

# University of Wollongong - Research Online

## Thesis Collection

Title: A weighting scheme for content-based image retrieval

Author: Yuan Zhong

Year: 2007

Repository DOI:

### Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following: This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part of this work may be reproduced by any process, nor may any other exclusive right be exercised, without the permission of the author. Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material.

Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

**Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.**

Research Online is the open access repository for the University of Wollongong. For further information contact the UOW Library: [research-pubs@uow.edu.au](mailto:research-pubs@uow.edu.au)

*University of Wollongong Thesis Collections*

*University of Wollongong Thesis Collection*

---

*University of Wollongong*

*Year 2007*

---

# A weighting scheme for content-based image retrieval

Yuan Zhong  
University of Wollongong

Zhong, Yuan, A weighting scheme for content-based image retrieval, M.Comp.Sc.-Res thesis, School of Computer Science and Software Engineering, University of Wollongong, 2007.  
<http://ro.uow.edu.au/theses/667>

This paper is posted at Research Online.  
<http://ro.uow.edu.au/theses/667>

## **NOTE**

This online version of the thesis may have different page formatting and pagination from the paper copy held in the University of Wollongong Library.

## **UNIVERSITY OF WOLLONGONG**

### **COPYRIGHT WARNING**

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site. You are reminded of the following:

Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material. Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.



# **A WEIGHTING SCHEME FOR CONTENT-BASED IMAGE RETRIEVAL**

A Thesis Submitted in Partial Fulfilment of  
the Requirements for the Award of the Degree of

Master of Computer Science (Research)

from

UNIVERSITY OF WOLLONGONG

by

Yuan ZHONG

School of Computer Science and Software Engineering  
Faculty of Informatics

2007

---

© Copyright 2007

by

Yuan ZHONG

ALL RIGHTS RESERVED

## CERTIFICATION

I, Yuan ZHONG, declare that this thesis, submitted in partial fulfilment of the requirements for the award of Master of Computer Science (Research), in the School of Computer Science and Software Engineering, Faculty of Informatics, University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

(Signature Required)

Yuan ZHONG

30 March 2007

*Dedicated to*

*My Parents  
and  
My Supervisors*

# Table of Contents

List of Tables . . . . .	iii
List of Figures/Illustrations . . . . .	vi
ABSTRACT . . . . .	vii
Acknowledgements . . . . .	viii
<b>1 Introduction</b>	<b>1</b>
1.1 Introduction to Image Retrieval . . . . .	1
1.2 Benchmark datasets and performance indicators . . . . .	6
1.3 Issues in Image Retrieval . . . . .	8
1.4 Human Perception in CBIR . . . . .	10
1.5 Contributions of the Research . . . . .	11
1.6 Organization of the Thesis . . . . .	12
<b>2 Literature Review</b>	<b>13</b>
2.1 Introduction . . . . .	13
2.2 Human visual perception and visual features . . . . .	14
2.2.1 Theory of Human Perception . . . . .	14
2.2.2 Visual Features . . . . .	18
2.2.3 Perception Based CBIR Systems . . . . .	25
2.3 Query-by-example . . . . .	27
2.3.1 Region Based QBE . . . . .	31
2.3.2 Object Based QBE . . . . .	35
2.4 Relevance feedback . . . . .	39
2.4.1 Approaches in Relevance Feedback . . . . .	41
2.4.2 Combining Partial Results . . . . .	48
2.5 Query by multiple images (QBMI) . . . . .	51
2.5.1 Approaches in QBMI . . . . .	52
2.6 Formulation of the problem . . . . .	59
<b>3 Perceived Similarity and Visual Descriptions in Image Retrieval</b>	<b>61</b>
3.1 Introduction . . . . .	61
3.2 Psychological Experiments on Perceived Similarity of Images . . . . .	63
3.3 Perceived Similarity and Similarity Measured by MPEG-7 Descriptors . . . . .	66
3.3.1 Bushes . . . . .	68



3.3.2	Party . . . . .	71
3.3.3	Other categories . . . . .	71
3.4	Further Analysis of Experimental Results . . . . .	76
3.5	Effect of Weights in Combining Visual Descriptors . . . . .	82
3.6	Summary and Conclusions . . . . .	90
<b>4</b>	<b>A New Weighting Method for QBMI</b>	<b>91</b>
4.1	Interpretations of QBMI . . . . .	91
4.1.1	Compositional View . . . . .	92
4.1.2	Descriptive View . . . . .	93
4.1.3	Discussion . . . . .	93
4.2	Descriptive View Based QBMI . . . . .	94
4.3	Calculation of Inter-weights Using MI . . . . .	97
4.3.1	Mutual Information . . . . .	98
4.3.2	Calculation of inter-weights from MI . . . . .	99
4.4	Proposed Weighting Method . . . . .	100
4.4.1	Normalization of the distances . . . . .	101
4.5	Conclusion . . . . .	103
<b>5</b>	<b>Experimental Results</b>	<b>104</b>
5.1	Overview of the QBMI system . . . . .	104
5.2	Experimental Setup . . . . .	106
5.3	Experimental Results . . . . .	107
5.3.1	Multiple descriptors vs. Single descriptor . . . . .	107
5.3.2	Proposed weights vs. equal weights . . . . .	109
5.3.3	Proposed weights vs. heuristic weights . . . . .	111
5.3.4	Overall System Performance . . . . .	113
5.3.5	Semantic retrieval . . . . .	117
5.4	Conclusion . . . . .	121
<b>6</b>	<b>Conclusion</b>	<b>124</b>
6.1	Summary and Conclusion . . . . .	124
6.2	Future Work . . . . .	125
	<b>References</b>	<b>139</b>

# List of Tables

3.1	Mean values and standard deviations for normalized distances on CLD feature space . . . . .	80
3.2	Mean values and standard deviations for normalized distances on CSD feature space . . . . .	81
3.3	Mean values and standard deviations for normalized distances on EHD feature space . . . . .	81
3.4	Mean values and standard deviations for normalized distances on HTD feature space . . . . .	82
5.1	Feature weights for “Sphinx” . . . . .	111
5.2	Feature weights for “Yellow Butterfly” . . . . .	113
5.3	Feature weights for “Sphinx” . . . . .	113
5.4	Feature weights for “Flowers” and “Red Flowers” . . . . .	119

# List of Figures

1.1	Basic Procedure of Image Retrieval . . . . .	2
1.2	An example of CBIR system architecture . . . . .	4
2.1	Duck/Rabbit figure . . . . .	15
2.2	People will see two images with similar objects in it. Then they will find some differences when they look into more details [31] . . . . .	17
2.3	An example of the Polar Shape Representation [46] . . . . .	25
2.4	The Flow Chart of Genetic Algorithm [83] . . . . .	32
2.5	IRM is more robust to poor image segmentation than traditional methods [45] . . . . .	36
2.6	The Flow Chart of Relevance Feedback Based Image Retrieval [34] . . . . .	40
2.7	Semantically related images are usually different in visual features [37] . . . . .	54
2.8	Semantically related images are scattered in several visual clusters [37] . . . . .	54
2.9	Retrieval by query center of multiple queries may achieve different effects when the queries are located in one or more than one cluster [37] . . . . .	55
2.10	Two strategies for multi-query based IR . . . . .	57
2.11	Example of Multi-Query Based IR [77] . . . . .	59
3.1	User Interface of the Experiment System . . . . .	65
3.2	Experimental Results: “Bushes” . . . . .	66
3.3	Ground Truth Set: “Bush” . . . . .	67
3.4	Precision-Recall Curves - “Bush”. . . . .	69
3.5	Retrieval Results by CLD for “Bush” . . . . .	69
3.6	Retrieval Results by CSD for “Bush” . . . . .	70
3.7	Retrieval Results by EHD for “Bush” . . . . .	70
3.8	Retrieval Results by HTD for “Bush” . . . . .	71
3.9	Precision-Recall Curves - “Party”. . . . .	72
3.10	Retrieval Results by Human Subjects for “Party” . . . . .	72
3.11	Ground Truth Set - Parties . . . . .	73
3.12	Retrieval Results by CLD for “Party” . . . . .	74
3.13	Retrieval Results by CSD for “Party” . . . . .	74
3.14	Retrieval Results by EHD for “Party” . . . . .	75
3.15	Retrieval Results by HTD for “Party” . . . . .	75
3.16	Precision-Recall Curves - Car. . . . .	76

3.17	Precision-Recall Curves - Horse. . . . .	77
3.18	Precision-Recall Curves - Beach. . . . .	77
3.19	Precision-Recall Curves - Flower. . . . .	78
3.20	Precision-Recall Curves - Mountain. . . . .	78
3.21	Precision-Recall Curves - Opera House. . . . .	79
3.22	Precision-Recall Curves - Ship in Ocean. . . . .	79
3.23	Precision-Recall Curves - Sunset. . . . .	80
3.24	Combine the features by putting greater weights on more important features in category "Bush". . . . .	84
3.25	Retrieval results with combined descriptors for "Bush" . . . . .	85
3.26	Combine the features by putting equal weights to the descriptors in category "Bush". . . . .	86
3.27	Combine the features by putting greater weights on less important features in category "Bush". . . . .	87
3.28	Combine the features by putting equal weights to all descriptors in category "Party". . . . .	88
3.29	In category "Party", weights of CSD and EHD are set to 0.3 and weights of CLD and HTD are set to 0.2 . . . . .	89
3.30	In category "Party", weights of CSD and EHD are set to 0.2 and weights of CLD and HTD are set to 0.3 . . . . .	89
4.1	The retrieval procedure of a QBMI using MPEG-7 visual descriptors . . . . .	96
5.1	Four models of a QBMI system . . . . .	105
5.2	Query set and ground truth for "Sunset" . . . . .	108
5.3	Single descriptor vs multiple descriptors for "Sunset" . . . . .	109
5.4	Query set and ground truth for "Tower" . . . . .	110
5.5	Single descriptor vs multi descriptors for "Tower" . . . . .	110
5.6	Query set and ground truth for "Sphinx" . . . . .	111
5.7	Equal weights vs proposed weights for "Sphinx" . . . . .	112
5.8	Query set and ground truth for "Yellow Butterfly" . . . . .	113
5.9	Proposed weights vs heuristic weights in category "Yellow Butterfly" . . . . .	114
5.10	Proposed weights vs heuristic weights in category "Sphinx" . . . . .	114
5.11	Average precision and recall curve over twenty image categories . . . . .	115
5.12	Query set and ground truth for "Lilies" . . . . .	116
5.13	Precision and recall curve for "Lilies" . . . . .	116
5.14	Query set and ground truth for "Flowers in Different Colour" . . . . .	117
5.15	Precision and recall curve for "Flowers in Different Colour" . . . . .	118
5.16	Query set for "Red Flowers" . . . . .	118
5.17	Ground truth for "Red Flowers" . . . . .	119
5.18	Retrieval result of "Flowers in Different Colour" . . . . .	119
5.19	Retrieval result of "Red Flowers" . . . . .	120
5.20	Semantic retrieval: "Flowers" vs "Red Flowers" . . . . .	120
5.21	Query set for "Building in a desert" . . . . .	121

