognition*, 1908:173–187, Feb. 1993.

[88] T. Chen O. Huseyin and H. R. Wu. Performance evaluation of multiple regions-of-interest query for accessing image databases. In *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing*, pages 300–303, May 2001.

[89] A. Ono, M. Amano, M. Hakaridani, T. Satou, and M. Sakauchi. A flexible content-based image retrieval system with combined scene description keyword. In *IEEE International Conference on Multimedia Computing and Systems*, June 1996.

[90] B. C. Ooi, K. L. Tan, T. S. Chua, and W. Hsu. Fast image retrieval using color-spatial information. *Very Large Data Bases (VLDB)*, 7(2):115–128, 1998.

[91] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and T. Huang. Supporting similarity queries in MARS. In *Proceedings of The Fifth ACM International Multimedia Conference*, pages 403–414, Nov 1998.

[92] E. Paquet and M. Rioux. The MPEG-7 standard and the content-based management of three-dimensional data: A case study. In *International Conference on Multimedia Communications Systems (ICMCS)*, volume 1, pages 375–380, 1999.

[93] D. K. Park, Y. S. Jeon, and C. S. Won. Efficient use of local edge histogram descriptor. In *Proceedings of the ACM Multimedia 2000 Workshop*, pages 51–54, Nov. 2000.

[94] G. Pass, R. Zabih, and J. Miller. Comparing images using color coherence vectors. In *Proceedings of the Fourth ACM Multimedia Conference*, pages 65–74, Nov. 1996.

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

[96] D. Petkovic, W. Niblack, M. Flickner, D. Steele, D. Lee, J. Yin, J. Hafner, F. Tung, H. Treat, R. Dow, M. Gee, M. Vo, P. Vo, B. J. Holt, J. Hethorn, K. Weiss, P. Elliott, and C. Bird. Recent applications of IBM's query by image content (QBIC). In *ACM Symposium on Applied Computing (SAC)*, pages 2–6, 1996.

[97] R. W. Picard. Computer learning of subjectivity. *ACM Computing Surveys*, 27(4):621–623, Dec. 1995.

[98] R. W. Picard. A society of models for video and image libraries. *IBM Systems Journal*, 35(3/4):292–312, 1996.

[99] R. W. Picard and T. P. Minka. Vision texture for annotation. Technical Report 302, MIT Media Lab Vismod, 1994.

[100] R. W. Picard, T. P. Minka, and Martin Szummer. Modeling user subjectivity in image libraries. In *Proc. of Int. Conference on Image Processing*, volume 2, pages 777–780, Sep. 1996.

[101] F. Porikli. Automatic threshold determination of centroid-linkage region growing by MPEG-7 dominant color descriptors. In *International Conference on Image Processing (ICIP)*, pages 793–796, 2002.

[102] B. G. Prasad, K. K. Biswas, and S. K. Gupta. Region-based image retrieval using integrated color, shape, and location index. *Computer Vision and Image Understanding*, 94(1-3):193–233, Apr. 2004.

[103] B. G. Prasad, S. K. Gupta, and K. K. Biswas. Color and shape index for region-based image retrieval. In *International Workshop on Visual Form (IWVF)*, pages 716–728, 2001.

[104] A. Rao, R. K. Srihari, and Z. Zhang. Spatial color histograms for content-based image retrieval. In *International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 183–186, 1999.

[105] Y. Rui, T. Huang, and S. Chang. Image retrieval: Current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(4):39–62, Apr. 1999.

[106] Y. Rui and T. S. Huang. A novel relevance feedback technique in image retrieval. In *ACM Multimedia*, pages 67–70, 1999.

[107] Y. Rui and T. S. Huang. Optimizing learning in image retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 236–245. IEEE, June 2000.

[108] Y. Rui, T. S. Huang, and S. F. Chang. Image retrieval: Past, present, and future. In *International Symposium on Multimedia Information Processing*, volume 10, pages 1–23, 1997.

[109] Y. Rui, T. S. Huang, and S. Mehrotra. Content-based image retrieval with relevance feedback in MARS. In *The IEEE International Conference on Image Processing (ICIP)*, volume 2, pages 815–818, 1997.

[110] Y. Rui, T. S. Huang, Sharad Mehrotra, and Michael Ortega. A relevance feedback architecture for content-based multimedia information retrieval systems. In *Workshop on Content-Based Access of Image and Video Libraries (CBAIVL)*, pages 82–89, June 1997.