5.22	Ground truth for “Building in a desert” . . . . .	122
5.23	Semantic retrieval: “Desert Buildings” vs “Sphinx” . . . . .	123

# A WEIGHTING SCHEME FOR CONTENT-BASED IMAGE RETRIEVAL

Yuan ZHONG

A Thesis for Master of Computer Science (Research)

School of Computer Science and Software Engineering  
University of Wollongong

## ABSTRACT

In a query-by-example(QBE) image retrieval, the user is required to provide a single query image that most represents the features of the target images. On the other hand, “query by multiple images” paradigm assumes that the user is able to describe the target images more accurately by using multiple query images rather than one. Low level features, such as color, texture or edge information, are used to represent the images. These features are combined and expected to match the human perception properly. A set of psychological experiments are designed and conducted in this thesis with the aim of gaining insight into how a user perceives similar images. The retrieval results obtained by human subjects are compared with those obtained by using MPEG-7 visual descriptors. It is found that proper weight assignment in combining different features for retrieval can improve the retrieval performance. A novel weighting scheme for Query-by-Multiple-Images (QBMI) retrieval systems which aims to match human perception is proposed. The weights are derived by a new method of ascribing relative importance to feature descriptors for given a query set. Experimental results have shown that our weighing method is more effective than both equal weights and heuristic weighting method.

**KEYWORDS:** Content Based Image Retrieval, Query By Example, Multiple Image Query, Feature Weighting Scheme

# Acknowledgements

- I would like to thank, first and foremost, my supervisor, Dr. Lei Ye, for his guidance and support throughout my study and during the completion of this thesis.
- I am also extremely grateful to my co-supervisors Dr. Wanqing Li, and Prof. Philip Ogunbona, for helping me through the most important years of my life. This thesis would not have been possible without their encouragement, consistent efforts and true desire to keep me on track.
- I would like to acknowledge the students and other staff members in SITACS, especially the postgraduate students in my research lab, Fenghui Ren, who taught me a lot about image processing and information retrieval, and Liang Lu, who taught me many skills in research methodology. Also, thanks to Yi Gao, Sherry and her husband, and all the other friends who devoted their time to test my programs and to be my participants in the human perception experiments.
- And last but not least, I would like to thank my family for supporting me in my educational pursuits and to all my friends for their suggestions and encouragement.

Without the help of so many people, the thesis would never have come to be. I am very glad that I can have the chance to express my sincere thanks to all of YOU.

# Chapter 1

## Introduction

### 1.1 Introduction to Image Retrieval

The prevalence of digital cameras has brought about an exponential growth in the amount of images and videos found on the World Wide Web and in both private and commercial collections. These digital images and videos have found useful application in many fields including business, education, defence and security, to mention a few. The large volume of image and video archives has necessitated the development of efficient methods to enable searching, locating and retrieval. The research community has responded to this challenge and we have witnessed the development of a plethora of techniques and working systems in the last two decades. Image retrieval has become a very active research area since the 1970's, with the thrust from two major research communities: Database Management and Computer Vision. There are two main methods employed in image retrieval, namely, Text-Based Image Retrieval and Content-Based Image Retrieval(CBIR). Figure 1.1 shows the basic procedure of image retrieval.



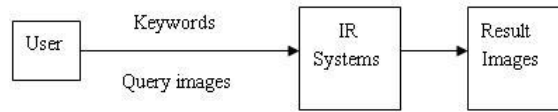


Figure 1.1: Basic Procedure of Image Retrieval

#### 1.1.0.1 Text-Based Image Retrieval

The text-based image retrieval is a well-known method that can be traced back to the late 1970's. It is still widely used today and most current large-scale web image search engines exploit text to search for images. Text-based image retrieval requires the existence of appropriate text annotation or indexing of the images in the database of interest. Users are then required to pose their query as a combination of text tokens or keywords. Many keyword-based text information retrieval systems have been implemented and have achieved great success for indexing image collections [26]. Kodak Picture Exchange System (KPX) and PressLink are two examples of such systems [1]. They are able to represent general and specific information about the objects in images.

Textual information, such as filename, caption and description, are often used to represent the image in a text-based image retrieval system. However, textual representation of an image may sometimes be non-informative and could provide ambiguous description of image content. The name of the image file might be misleading, the words in caption and description might contain multiple senses and cause misunderstanding as well. To overcome this problem, some text-based image retrieval systems are implemented so as to include further user interaction. The feedback from the user is passed back to the system and the system could adjust the retrieval result based on the feedback. [82] proposed an interactive text-based image retrieval system using user term feedback. The system uses the feedback from a user on specific terms

regarding their relevance to the target information. The system proposed in [49] is another example of text-based image retrieval approach applying relevance feedback technique. Based on the use of the feedback approach, misunderstanding between the indexers and the users could be reduced.

#### **1.1.0.2 Content Based Image Retrieval**

In the early 1990's, since the use of large image databases, the disadvantages of text-based image retrieval became apparent. There are many limitations inherent in metadata-based systems. Textual information about images can be easily searched using existing technology, but requires humans to personally describe every image in the database. This is impractical for very large databases, or for images that are generated automatically, e.g. from surveillance cameras. It is also possible to miss images that use different synonyms in their descriptions. Systems based on categorizing images in semantic classes like "cat" as a subclass of "animal" can reduce such issues but we still have to face this problem. In order to overcome these disadvantages, content-based image retrieval was proposed.

Content-based Image Retrieval(CBIR), also known as Query by Image Content(QBIC) and Content-Based Visual Information Retrieval(CBVIR), is the application of computer vision to the problem of image retrieval. The term CBIR seems to have originated in 1992, when it was used by T. Kato to describe experiments into automatic retrieval of images from a database, based on the colours and shapes present [71]. Since then, the term has been used to describe the process of retrieving desired images from large databases based on image features. Rather than use manually annotated text-based keywords, visual content, such as colour, texture or shape is used to index the images. It is now a very important area of research and has been given tremendous importance during the last decade [17]. Typically, such a CBIR system extracts the visual features from a given query image and stores it. The visual features are then used to perform

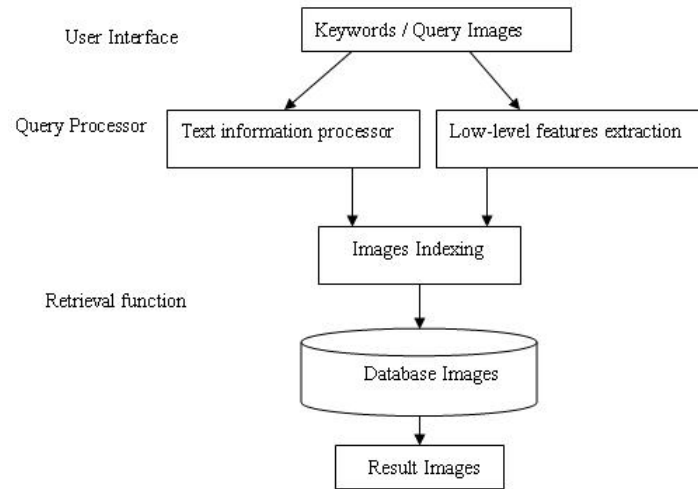


Figure 1.2: An example of CBIR system architecture

the comparison with the features of other images stored in the image collection. The similarity function used to calculate the distance between images is thus based on the abstracted image content rather than the image itself. Figure 1.2 gives an example of CBIR system architecture.

Different categories of CBIR methods can be identified, depending on the purpose and approach. Based on the number of queries: Single Query Based/Multiple Query Based Image Retrieval. Based on number of features used: Single Feature Based/Multiple Feature Based Image Retrieval. Based on the level of human perception: Low-level Features Based/Perception Based Image Retrieval.

Several typical CBIR systems are now described briefly. QBIC [21][61] was developed at the IBM Almaden Research Center. It is the first commercial CBIR application and plays a vital role in the evolution of CBIR systems. The QBIC system supports low level image features of average colour, colour histogram, colour layout, texture and shape. Additionally, users can provide pictures or draw sketches as example images in a query. The visual queries can also be combined with textural keyword predicates. VisualSEEk [75] is a highly functional prototype system for searching by visual

features in an image database. The novelty lies in that the user forms the queries by diagramming spatial arrangements of colour regions. The system finds the images that contain the most similar arrangements of similar regions. Prior to the queries, the system automatically extracts and indexes salient colour regions from the images. By utilizing efficient indexing techniques for colour information, region sizes and absolute and relative spatial locations, a wide variety of complex joint colour/spatial queries may be computed. CIRES, mentioned in [34], is a robust image retrieval system that serves queries ranging from images containing conspicuous structure, such as buildings, bridges and towers, to images containing purely natural objects, such as vegetation, water, sky and clouds. PicSOM [44] is a content-based image retrieval system using technologies such as pictorial examples, relevance feedback, vector quantization. The main indexing method used in the system is called Self-Organizing Map (SOM) [41]. The SOM method defines an elastic, topology-preserving grid of points that is fitted to the input space. Usually it can be used to visualize multidimensional data on a two-dimensional grid. PicSOM uses a special form of SOM which is called Tree Structured Self-Organizing Map (TS-SOM) [42]. During the retrieval process, neural units(images) which possess similar characteristics are located together on the TS-SOM layer surface. Then both positive and negative units are separated from each other by user's interactions. The MPEG-7 visual descriptors are employed in this system. PicToSeek [24][25] automatically collects, indexes and catalogues visual information entirely on the basis of the pictorial content. It allows for content-based image retrieval conducted in an interactive, iterative manner guided by the user by relevance feedback. The client takes care of interactive query formulation and the relevance feedback specification given by the user. The server takes care of the image feature extraction, feature weighting from relevance feedback, k-nearest neighbor feature classification, and image sorting. Photobook [66] was developed at the MIT Media Lab.