[111] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Automatic matching tool selection using relevance feedback in MARS. In *International Conference on Visual Information Systems*, pages 109–116, 1997.

[112] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Trans. Circuits and Systems for Video Technology*, 8(5):644–655, Sep. 1998.

[113] Huang Y. S. and Suen C. Y. Combination of multiple classifiers with measurement values. In *Proceedings of the Second International Conference on Document Analysis and Recognition*, pages 598 – 601, Oct. 1993.

[114] G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[115] B. Scassellati, S. Alexopoulos, and M. Flickner. Retrieving images by 2D shape: A comparison of computation methods with human perceptual judgments. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 2–14, 1994.

[116] Stan Sclaroff, Leonid Taycher, and Marco La Cascia. ImageRover: A content-based image browser for the world wide web. In *Workshop on Content-Based Access of Image and Video Libraries (CBAIVL)*, page 2, June 1997.

[117] H. Shao, J. W. Zhang, W. C. Cui, and H. Zhao. Automatic feature weight assignment based on genetic algorithm for image retrieval. In *2003 International Conference on Robotics, Intelligent Systems and Signal Processing*, pages 731–735, Oct. 2003.

[118] T. Sikora. The MPEG-7 visual standard for content description- an overview. *IEEE Trans. Circuits Syst. Video Techn*, 11(6):696–702, 2001.

[119] A. W. M. Smeulders, M. Worring, A. Gupta S. Santini, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell*, 22(12):1349–1380, Dec. 2000.

[120] J. R. Smith. Quantitative assessment of image retrieval effectiveness. *Journal of the American Society for Information Science and Technology (JASIST)*, 52(11):969–979, 2001.

[121] J. R. Smith and S. F. Chang. *Intelligent Multimedia Information Retrieval*, chapter Querying by Color Regions Using the VisualSEEK Content-Based Visual Query System. 1996.

[122] J. R. Smith and S. F. Chang. VisualSEEk: A fully automated content-based image query system. In *Proceedings of the Fourth ACM International Conference on Multimedia*, pages 87–98, 1996.

[123] J. R. Smith, Y. C. Chang, and C. S. Li. Multi-object multi-feature content-based search using MPEG-7. In *International Conference on Image Processing (ICIP)*, volume 3, pages 584–587, 2001.

[124] R. O. Stehling, M. A. Nascimento, and Alexandre X. F. On shapes of colors for content-based image retrieval. In *Proceedings of the 2000 ACM Workshops on Multimedia*, pages 171–174, 2000.

[125] S. Sural, G. Qian, and S. Pramanik. A histogram with perceptually smooth color transitionfor image retrieval. In *Joint Conference on Information Sciences (JCIS)*, pages 664–667, 2002.

[126] M. J. Swain. Interactive indexing into image databases. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 95–103, 1993.

[127] D. L. Swets and J. J. Weng. Efficient content-based image retrieval using automatic feature selection. pages 85–90, 1995.

[128] T. F. Syeda-Mahmood and D. Petkovic. On describing color and shape information in images. *Signal Processing: Image Communication (SP:IC)*, 16(1-2):15–31, Sep 2000.

[129] S. M. M. Tahaghoghi, J. A. Thom, and H. E. Williams. Are two pictures better than one? In *Australasian Database Conference (ADC)*, pages 138–144, 2001.

[130] H. Tamura, S. Mori, and T. Yamawaki. Textural features corresponding to visual perception. *IEEE Trans. Syst., Man, Cyb.*, 8(6):460–473, 1978.

[131] J. Tang and S. Acton. An image retrieval algorithm using multiple query images. In *Proceedings of Seventh International Symposium on Signal Processing and Its Applications*, pages 193–196, July 2003.

[132] L. Taycher, M. L. Cascia, and S. Sclaroff. Image digestion and relevance feedback in the imagerover WWW search engine. Technical report, CS Department, Boston University, Aug. 1997.

[133] Q. Tian, P. Hong, and T. S. Huang. Incorporate support vector machines to content-based image retrieval with relevance feedback. In *International Conference on Image Processing (ICIP)*, volume 3, pages 750–753, 2000.

[134] Q. Tian, P. Hong, and T. S. Huang. Update relevant image weights for content-based image retrieval using support vector machines. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 2, pages 1199–1202, 2000.

[135] Q. Tian, Y. Wu, and T. S. Huang. Combine user defined region-of-interest and spatial layout for image retrieval. In *International Conference on Image Processing (ICIP)*, volume 3, pages 476–479, Sep. 2000.