It is a set of interactive tools for searching and querying images. It is divided into three specialized systems, namely Appearance Photobook (face images), Texture Photobook, and Shape Photobook, which can also be used in combination. The features are compared by using one of the matching algorithms. These include Euclidean and Mahalanobis distances, divergence, vector space angle, histogram, Fourier peak, and wavelet tree measures, as well as any linear combination thereof. NETRA [52][53] is a prototype image retrieval system that has been developed at the University of California, Santa Barbara (UCSB). NETRA supports features including colour, texture, shape and spatial information of segmented image regions to perform region-based search. Images are segmented into homogenous regions. Using the region as the basic unit, users can submit queries based on features that combine regions of multiple images. For example, queries like “All images that contain regions having colour of a region of image  $I_1$ , texture of a region of image  $I_2$  and shape of a region of image  $I_3$ ”.

## 1.2 Benchmark datasets and performance indicators

### 1.2.0.3 Performance indicators

Precision and Recall Rate are the most common methods used to evaluate the performance of CBIR Systems [20][35][45][73].

For a query image, a retrieved image is relevant if it belongs to the same category as the query image. Retrieval Precision is defined as the percentage of retrieved relevant images in the top-N retrieved images.

$$precision = \frac{Number\ of\ relevant\ images\ retrieved}{Number\ of\ total\ images\ retrieved}$$

Recall Rate is the percentage which indicates the number of relevant images retrieved over the number of relevant images in the answer set.

$$recallrate = \frac{Numberofrelevantimagesretrieved}{Numberoftotalrelevantimages}$$

Systems in [77] use precision and recall as the performance indicator. In [37], average precision at the first 50 retrieved images(P50) and average precision at the first 100 retrieved images(P100) are used as evaluation metrics. In [76] and [85], average precision is also used to evaluate the performance.

In [50], two complementary measures for precision and recall are proposed based on the relevance feedback technique. Actual Recall is the percentage of relevant images retrieved at each iteration over the number of relevant images in the feedback images, which is also called the answer set. Actual Precision is the percentage of relevant images retrieved at each iteration over the number of retrieved images at each iteration. New Recall is the percentage of relevant images that were not in the set of the relevant images retrieved during previous iterations over the number of relevant images (Measured after the first iteration). New Precision is the percentage of relevant images that were not in the set of the relevant images retrieved during previous iterations over the number of retrieved images.

#### 1.2.0.4 Datasets

Most CBIR Systems choose COREL Image Database as the testing database. Its popularity is exemplified by the number of publications that have used it to benchmark their proposed algorithm or system. These include, [20, 35, 38, 50, 60, 84] and [7, 30, 37, 45, 85, 76]. It is noted that the database is not available for free download and is very expensive.

Other popular databases include the public Stanford10k2 dataset[50], the “Wash-

ington” image database[77] and S3 data set of MPEG-7[36].

There is no agreement in the research community as to which database is standard. The United States National Institute of Standards and Technology (NIST) has made significant effort to provide “standard” data set and query set that researcher can use to benchmark their proposals. However, these are for video retrieval.

## 1.3 Issues in Image Retrieval

Although there has been significant improvement in both text-based image retrieval and content-based image retrieval, there are still many limitations. Based on the procedure of the image retrieval, we can divide the issues into three categories: Defining the database, specifying the query and similarity criterion definition.

### 1.3.0.5 Defining the Database

There are several approaches to specifying the database in image retrieval. One is to use textual information such as caption and description to be the index of the images. This needs a large amount of manual annotation but describes the images more accurately. However, it is very difficult to compare two images with such text information.

Another approach is based on attribute representation proposed by database researchers where image contents are defined as a set of attributes which are extracted manually and are maintained within the framework of the conventional database management systems. [81] proposed a technique that perform rapid subset selection within large image collections. They used text-based image retrieval techniques to index extracted MPEG-7 features of images (using the MPEG-7 eXperimentation Model) and select subset from image database. They then compute similarity measures on the subset and obtain a ranking of images. Although it makes the comparison easier by

using formatted image features, it still requires a large amount of human labor in the manual annotation.

The third approach uses images directly. It depends on an integrated feature-extraction / object-recognition sub system to overcome the limitations of attribute-based retrieval. This kind of system automates the feature-extraction and object-recognition tasks that occur when an image is inserted into the database. These automated approaches to object recognition are computationally expensive, difficult and tend to be domain specific [17].

#### **1.3.0.6 Specifying the Query**

Content-based image retrieval is considered as a bottleneck in the access of multimedia databases [17] because there still remains a vast difference in the perception capacity of human vision and computer vision. It is difficult for users to specify visual queries through the use of low-level visual features. Low level features of images are poor and inadequate in precisely describing and presenting user queries. In essence there is the so called “semantic gap” that separates the users’ intention at the semantic level and the description provided in the low-level feature space.

Compared with content-based image retrieval, the queries in text-based image retrieval can describe the users’ semantic expectation better. They are more intuitive and more natural for the users to specify the information they need. Text-based image retrieval has several limitations that make it non-ideal for retrieval tasks. It is time consuming since it requires manual indexing of the entire database content. This may lead to inaccurate retrieval results. An image may have different meanings to different users. Moreover, it could also mean different things to the same user at different times and in different circumstance. The interpretation of the image may differ between the person who did the indexing and the user of the database who issues the query. Furthermore, when given the same image, people could well use different words to



describe the content of the image. So we have to manually label each of the image in the database with suitable title, caption, descriptions or other key words. If an image is not accompanied by sufficient, meaningful metadata, it is unlikely to be found.

#### 1.3.0.7 Similarity Definition

The last issue and by no means the least important in image retrieval is how to compare images to find out if there is some perceptual congruence. Similarity is computed by calculating the distances in the feature space and the results are used to rank the images returned to the user.

For text based image retrieval, the comparison is very difficult since the task of describing image content is highly subjective, this approach suffers from low keyword agreements between the indexers' and the users' queries.

With the help of pre-defined feature descriptors of the image, CBIR can perform much better in image comparison. Given the query image and database images, the system employs the same pre-defined feature descriptors to perform the comparison.

## 1.4 Human Perception in CBIR

The final objective of content based image retrieval is to find images that are perceptually similar to a given query. In an ideal situation, image retrieval systems should have the ability to extract all the relevant semantic features from the image in the same way as a human being does. The perception features include objects, illumination, camera position or zoom, or any combination of these semantic aspects. Most current CBIR systems use low-level features such as colour and texture to calculate the similarity between images. Although some of them are quite successful, there is still a long way to go to achieve the ultimate goal. Given an image, it is conceivable that users may have different perceptions and hence interpretations of the image content.

It is expected that this will influence the semantic value of the image content and subsequently the way a content-based query is posed. There is a need to consider human perception when designing retrieval systems. Attempts to develop perception-based image retrieval systems has been witnessed in recent years. These systems extract the low-level features of images and combine them into high-level semantic features such as regions or objects.

## 1.5 Contributions of the Research

This thesis focuses on empirical research into human perception of similar images and its relationship to low-level image features, and algorithm development for weighting the features to match human perception in QBMI. Specifically, the contributions include:

- A set of human perception experiments was designed and conducted to gain insight into human perception of similar images. Analysis of the experimental results by using MPEG-7 descriptors reveals that humans tend to perceive similar images by using multiple low-level visual features with each feature being weighted differently according to its ability to describe the content of the images.
- A novel method to assign weights to different features has been proposed for QBMI, employing the concept of Mutual information based feature selection and distance entropy. Experimental results have shown that the weighing method is more effective than both equal weights and heuristic weighting method.
- A QBMI system is implemented based on the proposed weighting method. The system adopted MPEG-7 descriptors as the low-level image features and performs satisfactorily for a wide range of images.

## 1.6 Organization of the Thesis

The thesis is organized as follows

Chapter 1 presents an overview of content-based image retrieval and highlights the challenges in this field.

Chapter 2 is a literature review on the relevant research topics including human visual perception, visual features. Query-by-Example, relevance feedback and Query-By-Multiple-Images.

Chapter 3 describes a human perceptual experiment system and reports a set of psychological experiments that were specifically designed to verify how human beings perceive similar images. The results obtained by human subjects were compared with those obtained by using MPEG-7 visual descriptors.

Chapter 4 presents two basic interpretations, *compositional* and *descriptive*, of how users may relate a set of example images with the images that they desire to search for in a Query By Multiple Images (examples) (QBMI) system. A new method is proposed to assign weights to different features for QBMI.

Chapter 5 describes a QBMI system based on the proposed new weighting method and reports its performance on an image database that consists of about 9,000 diverse images. The performance is measured in terms of precision and recall and compared with other weighting methods. Experimental results demonstrated that the proposed method outperforms the others.

Chapter 6 concludes the thesis with general comments and gives suggestions for future research and improvements.

# Chapter 2

## Literature Review

### 2.1 Introduction

When a user seeks an image of interest from a database there is usually some idea or description of the target in mind. This description may be clear, specific and conceived a priori. For example the query, “I want an image of a red car next to a white building”. It is also possible that the description is less specific and a broader concept is implied by the query. Such will be the case with, “a car of any colour”. Human perception plays a major role in image retrieval insofar as the description of the target is based on the perception of the inquirer (user).

Review of psychophysics of perception is carried out in this chapter so as to explore and document what is currently known about how humans use colour and edges in perception. Work in the computer vision literature is discussed here as well. Query by example is a very popular CBIR paradigm which assumes that the sample image given by the user embodies some of the perceptual cues that the user associates with the target images. The use of multiple example images in query by example is reviewed as well and its advantage is highlighted from the viewpoint of human perception reinforcement. A review of similarity measures is conducted with a view of highlighting

how to incorporate weighting emphasizing the importance of features. Methods of combining features are also reviewed.

## 2.2 Human visual perception and visual features

There are several perceptual cues that can be associated with different visual features used in image retrieval. The effectiveness of the use of visual features (especially colour, edge and texture) in retrieval is discussed and evaluated.

### 2.2.1 Theory of Human Perception

The human **mind** is considered as an information processing system[2] that interacts with the external world, thus making us active processors of information. Pre-acquired knowledge and other neural processes are combined to interpret the sensations impinging on the **mind**. It is important to consider the human mind when modeling the image retrieval process. Given an image, it is expected that users will respond differently, especially from an interpretation viewpoint, because of differences in pre-acquired knowledge and overall perception. Consider for example, the image in Figure 2.1.

This figure is a famous example of an ambiguous image that is prone to different perception. There seems to be a rabbit facing left and a duck facing right. The experience and expectation that it was a figure of a duck would make the viewer see it first as a duck rather than a rabbit, and may make it difficult to ‘reverse’ the interpretation and see it as a rabbit, and vice versa. It is possible for the same image (or images) to be selected by different users with different perception or focus on different parts of the image. For example, an image with a red car and possibly other objects may satisfy the query of a user seeking an image with red coloured objects and at the same time satisfy a user seeking an image with car-shaped object.

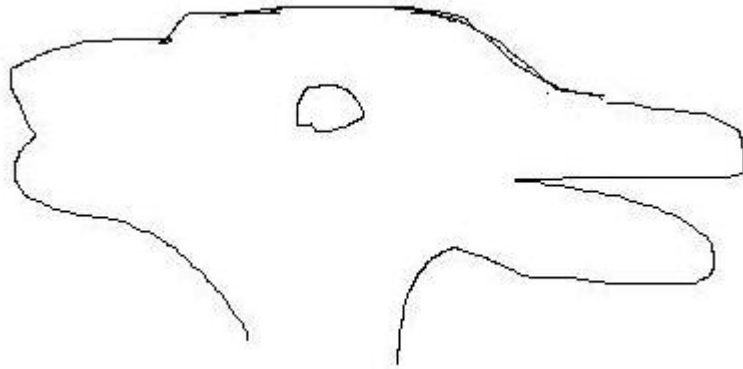


Figure 2.1: Duck/Rabbit figure

Based on these different purposes, using only one image is definitely not enough - it is sometimes ambiguous. Using multiple query images to define our intention to reduce the ambiguity is an approach to solve this problem. This will be discussed in the following sections.

Perception is part of human intelligence underlined by a hierarchical structure of different attention stages. The limitation of the information processing capacity of the human brain is somewhat compensated by the attention process whereby selection and selective processing of inputs take place. Visual attention, which is part of human perception, is the ability of a vision system to rapidly detect potentially relevant parts of a visual scene. Higher level vision tasks such as object recognition can then focus on the selected parts. [69] studied human attentional processes, and analyzed current approaches to the design of attention aware systems. They found that attention can be broken down into two stages. One stage is referred to as *pre-attentive* and the other is called *attentive*. First, a parallel (several stimuli may be processed at the same time) pre-attentive stage encodes simple physical properties of incoming stimuli. The recognition of the simple physical properties allows one to filter out all the irrelevant (unattended) stimuli and to pass on only selected stimuli to the second

stage. The second stage is capable of only limited processing and it encodes more abstract properties such as semantics of the attendant stimuli. In essence human perception is the result of both bottom-up and top-down processing.

In human perception, people usually notice the low-level features of an image and then combine them to be objects in their mind. This is called “Bottom-up” processing. Bottom-up processing begins with external input and travels ‘up’ through the cognitive system. The brain builds the perception of objects by combining the perception of features through attention. For example, patterns of light can be transformed into edges, and furthermore information about objects. In bottom up processing, the fundamental units of perception is at the level of features, not at the level of objects. Features are combined into more complex objects by attention.

However, [31] proposed that explicit vision advances in reverse hierarchical direction, as shown in perceptual learning. They proved that conscious perception begins at the top of the hierarchy, gradually returning downward as needed. This is called “Top-down” processing. “Top-down” processing refers to the higher-level cognitive elements (goals, intentions, expectations, knowledge etc.) on lower-level processes. In this model, one first sees unified whole images, then the features are perceived through attention. Consider the two images in Figure 2.2.

A first glance at these two images may result in their being treated as similar images because one only notices the object-level features, namely, two houses with the same shape. Further attention will result in seeing some details of these two houses and realization that there are some differences between them. For example, different items on the grass, different positions or shapes of the windows and chimney, and different steps to the door. This observation may be said to prove Hochstein and Ahissar’s reverse hierarchical direction of perceptual learning theory. Based on this theory, most CBIR approaches that initially use low-level features of the image cannot

Figure 2.2: People will see two images with similar objects in it. Then they will find some differences when they look into more details [31]

be said to model human perception well. A better approach is to guide the system by identifying or pointing out the object of interest. Since there are usually many objects in one image, it is hard to tell which is the object of interest and this may be different from one user to another. Hence the use of one image is not enough to identify, algorithmically, to the system that may be of interest. It is conceivable that the use of multiple query images could achieve the desired goal of determining the object of interest. By offering multiple queries, with the same object we want but different background, the system could identify which information is useful and which is not important for a certain retrieval. This also obeys human vision perception theory - we can recognize the same object in different images easily; the details of the object are considered in the second level.

In order to retrieve images based on human perception, we determine which features best match human perception. Processes in the pre-attentive stage are responsible, chiefly, for the perception of colour and edges. [78] used a visual search task was used to show which features were important at a perceptual level, that is, which features



formed the building blocks of human perception. The result showed that the features include the so called primitive features, namely, colour, orientation, curvature and line intersections. Individual features are combined into objects and it is recognized that feature combination requires attention to bind the features together. Different visual features which affect human perception are discussed in the following section.

### 2.2.2 Visual Features

All the approaches used in CBIR systems employ colour, texture, shape, spatial constraints, and other information of the image content, for retrieval. Some of the commonly used features are now presented.

#### 2.2.2.1 MPEG-7 Standards

The World Wide Web holds vast amount of multimedia content but lacks adequate strategy to provide seamless access across the media types. This problem is due in part to the unavailability of suitable indexing standards.

The MPEG-7 [27][28] standard is being developed by the Moving Pictures Expert Group (MPEG), a working group of ISO/IEC. The goal of the MPEG-7 standard is to provide a rich set of standardized tools to describe multimedia content. It standardizes descriptors for defining syntax and semantics of each feature representation and description schemes for specifying relationships between components (both descriptors and descriptor schemes). MPEG-7 automates the process of extracting features from a multimedia data.

The data is structured using the MPEG-7 Description Definition Language (DDL), which is an XML based mark-up language. The DDL information is written to a MPEG-7 bitstream file. When a search is performed, features are extracted from the reference image / movie / audio clip and compared to the features listed in the

bitstream file [5, 57].

#### 2.2.2.2 Colour Features

Colour perception is an important aspect of human perception. Based on colour difference, we can distinguish an object from its background as well as from among various objects in the environment [23].

Colour is one of the most expressive features among all the visual features and have been extensively studied in the image retrieval literature. [39] conducted experiments to compare computer models of visual attention with human attention. The contribution of colour in visual attention was quantitatively measured as the increase in similarity when the one-cue computer model for gray scale is replaced by the two-cues computer model for colour. The result of their experiment suggested that colour has a considerable influence on human visual attention and is very effective for the calculation of image similarity. It is not surprising that in many image retrieval tasks, colour is considered to be a very important feature. Another advantage of the use of colour is that the colour of an object is usually independent of viewing position or viewing distance.

The representation of colour is very important in colour image processing and in image compression in particular. Its representation plays a crucial role in image retrieval efficiency and in the required computational complexity. The characterization of colours in an image should be related to its perception, coherency and spatial distribution. An important aspect of specifying colour features is what colour space to use. For example MPEG-7 supports three additional colour spaces, apart from the usual monochrome, RGB and YCrCb. These are the HSV, the HMMD and the linear transformation matrix with reference to the RGB. Both the HSV and the HMMD represent non-linear transformations. They are meant to be closely related to the human perception of colour and hence can provide better results in searching and

retrieval. The fact that there is an arbitrary linear transformation with respect to RGB, means that a large number of additional colour spaces can be defined. The choice of a colour space for representation still leaves room for some redundancy as the number of distinct colours may be excessive for the task of retrieval. Usually, the colour space is quantized to achieve an efficient representation.

Quantization is the reduction of the number of unique colours in an image and may be achieved through linear, non-linear or lookup table transformation. In the linear case, the normalized colour value range is divided into equal intervals. Each quantized colour is represented by a colour value index that can be decoded to the correct colour value in three components according to the quantization type used. When a lookup table is used the quantization can be performed instantly, otherwise it has to be calculated from predefined formulas and the current colour index.

MPEG-7 defines five colour descriptors, namely dominant colours, colour layout, group of frames / group of pictures colour histogram, colour-structure histogram and scalable colour.

**Scalable Colour:** The Scalable Colour Descriptor is a Colour Histogram in HSV colour space, and it is encoded by a Haar transform. It has a binary representation that is scalable in terms of the specified number of bins and accuracy of the bit representation. The Scalable Colour Descriptor is useful for image-to-image matching and retrieval based on colour feature. Retrieval accuracy increases with the number of bits used in the representation.

**Dominant Colours:** In specifying the dominant colour descriptor, colour quantization is used to extract a small number of representative colours in each region or whole image. The percentage of each quantized colour in the region (or whole image) is calculated correspondingly. A confidence measure on the entire descriptor is also defined, and is used in similarity retrieval. This descriptor finds

application in representing local (object or image region) features where a small number of colours is enough to characterize the colour information in the region of interest.

**Colour Layout:** The descriptor consists of DCT (discrete cosine transform) coefficients of an 8x8 icon, which is defined as a set of dominant colours on an 8x8-grid layout. Scalability is achieved by specifying the number of coefficients included in the descriptor. In general it requires only 64 bits in default setting to describe spatial distribution and its similarity calculation process is simple enough to achieve very high-speed retrieval. It can be used for both natural images and sketches.

**Colour-Structure Histogram:** This descriptor embeds local colour structure information into the histogram. In other words, the extraction method takes into account the colours in a local neighborhood of pixels, rather than considering each pixel separately.

**GoF/GoP Colour:** The Group of Frames/Group of Pictures colour descriptor extends the Scalable Colour descriptor defined for a still images to the description of a video segment or a collection of still images. This is achieved simply by aggregating colour histograms of all frames and pictures with proper normalization. The choice of aggregation method include average, median or intersection (minimum of bin values across images) and is usually indicated by two bits. The Haar transform is applied on the aggregated histogram.

Although colour is a very useful visual feature, it is hardly adequate as the only cue in image retrieval tasks. It is conceivable that a user might be interested in both colour and texture as a description of the object of interest. In general content based retrieval systems will employ not only colour information, but also other features such

as texture and spatial relationships.

### 2.2.2.3 Texture Features

Similar to color feature, texture is another very expressive visual feature. It is a powerful low-level descriptor in image search and retrieval applications. Three texture descriptors are defined in MPEG-7 standard, namely, Texture Browsing Descriptor, Homogenous Texture Descriptor and Edge Histogram Descriptor.

**Texture Browsing Descriptor:** This descriptor is computed from images by filtering with a bank of orientation and scale tuned filters (Gabor functions are usually employed) and identifying the two dominant texture orientations from the filtered outputs. Next, the filtered image projections along the dominant orientations are analyzed to determine texture properties of regularity and coarseness.