[136] M. Tico, T. Haverinen, and P. Kuosmanen. A method of color histogram creation for image retrieval. In *IEEE Nordic Signal Processing Symposium (NORSIG)*, pages 1–4, June 2000.

[137] G. Tziritas and S. Liapis. Colour and texture image retrieval using chromaticity histograms and wavelet. In *Application Areas of Visual Information Systems (VISUAL)*, volume 6, pages 676–686, Oct. 2004.

[138] A. Vadivel, A. K. Majumdar, and S. Sural. Characteristics of weighted feature vector in content-based image retrieval applications. In *Proceedings of International Conference on Intelligent Sensing and Information*, pages 127–132, 2004.

[139] B. Verma and S. Kulkarni. A fuzzy-neural approach for interpretation and fusion of colour and texture features for CBIR systems. *Appl. Soft Comput*, 5(1):119–130, 2004.

[140] C. Vertan and N. Boujemaa. Using fuzzy histograms and distances for color image retrieval. In *Challenge of Image Retrieval (CIR)*, pages 1–6, May 2000.

[141] K. Vu, K. A. Hua, and W. Tavanapong. Image retrieval based on regions of interest. *IEEE Trans. Knowl. Data Eng*, 15(4):1045–1049, 2003.

[142] H. Wang and S. F. Chang. Compressed-domain image search and applications. Technical report, Columbia University, 1995.

[143] S. Wang, L. Chia, and D. Rajan. Efficient image retrieval using MPEG-7 descriptors. In *International Conference on Image Processing (ICIP)*, pages 509–512, 2003.

[144] S. Wang, L. T. Chia, and D. Rajan. Image retrieval using dominant color descriptor. In *International Conference on Imaging Science, Systems, and Technology (CISST)*, pages 107–110, 2003.

[145] C. S. Won, D. K. Park, and S. J. Park. Efficient use of MPEG-7 edge histogram descriptor. *ETRL Journal*, 24(1):23–30, Feb. 2002.

[146] H. Wu, H. Lu, and S. Ma. WillHunter: Interactive image retrieval with multilevel relevance measurement. In *International Conference on Pattern Recognition (ICPR)*, volume 2, pages 1009–1012, 2004.

[147] J. K. Wu, A. D. Narasimhalu, B. M. Mehtre, C. P. Lam, and Y. J. Gao. CORE: A content-based retrieval engine for multimedia information systems. *Multimedia System*, 3(1):25–41, 1995.

[148] Y. Wu and A. Zhang. Category-based search using metadatabase in image retrieval. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 1, pages 197–200, Aug. 2002.

[149] R. Subramanya Y. Ishikawa and C. Faloutsos. Mindreader: Querying databases through multiple examples. In *Proceedings of 24rd International Conference on Very Large Data Bases (VLDB)*, pages 218–227, Aug. 1998.

[150] T. S. Huang Y. Rui and S. Mehrotra. Relevance feedback rechniques in interactive content-based image retrieval. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 25–36, 1998.

[151] R. R. Yager. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. *IEEE Transactions on Systems, Man, Cybernetics*, 18:183–190, 1988.

[152] X. Zeng and T. R. Martinez. Feature weighting using neural networks. In *IEEE International Joint Conference on Neural Networks*, pages 1327–1330, July 2004.

[153] D. Zhang and G. Lu. Generic Fourier descriptor for shape-based image retrieval. *IEEE International Conference on Multimedia and Expo (ICME)*, 1:425–428, Aug. 2002.

[154] D. Zhang and G. Lu. Evaluation of similarity measurement for image retrieval. In *Proceedings of the 2003 International Conference on Neural Networks and Signal Processing*, volume 2, pages 928–931, Dec. 2003.

[155] H. J. Zhang, L. Wenyin, and C. Hu. iFind: A system for semantics and feature based image retrieval over internet. In *Proceedings of the 8th International ACM Conference on Multimedia*, pages 477–478, Oct.–Nov. 2000.

[156] X. S. Zhou and T. S. Huang. Exploring the nature and variants of relevance feedback. In *Workshop on Content-Based Access of Image and Video Libraries (CAIVL)*, page 94, 2001.

[157] L. Zhu and A. Zhang. Supporting multi-example image queries in image databases. In *IEEE International Conference on Multimedia and Expo (ICME)*, volume 2, pages 697–700, 2000.