The Texture Browsing descriptor is useful for representing homogeneous textures for browsing type applications, and requires only 12 bits (maximum). It provides a perceptual characterization of texture, similar to a human characterization, in terms of regularity, coarseness and directionality.

**Homogeneous Texture Descriptor:** The Homogeneous Texture Descriptor provides a quantitative representation using 62 numbers (quantified to 8 bits each) that is useful for similarity retrieval. The extraction proceeds by applying a bank of orientation and scale tuned filters (usually Gabor functions) to an image. The first and second moments of the energy in the frequency domain in the corresponding subbands are then used as the components of the texture descriptor. The number of filters used is  $5 \times 6 = 30$  where 5 is the number of “scales” and 6 is the number of “directions” used in the multi-resolution decomposition using Gabor functions.

**Edge Histogram Descriptor:** This descriptor represents the spatial distribution of five types of edges, namely four directional edges and one non-directional edge. The extraction proceeds by applying spatial edge filters to the image and quantifying the output for each direction. Edges play an important role in the visual perception of images and thus can be used to retrieve possibly semantically related images.

Texture can provide an important and useful cue in CBIR but in cases where the texture is not well defined it may prove inadequate. In general a combination of texture descriptors will be required to capture the texture features in an image. [64] compared several different CBIR methods. They used perceptually-derived criteria and rank correlation to evaluate the textural computational methods, and draw the conclusion that general image content-based retrieval methods do not match human perception to any great extent.

#### 2.2.2.4 Shape Features

Arguably, shape features contain the most attractive visual information for human perception as they can provide a powerful clue to identify objects. This characteristic distinguishes shape feature from other low-level visual features since neither colour nor texture alone can fully identify an object. MPEG-7 defines three shape descriptors.

**Region Shape:** This descriptor makes use of all pixels constituting the shape within an image frame, thus making it useful in describing any shape. In particular it can describe shapes with a single connected region as well as complex shapes that consist of holes in the object or several disjoint regions. The Region Shape descriptor is also robust to minor deformation along the boundary of the object.

**Contour Shape:** The Contour Shape descriptor can capture shape features of an object on its contour. It is based on the Curvature Scale Space(CSS) representation

of the contour, which captures perceptually meaningful features of the shape.

**Shape 3D:** The 3D Shape Descriptor provides an intrinsic shape description of 3D mesh models. It exploits some local attributes of the 3D surface. Only a brief description is give here as 3D models are outside the scope of this research.

Apart from MPEG-7 descriptors, several other shape representation approaches have been proposed. For example, [46] introduced a shape representation method refereed to as polar representation and distance sequences. In this representation the contour is characterized through a sequence contour point in polar coordinates (see Figure 2.3). If  $n$  is the number of contour points in an object, the pole, considered to be the centroid of the shape, can be calculated as,

$$x_c = \sum_{i=1}^n \frac{x_i}{n}$$

$$y_c = \sum_{i=1}^n \frac{y_i}{n}$$

The contour graph can be represented by a polar equation  $d = f(\theta)$ . This polar description allows us to represent the contour points as a sequence,  $(d_0, d_1, \dots, d_{n-1})$ , where  $d_i = f(\theta)$  and  $\theta_i = \frac{i \times 2\pi}{n}$ . The distance sequence is obtained by successively rotating a ray emanating from the pole through a fixed angle  $\Delta\theta$ , where  $\Delta\theta = 2\pi/n$  for a positive integer  $n$ .

The matching of two shapes is achieved by calculating the Euclidean distance between two sequences. Thus,

$$d(U, V) = \|V - U\| = \sqrt{\sum_{i=0}^{n-1} (v_i - u_i)^2} \quad ,$$

where  $U, V$  are respectively, the query image and database images,  $v_i, u_i$  are their  $i_{th}$  features respectively, and  $n$  is the dimension of the feature space.

Figure 2.3: An example of the Polar Shape Representation [46]

### 2.2.3 Perception Based CBIR Systems

Although the goal of all image similarity metrics is to be consistent with human perception, relatively little work has been done to incorporate human perception of similarity in CBIR systems in a systematic manner. Recent research results indicate that attention is now being devoted to the development of systems that try to infer users' intention and how this might be used to guide the output of retrieval applications.

[9] proposed a CBIR system for retrieving medical images with focus on objects. This system incorporates models of human perception and is used to guide the search for an optimum similarity function. Twenty shape features are computed for each region after the image is segmented. The features are used to construct a feature matrix which is subjected to a principal component analysis for dimensionality reduction. They use a human perception similarity experiment to obtain a human response matrix. The distance function is designed using a genetic algorithm.

[62] proposed an image retrieval approach based on the assumption that the appreciation (or visual memory) of an image occurs in two sequential stages. In the first stage, there is a rough gaze over the whole image and in the second stage attention is focused on specific objects within the image. Based on this assumption, a three layered image expression model which consists of domain, object layout and objects' layout is proposed. A three-layer model of the visual memory then guides the retrieval task. Retrieval based on areas or regions is achieved by comparing the colour



of sampling points whose colour space is considered as three dimension of the RGB intensity values. The retrieval based on object layout, which is also considered as the spatial relationship of the objects, is achieved through a comparison of arcs between two objects. Retrieval based on object attributes is considered important only when the number of objects in the image is small.

[10] proposed a perception-based image retrieval architecture that can dynamically construct a pipeline of filters to meet the requirements of a search task and adapt to individuals' search objectives. They first design the filters for individual images. The filters includes colour masks, pixel masks, shape filters, etc. They investigated how some of the psychological and physiological invariants affect human perception and how they can be considered in characterizing visual data and in measuring similarity for a filter. They treat the human visual system as being divided into two parts: the eyes (the front-end) perceive and the brain (the back-end) recognizes images. Visual data is collected by the front-end and back-end can process the data with different filters, which can be treated as a particular way of perceiving an image. The front-end responds flexibly in perceiving visual data by selecting, ordering and adjusting visual filters differently.

[60] implemented an image indexing system and evaluated it based on some of the known properties of the early stages of human vision. The system was evaluated by comparing the similarity judgments made by the algorithm to judgments made by human observers. They quantitatively measured the relationship between the similarity order induced by the indexes and perceived similarity. The experiment results show that the rank orders produced by their system predict the perceived similarity between images. They also proved that, combining different information sources substantially improved the correspondence with the observers. The luminance and the Fourier histogram both contribute to the similarity judgments and the percentage of agreement

increases considerably if the luminance and the Fourier information are combined with the chromaticity index. A total of 60,000 digital images from the COREL database was used in the experiments. Their results show that psychophysical methods should be and can be successfully used to evaluate and compare different CBIR systems.

[55] proposed a model for weighting image objects in home photographs. The model is designed according to human perception criteria about what is estimated as important in photographs. An image object is considered as a 2D representation of a visible, real object which is part of the scene displayed in a photograph. The aim of their research was to evaluate how important these objects are when considering the image relevance. According to the experiment result, users tend to describe images based on visible objects appearing in them. So the visible objects are considered as significant in images and provide better description than mere colour regions. They define the notion of importance of a given image object relative to the notion of image content. They considered four features in the description of the characteristics of an image, namely, Size, Position, Fragmentation and Homogeneity. The conclusion drawn from the experimental results was that the importance of an image object will change based on its size. Furthermore, the importance of an image object is maximal when its position is in the center of the image, and decreases when the distance from the image center increases. Their results also indicated that the importance of an image object is maximal when it is not fragmented, and decreases when its fragmentation increases. With respect to homogeneity, the importance of an image object changes in the same way as the homogeneity changes.

## 2.3 Query-by-example

Query specification is an important aspect of image retrieval requiring significant attention. The specification should be expressive enough to allow the user to pose a

query that reflects their intention. Text-based query is perhaps the easiest method for users to pose a query to an image retrieval system. However, as the popular saying indicates, a picture is worth a thousand words. This indicates a drawback in the use of text-based query systems because different users may use different words to describe the same image and comparison becomes a challenge. A compromised approach is to use an image instead of textual information as the query. This approach is commonly referred to as Query-By-Example(QBE). Although using an image is not as good in describing a desired image as using text, it makes the comparison much easier. Computers can compare the extracted features of images easily.

QBE is a method of query presentation that allows the user to search for images based on an example. Most CBIR systems adopt the QBE concept. The idea of QBE in image retrieval is that, given an image query (supplied by the user or chosen from a random set) to the CBIR system, the system will extract various features such as colour, texture and shape from the query image to form feature vectors. Each of these individual feature vectors is then used to compare the feature vectors of the database images for similarity searching. Different kinds of features should be integrated because they may play different roles and have different relative importance when making the final decision. Most current image retrieval systems use a weighted Euclidean distance to combine the similarity measurements of different feature classes. Some fusion methods that employ neural networks have also been used to find the weights among different features.

Image retrieval engine compares the feature vector of the query image with that of the database images and presents to the users the images of highest similarity in order as the retrieved images[73]. However, the elements in the feature vector carry different kinds of information: shape, texture and colour, which are mutually independent. Hence, they should be handled differently depending on their nature and

interpretation for the problem. Early work shows that most image retrieval systems apply Euclidean distance which is not good enough to represent the dissimilarity of images. One issue is how to combine the distance of multiple features. [6] proposed an approach to do that, using the following function to calculate the distance between two images with multiple features:

$$distance = \sum_i d_i$$

where  $d_i$  is the Euclidean distance of  $i$ th features of the images being compared. This is the Addition Distance of the images. Since different features may play different importance during the retrieval, a better function is:

$$distance = \sum_i w_i d_i$$

where  $w_i$  is the weight for the  $i$ th feature. Now the problem becomes how to select proper weight for each feature. That is still a very difficult problem. It is clear that for images with high dimensional features, the Euclidean distance is not good enough.

Some experiments show that similarity between two images are not usually judged by all possible features, which means even visually similar images may be not similar in some features. Two images may have similar textures but different colours, or similar colours but different shapes.

[74] proposed a similarity measure method. If  $K$  features out of  $M$  from two images match, the two images are considered similar. If the value of  $K$  is set too low, the similarity measure will be too generalized and flexible. But if the value of  $K$  is set too high, it will be the same problem as the Euclidean problem mentioned above. Although a high value of  $K$  will increase the precision, it may also fail to retrieve a sufficient number of images. A better approach will be to use different values for  $K$  in

different retrieval tasks.

Based on their previous study, [73] have proposed a human perception based similarity measure and an indexing scheme that exploits the features of a novel relevance feedback scheme. If the Euclidean distance is considered, the same distance corresponding to different set of features may not play the same importance or role. The elements of the feature vector carry different kinds of information and are to be treated differently as dictated by their characteristics. To cope with this problem, they map real values of features to character based tags. For example, if  $M$  is the number of features, each feature value in a certain range is substituted by a corresponding character in the set  $A, B, C, \dots$ . For example, if we have 8 features and 10 buckets. The feature vector is converted into a tag consisting of 8 characters, which may look like “BCAHI-ADG”. By doing this, the similarity between two images is measured by matching the corresponding features or subset of features based on the criteria suitable to them rather than using a single distance measures consisting of all the features. A counter, initially set to zero, is increased if a feature is matched and similarity is declared by comparing the count with  $K$ . The retrieved images may be order based on this count for top order retrieval.

In most QBE systems, images are represented and retrieved based on their low-level features. But for humans, retrieval is predicated on high level features such as objects. There is a gap between low-level visual features and high-level human perceptual concepts, which is called the “semantic gap”. Because of the gap between low-level features and high-level semantic meanings, different semantic objects may share similar low-level features, while similar objects may have very different low-level features. This introduces inaccuracies in the retrieval result. To narrow this “semantic gap”, many CBIR approaches using high-level features such as regions or objects have been proposed. We call them perception based image retrieval. Two common

perception based QBE approaches in CBIR are region based image retrieval and object based image retrieval. Some basic concept and examples of these two approaches will be introduced and discussed in the next section.

### 2.3.1 Region Based QBE

Based on current research on the human visual system (HVS), we know that the basic elements that carry semantic information are image regions which correspond to natural objects such as cars, horses or flowers. Of course this is based on the assumption that the image segmentation is ideal. According to the psychological research on HVS, humans could easily point out semantic objects even with ambiguous boundaries in an image and recognize them with the interaction of fundamental visual components such as colour and texture. Based on the HVS theory mentioned above, regions are considered to important feature elements in CBIR systems. Approaches which use regions are called region based image retrieval (RBIR). In order to support region-based image retrieval, we need to segment each image into several semantic regions. Then we can compare the regions instead of viewing each image as a whole to calculate the similarity. So segmentation plays a very important role in RBIR.

A fast yet effective image segmentation method called WavSeg [83] is used to partition the images. A wavelet analysis in concert with the SPCPE algorithm [11] is used to segment an image into regions. Both local colour and local texture features are then extracted for each image region. Thirteen representative colours are identified based on HSV value ranges. They are black, white, red, red-yellow, yellow, yellow-green, green-blue, blue, blue-purple, purple, purple-red and gray. For texture features, one-level wavelet transformation using Daubechies wavelets are used to generate the horizontal detail sub-image, the vertical detail sub-image and the diagonal detail sub-image. For the wavelet coefficients in each of the above three subbands, the mean

Figure 2.4: The Flow Chart of Genetic Algorithm [83]

and variance values are collected respectively as texture features. Thus, each object of the image has nineteen features. Then a genetic algorithm is applied to find optimal combination of clustering solutions. The flow chart of the genetic algorithm is shown in Figure 2.4

[76] proposed a hierarchical region-based image retrieval system. Region feature vector is hierarchically extracted from all different wavelet frequency sub-bands, which captures the distinctive features inside one region finely. First of all, automated image segmentation is performed in Low-Low (LL) frequency sub-bands of image wavelet transform which show low-resolution images. The boundaries between segmented regions are also detected to increase the accuracy of the region description. The region feature vectors are hierarchically represented by the information in all wavelet sub-bands which describe detailed image content with different spatial-frequency resolutions, and each feature component of feature vector is a combined colour-texture

feature. Six features are considered in their approach for clustering to reflect colour and proper texture information. The unsupervised K-means clustering algorithm is used in LL sub-band for the segmentation. HSV colour space is used because it is thought to be similar to human perception. A tree structure named spatial orientation tree is used to define the spatial relationship between coefficients in the low-frequency sub-band and high-frequency sub-band. More semantically significant information such as shape could be considered to improve the system. Retrieval precision was used to evaluate the performance of their approach.

Huang et al.[33] proposed a RBIR System which uses the luminance and chromatic components of HLS (Hue, Luminance/intensity and Saturation) colour coordinate system to form the feature vectors. When doing the retrieval, an image is segmented into regions by its feature vectors. The segmentation uses a method called c-means to decide the cluster centers of the feature vectors and group the similar feature vectors into same regions. By using region-based IR, the retrieval is close to human visual perceptual. Region based IR is considered to be an approach to bridge the gap between low image features and high-level semantic features.

[85] proposed a region-based retrieval system which applies unsupervised image segmentation to partition an image into regions. To segment an image, the system divides the image into  $4 \times 4$  blocks and extracts a feature vector from each block. Six features are used to form the vector. Three of them are average LAB colours of the block. The other three are the energy in the high frequency bands of the Harr wavelet transform, which is the square root of the second order moment of wavelet coefficients in high frequency bands. These three features shows the texture information of the block in different directions. The K-means algorithm is then used to cluster the feature vectors into several classes. Color , texture and shape are often used to describe regions. But there might be colour inaccuracy perceived by human vision, as well as



texture inaccuracy obtained by different texture measures. To improve the robustness and effectiveness, [85] proposed a fuzzy approach to calculate the region descriptors. In each segmented region, fuzzy logic is used to define the fuzzy colour, texture and shape. The fuzzy colour histogram, fuzzy texture and shape properties of the regions are calculated to describe the segmented regions. Once the fuzzy colour, texture and shape feature vectors are calculated, the distance between query image regions and database image regions can be computed. Then the distance between database images and query image can be calculated based on the distances between regions. 10,000 images from COREL are used as the testing set. The images are divided into 10 well-defined groups such as people, nature scene, building, etc. They compared the performance of their system with the IRM system [35] and geometric histogram based system [67]. For each group they randomly select 1000 images as queries submitting to the system and calculate the average precision for each group based on the returned top 100 images of these two approaches. Their experiments proved that their approach is much better than spatial histogram. It is also better than the IRM approach in most groups.

To reduce the search space, clustering is used to group a set of physical or abstract objects into classes based on some similarity criteria. In this way the search space can be reduced to a few clusters that are relevant to the query region. K-means is a traditional clustering method and has been widely used in image clustering. However, it is incapable of finding non-convex clusters and tends to fall into local optima especially when there is a large number of data objects. [84] adapted Genetic algorithm [32] to cluster image regions because of its robustness and ability to approximate global optimum.

### 2.3.2 Object Based QBE

RBIR has proven to be an efficient method in image retrieval. But there are still disadvantages. First, regions is a higher level feature of an image, but it sometimes does not match human perception very well. Region-based retrieval systems attempt to represent images at object-level. A RBIR system applies image segmentation to decompose an image into regions, which correspond to objects if the decomposition is ideal. The object-level representation is intended to be close to the perception of the human visual system (HVS). Many existing systems compare images based on individual regions. Although some systems allow users to query on limited number of regions, the query is performed by merging single-region query results. Actually, it is still very difficult to get an ideal segmentation result. As a result, most systems tend to partition one semantic object into several meaningless regions. Thus it is often difficult for users to determine which regions should be used for retrieval. For example, a man in an image might be segmented into several regions: head, body, arms, legs, etc. The system does not know how to re-organize these regions to get a man-like shape. Actually the feature which best matches human perception is object. As discussed in [23], much of perception is object perception. An object means a coherent, bounded volume of matter. Usually we use segmentation to get the objects. But the ‘objects’ mentioned here are meant to be segmented perceptual units. They are more primitive than the conceptual objects, or real-world objects, which we discuss verbally in daily life[18]. They are called proto-objects and do not need to correspond exactly with conceptual or recognizable objects. Instead, they reflect the visual systems’s segmentation of current visual input into candidate objects. To get more accurate objects, we need to group or reorganize these perceptual regions.

[45] proposed a similarity measure of images called Integrated Region Matching (IRM). This is developed based on region representations to reflect semantics more

Figure 2.5: IRM is more robust to poor image segmentation than traditional methods [45]

precisely. One of the advantages of this method is that it allows one region of an image to be matched to several regions of another image. Figure 2.5 shows the difference between IRM and traditional region-based matching.

A matching between two regions  $r_i$  and  $r_j$  is assigned a significance credit  $s_{i,j}$  ( $s_{i,j} \geq 0$ ). The significance credit indicates the importance of the matching for determining similarity between images. The matrix  $S = \{s_{i,j}\}$  ( $1 \leq i \leq n, 1 \leq j \leq m$ ) is referred to as the significance matrix. With the help of significance matrix, we can calculate the distance between two region sets (IRM Distance) using :

$$d(R_1, R_2) = \sum_{i,j} s_{i,j} d_{i,j}$$

where  $d_{i,j}$  is determined by local features such as colour, texture and shape characteristics of the regions. The significance is chosen based on the *area percentage scheme*.

This means the larger area the object takes, the more important we consider it is. Based on calculated IRM Distances between different region sets, we can compute the difference between images and finally rank the database images. They tested the system on a general-purpose image database from COREL including about 200,000 pictures. Average Precision is computed to evaluate the performance. The authors compared their system with WBIIS system [80] and reach the conclusion that on average, the precision of their system is higher than that of WBIIS .

Automatic analysis of image content is a challenging problem. The ability to extract and describe distinct objects in a complex scene is crucial for image understanding [34]. It is difficult to describe the content of some images using only limited words even for human observers. For example, an image with a building and some trees around it might be assigned to either a structural scene or a natural scene from a different viewpoint. The coexistence of natural objects and manmade objects exacerbates the computer perception of the images. It is difficult to establish complete boundaries between the objects of interest and the background objects. The difficulty comes from the fact that automatic segmentation is a difficult problem. Objects are hard to extract from images by using current segmenting techniques. We cannot perform object based image retrieval well until a complete automatic solution to the segmentation problem is achieved.

Some other disadvantages have also been pointed out by [63] from the context of the concept of gestalt. Gestalt psychologists are famous for their work on understanding how images are organized on the retina into objects of perceptual experience. It is a matter of perceptual organization. Sometimes the objects of conscious perception are not directly given in any simple or direct way in the retinal image, but must be constructed through activity of the visual nervous system. For example, given an image with a car behind a tree, you can perceive a single, unified car rather than two

halves of disconnected, independent objects. Based on Gestalt laws, or principles of grouping, objects are grouped together when they are close, similarly moving together, and so forth. Grouping is among the best known but least understood phenomena of visual perception. In image retrieval, if we want to get an object from the image, segmentation techniques are used. But based on current segmentation methods, we cannot achieve the grouping as well as human perception. In other words, it is difficult to automatically treat the two halves of objects as a unified one by using segmentation. [19] listed several points which show the complexities involved in recognizing an object. First, there are usually many different objects in the image and the objects always overlap with each other. We have to decide where one object ends and the next starts, which is very difficult. Second, objects can be recognized over a wide range of viewing distances and orientations. If we look at a round table from the top, we can see a round shape. But if we look at it from the same height, the shape would be very different. Third, objects often vary in their visual properties. If we want to search for a car, we may have many cars with different colour, size or shape. Based on the complexities listed above, it is not easy to do image retrieval using objects. Furthermore, these complexities assume we can get a good segmentation result. But actually we have not got an image segmentation method to perfectly segment the image into objects. So object-based image retrieval is not good enough to do the retrieval.

To solve the problems of object based image retrieval listed above, multi-query based image retrieval is introduced. For the first problem, we can use multiple images as the query, each of them has the object which we are interested in. Since the object of interest is the common feature, we can easily pick it up from the database images. For the second problem, we can offer the system queries with the same object as well. Object in these images are viewed from different distance and viewpoint. By doing this, we can find images with the object of interest. For the third problem, we can

also use multi-queries to specify the object we want. Since we may focus on a certain feature of the object, say colour, we can provide multiple images with the object of the same colour. Then the system would know better what we need and produce the result more accurately.

## 2.4 Relevance feedback

Most QBE systems use only one image as the query, which is sometimes not good enough to describe the user's intention. A computer does not have the idea how to map low-level features to high-level concepts as a human being can. The dilemma of a single image query can be illustrated as follows. If the user intends to retrieve an image identical to the query image, the search is meaningless as the required image is already at hand. On the hand, if the user intends to find images with some features in common with the query image, the system would be confused because it does not have a mechanism of knowing which features the user has in mind. According to the variety of users' experience, education or mood, even with the same query image, the results expected by the user might be very different. Conventional approaches used in CBIR are based on the assumption that relevant images are near the query image in some feature space. This is the basis of the cluster hypothesis [37]. However, semantically related images are often scattered across several visual clusters. Although traditional CBIR technologies can utilize the information contained in multiple queries, this is only a reformulation of the original query. As a result these strategies only get the images in some neighborhood of the original query as the retrieval result. This restricts the system performance. Relevance feedback techniques are generally used to mitigate this problem. Another reason to apply Relevance Feedback is that human's interests evolve during information exploration as they learn and discover more about the topic at hand. Based on this reasoning, although performing a single search will satisfy

Figure 2.6: The Flow Chart of Relevance Feedback Based Image Retrieval [34]

many users, there are those who will use the knowledge gained from such a search to form more complex searches. By specifying positive or negative feedbacks to the system, users can make the system know which feature plays more important role during the retrieval.

The concept of Relevance Feedback (RF)[70, 72, 87] has origins in the research in information retrieval. It is a query modification technique that tries to capture the users' demands through interactive feedback and query refinement [16]. It is a mechanism of providing input regarding the quality of retrieval to the image retrieval system after a query is performed. The system uses the input to return the query and, hopefully, get a better retrieval. The input is typically provided by specifying which of the retrieved images agree with the query, and which do not. There are several mechanism of feedback could be used by the system, for example, Cluster feedback and Multi-class feedback [34]. Since there is lack of a reliable framework for characterizing high-level features (semantics) of images and human perception, the use of RF can provide us an approach to learn case-specific query semantics. Figure 2.6 shows the flow chart of relevance feedback based image retrieval.

### 2.4.1 Approaches in Relevance Feedback

Before negative image examples are considered, the simplest approach of relevance feedback is to calculate the variance of each feature among the query examples[58]. Then the inverse of the variance becomes the weight of the feature. This gives higher weights to features in which example images are similar. If there is a high number of images, the whitening transform is the optimal choice. Later when negative examples are considered, the problems become two-class problems. The two-class means positive class and negative class. Positive images are those images that a user considers similar to the query image. Negative images are those images that are considered to be dissimilar by the user. With the help of negative feedback, the system can get rid of the images which user does not want.

Many approaches have been proposed in Relevance Feedback. The first thing to consider is the strategy of dividing feedback groups. [34] proposed two approaches to decide the feedback group, Cluster feedback and Multi-class feedback. Cluster feedback is essentially based upon the multi-image query paradigm. It uses only positive images in a modified framework of multi-image query. In a multi-image query, a user selects a number of query images before performing a query. In the case of cluster feedback, a user starts with a single or a multi-image query, and after the query selects a number of images from the retrieved set of images as feedback images. The selection of these feedback images corresponds to the users' judgment that he/she is not fully satisfied with all the results of a particular query, and now wants to use some returned images, which are judged by the user to be similar to the set of query images, to further refine the query. These selected images might or might not be added to the original set of query images for another query on the image set. This process can be repeated continuously until the user is satisfied with the query results, or a predefined threshold is met. Multi-class feedback is proposed as a classification problem. It builds upon the



concept of Cluster feedback by providing more than one cluster of feedback images. These sets typically represent images belonging to different classes that a user has envisaged. Multi-class feedback is basically the usual nearest-neighbor classification that is used frequently in pattern recognition. The positive images are assumed to fall in one class, whereas the negative image are considered to fall in as many classes as the user deems necessary. The classification problem is then reduced to the scenario that there is one relevant class, and all the remaining classes are irrelevant.

Most RF based systems adopt the second approach. One approach is to use Fisher's Discriminant Analysis (FDA) [56]. This approach, however, introduces undesirable side effects because it tries to cluster negative examples into one class. In the actual scenario, negative examples can be any types of images in the database other than the positive class. If we give such a set of negative images back to the system, the system will be confused and could not find out any common features for negative images. To overcome this drawback, [86] proposed a new relevance feedback algorithm, which takes negative examples into account effectively. They consider the relevance feedback problem as a  $(1 + x)$ -class problem, where  $(1 + x)$ -class means one positive (1) and multiple negative classes ( $x$ ). In this scheme, it is assumed that the negative examples are coming from an uncertain number of classes, while the positive can be clustered into one class. Their algorithm, named *Biased Discriminant Analysis* (BDA) is characterized by the following objective function,

$$W = \operatorname{argmax}_W \left| \frac{W S_{bias} W^T}{W S_W W^T} \right|$$

where,

$$S_W = \sum_{x \in C} (x - m)(x - m)^T$$

$$S_{bias} = \sum_{y \in D} (x - m)(x - m)^T$$

$C$  is a set of positive examples and  $D$  is a set of negative examples. In short, BDA tries to minimize within-class scatter matrix  $S_W$  of the positive example, while  $S_{bias}$  is keeping the negative examples away from the positive examples.

[59] extended Zhou’s work and implemented an image retrieval system called ImageGrouper. In this system, when the user specifies more than one groups as positive feedback, these groups are merged into one group for query purposes. However, this approach does not take full advantage of group-oriented interface. When doing a retrieval, a user might have a high-level concept in mind, for example, “beautiful flowers.” Such a concept could not be expressed by just one class of images. Although red flowers and white flowers may have common visual features, they also have very different features, namely colour. If the system tries to consider them as one image class, the colour features have to be discarded since it’s not the common feature they are sharing. This is not desirable because these features may be beneficial to retrieve a specific flower colour. So later, they extended the original system and proposed a new system which uses a new relevance feedback approach [58]. In the new approach, the problem becomes a  $(x + y)$ -class problem, which means the users could choose not only more than one group of negative feedbacks, but also more than one positive clusters as well.

[33] uses the relevance feedback approach in their CBIR system. The database images are firstly segmented into some regions. The system will do a similarity measuring process to calculate the distance between query image and incorporated regions. Then the system allows the user to select some regions they are interested in from the top 5 sampled images popped up by the system. These regions of interest are then fed back to the system. Based on the selected individual regions of query images, the overall similarity helps filter out some images which are irrelevant. This process can be repeated until the user finds the image they need.

Relevance feedback and region-based representation of images are two effective ways to improve CBIR system. Some researchers combine the advantages of both of these two approaches to achieve better accuracy. [20] proposed such an image retrieval system. The system uses relevance feedback approach based on region representation. They assume that every region might be helpful in retrieval and more important regions should appear in more positive examples. By combining the regions of the query image and positive feedback, we can get the optimal query at the next iteration of the retrieval and feedback process. The K-means algorithm is adopted to group the regions of all positive examples into a few classes. Each class corresponds to a new region of the optimal query. They selected 10,000 general purpose images from the COREL data set. Two test sets are chosen: 1. 1,000 images from total 79 groups of images. 2. 150 images from 10 selected categories. The selected categories are: Cat, eagle, elephant, flower, model, mountain, pyramid, sunset, train, waterfall. They compared the average precision within the first 30 retrieved images ( $P(30)$ ) of two test sets. The experiment result shows that their method is clearly better than [72].

To use more regional information in the retrieval, Wang et al.[35] proposed an image retrieval method using IRM (Integrated Region Matching) as the image similarity measure. IRM incorporates the properties of all segmented regions so that information about an image can be fully used to gain robustness and to avoid the side effects caused by inaccurate segmentation. In this approach, they segment the images using the K-means method and use the percentage of region area as the region importance based on the assumption that larger regions will play more important roles than smaller regions. The test set is 10,000 images from the COREL database. They form images using 10 categories, with each containing 100 pictures. Three statistics were computed for each query: 1) the precision within the first 100 retrieved images, 2) the mean rank of all the matched images, and 3) the standard deviation of the ranks of matched images.

It has been shown that the system they proposed is robust, faster and more accurate with existing algorithms.

Wang et al's approach which uses IRM is not good enough because semantic meaning is not very relevant to the region size. To reflect semantics more precisely, Jing et al.[38] proposed a novel region-based image retrieval method using relevance feedback. The new method, called Self-Learned Region Importance (SLRI), measures the similarity between images based on the region importance learned from user's feedback. The importance of the region is decided by user's feedback instead of the size of the region. In each iteration of the retrieval, the user can provide positive and negative examples to the system. The feedback is used to calculate new importance of the regions so that we can get more accurate results. When comparing two images, their regions are compared. They use the minimal importance of two matched regions because if one of the regions is insignificant, the matching of them is meaningless, even though the other region is very important. The algorithm is tested with about 8,600 images from the COREL database. The average precision within the first 100 retrieved images ( $P(100)$ ) is used to measure the performance. Compared with IRM, this approach is better in terms of average precision, especially with images with little but important region area.

Most CBIR systems perform retrieval based on a full image comparison. An image is shown to the system and the system returns all the images which are similar to the query. However, sometimes users might be interested in images which contain an image or an object similar to a query image. This is called sub-image retrieval. The difference between sub-image retrieval and object-based image retrieval is that, sub-image retrieval finds an image contained within another image, while object-based image retrieval searches for a region, perhaps the result of image segmentation. [50] proposed an approach called HTM (Hierarchical Tree Matching) which used a tree

to model a hierarchical decomposition of an image, encoding the colour feature sub-images which are in turn stored as an index sequence. The retrieval is effective and efficient by comparing the tree structures of query image and database images. They later improved their approach by using relevance feedback to learn the intentions of the user [51]. The user submits a query without concerning whether it is a tile or similar to any tile of any database image. The system retrieves the initial set of images which consists of database images. These images contain tiles similar to the query. 64 quantized colours in the RGB colour space is used to compute the distance. Positive and negative feedback examples are identified by the user and collected by the system. For each positive image, positive and negative examples are used to update the tile penalties of those tiles representing this image. The system will then update the query using positive images and their newly updated tile penalties. By using the revised query and new tile penalties for databases images, the system can re-rank the images and sort the results. The user can repeat these steps until he/she gets an ideal result. The test set consists of 10,150 colour JPEG images, selected from the public Stanford10k2 dataset and COREL datasets.

For the “semantic gap” problem, Relevance Feedback (RF) is widely used to incorporate the user’s concept with the learning process. As a supervised learning technique, it has been shown to significantly increase the retrieval accuracy. Most existing RF-based CBIR systems treat each image as a whole. However, sometimes users just concentrate on a semantic region of the query image. [84] proposed a CBIR system uses MIL (Multiple Instance Learning) to deal with the region based relevance feedback problems. The user can identify and label a returned image as “positive” or “negative” example. Then the system can judge which specific region of the query image the user is interested in. This is done by applying Diverse Density (DD) algorithm, which is proposed in the MIL framework [54]. With this DD approach, a function called DD

function is defined to measure the co-occurrence of similar instances(regions) from different images with the same label. Then we can find a point which is the closest to all the positive images and furthestmost from all the negative images. A COREL image database consisting of 10,000 images from 100 categories is used as the test set. The accuracy rates with different scopes, i.e. the percentage of positive images within top 6, 12, 18, 24 and 30 retrieved images, are calculated to measure the performance. As a result of the testing, the proposed system performs better than systems with two other relevance feedback algorithms: 1) Neural Network based MIL algorithm with relevance feedback. 2) General feature re-weighting algorithm

[30] proposed an adaptive distance computation method for Relevance Feedback based CBIR. By using this method, users can specify three types of groups, which are positive, negative or neutral groups. For each feature within these groups of query images, the range of the feature is computed and used to adjust its weight. The algorithm is described as follows. For each feature  $f_i$  in each positive, negative and neutral group, compute the lower bound ( $l_i$ ) and the upper bound ( $u_i$ ), and then compute its range distance ( $d_i$ ) =  $u_i - l_i$ . So we get three range distances for each feature  $f_i$ ,  $d_{ip}$ ,  $d_{ine}$ ,  $d_{inu}$  for positive, negative and neutral groups respectively. Then we compute  $d_i = \min\{d_{ip}, d_{ine}, d_{inu}\}$ . If  $d_i$  equals to  $d_{ine}$ , we set  $w_i$  to 0 and discard this  $d_i$ . The weight  $w_i$  can be calculated using the function:

$$w_i = 1 - \frac{d_i}{\sum_i d_i}$$

If  $d_i$  is equal to  $d_{inu}$ , we set the  $w_i$  using:

$$w_i = \frac{w_i}{\text{number of members in neutral group}}$$

Then the distance metric  $D(I, q)$  can be computed using the following equation:

$$D(I, q) = \sum |I_i - q_i| * (w_i / \sum w_i)$$

where  $(w_i / \sum w_i)$  is the normalization of  $w_i$  into the range  $[0, 1]$ . Finally we can sort the results in ascending order and list the top K images to the user.

### 2.4.2 Combining Partial Results

Just as with common CBIR approaches, using only one feature cannot achieve an ideal result. Multiple features are used in RF based image retrieval Systems. In previous work on relevance feedback, the weights are usually associated with different features (inter-feature), and feature components (intra-feature). Once a query is done, a user provides samples for positive images and negative images from the retrieved set of images. The weights should be automatically adjusted by using the feedback images to redo a new query and obtain a better result. Both positive and negative images are used in traditional approaches to adjust the weights. In each iteration of Relevance Feedback, we need to combine different features from different images, so that a new query can be generated and used to compare with database images. Based on the feedback images selected by the user, the system will assign different weights for different features. So we need to consider an approach to combine the features using these weights. We have several methods to assign weights to different features when they are combined together. The easiest way is to set the values of the weights equal, which means we set all the weights to be 1. Another approach is to set them to different values depending on the needs of the user. An approach is to set them manually for different images. Obviously it is not efficient if we want to find many images from a large image set. Also sometimes the user could not set the weight properly if they do not have enough knowledge and experience.

[34] used a linear combination method to combine structure, colour, and texture

features extracted from images. They combine distances in the product space of structure, colour and texture for retrieval. Distance between a query image and a test image in the database in the structure and texture feature spaces are calculated using the  $L_2$  norm. Colour distance is calculated using histogram intersection measure, which is a variant of the  $L_1$  norm. Weights are associated with distances, which are used to assign the degree of importance attached to features extracted from different methodologies. Since a relatively larger or smaller value in a feature space biases the calculation of the weighted distance, they used the Gaussian normalization that puts equal emphasis on the distances in the each of the three feature spaces before taking a linear combination.

Feature combination is based on the concept that we combine different features and get a mean feature vector for a set of images. This feature vector is used to represent the attributes of the image set and to compare with the features of other images. We have another way to compare multiple images with one image. We calculate the distances on each feature of the query set and database image. Then combine these distances to get the final result. [88] discussed different methods of how to merge the partial results to form the final result list. Several combination methods such as linear combination and non-linear combination are discussed and compared in the paper.

**Linear distance combination** is a common method used in information retrieval. Suppose we have the partial retrieval result set  $R_l$ :

$$R_l = \langle i_1, dist_{l1} \rangle, \dots, \langle i_j, dist_{lj} \rangle, \dots, \langle i_{nl}, dist_{lnl} \rangle$$

Since different features may have different value range, combining two values directly is fraught with difficulties. We need to normalize the distances by using the following function:

$$f(x) = \frac{x - min}{max - min + 1}$$



where  $x$  is the given distance value and  $[min, max]$  is the range of the distance to be represented. Then the linear result combination to calculate the final distance for each image  $i_j$  is defined as follows:

$$dist_j = \sum_{l=1}^k w_l * dist_{lj}$$

where  $w_l$  is defined by the user, as the weight for the corresponding distance. We can then obtain the final distances for all the images and sort it to find the final solution.

There are several methods that can be considered under **non-linear combination**.

**Non-linear Distance Summation:** This is similar to linear distance combination.

But it uses the power function of the distance is the formula to calculate the distance:

$$dist'_j = \sum_{l=1}^k w_l * dist_{lj}^t$$

where  $t$  is an integer greater than 1 (usually  $t = 2$ ). In this way,  $w_l$  would give more contribution than in the linear combination method.

**Rank Summation:** To make the distribution of the result even, we can use other equivalent criteria such as rank of distance instead of the distance  $dist_{lj}$ . Then we could use linear combination to the new criteria to generate the final result. In this situation, we have the following functions:

$$R'_l = \langle i_1, r_{l1} \rangle, \dots, \langle i_j, r_{lj} \rangle, \dots, \langle i_{nl}, r_{lnl} \rangle$$

$$dist_j = \sum_{l=1}^k w_l * r_{lj}$$

**Rank Multi-Merge:** This method is similar to Rank Summation. But after getting  $R'_l$ , we take the image with the highest rank in its remaining images repeatedly from  $R'_1$  to  $R'_k$ , and put it in the final result list, until the final set has  $n$  items.

**Rank Multi-Sort:** This is also similar to Rank Summation except that the image  $i_j$ 's final rank value is calculated by

$$d_j = \min\{r_{lj} | l \in [1, k]\}$$

As discussed in the sections above, in some CBIR systems which adopt high-level features such as Region-Based CBIR and Shape-Based CBIR, users can specify regions or shapes they are interested in. This eliminates the side effect of irrelevant areas. But the system still could not tell irrelevant features from the interested features. One query image usually contains both positive features and negative features. The system does not know how to assign the weights for these features. Relevance feedback can improve the performance of CBIR Systems according to users' expectations. By specifying the positive groups and negative groups, the system can identify which features are more important and assign high weights for them. But it needs more iterations and more computing so the speed could be very slow. It also needs a human being to interact with it and may not be considered very 'smart'. In order to overcome the drawbacks of these approaches, Query By Multiple Images (QBMI) is proposed. It will be introduced in the following section.

## 2.5 Query by multiple images (QBMI)

In the past few years, single-query based image retrieval has been a popular query approach in Image Searching and is still widely used in research prototypes and commercial products. Most image retrieval systems support the single-query based image

retrieval only. In the single-query based IR, a database of images is searched to find images similar to a given query image. However, there is a gap between visual features used and semantic information. It has been proved that such a paradigm could not realistically lead to scalable, satisfactory query performance. [7] argued that query-by-one-example is not good enough to get a scalable and satisfactory query performance, and it might be desirable to query an image database using more than one query images for detailed knowledge representation. An advantage of the multi-query based IR is that it overcomes the limitation on the specification of image content using a single-query model.

In QBMI, more than one image is employed in the query. By comparing the similarities and differences between different features of the query images, we can tell which features should be assigned higher weight. And since it does not require so many iterations, the performance speed is higher than the relevance feedback approach. This section gives an introduction to QBMI and some popular approaches in this area.

### 2.5.1 Approaches in QBMI

QBMI is a compromise of Single Image QBE and Relevance feedback approach as it overcomes the drawbacks of those two approaches. Many CBIR systems have been proposed and implemented in QBMI.

[7] examined the limitations of the single-query based image retrieval approach and showed that the approach inevitably leads to poor performance. They used the TSVQ clustering algorithm and clustered a small image collection into 14 different categories including waves, tiger, clouds, flowers and so on based on the images' perceptual features, and showed that these image clusters are not coherent to the semantic categories of the images. Although some image categories are well separated from the others in the input space formed by the perceptual features, most of the categories

are co-located in more than one cluster. For certain query concepts of well isolation (for example, fireworks and tools), the Single-Query Based IR can produce acceptable results. But for complicated concepts (bears, tigers, etc.), querying with one example can lead to high rates of false positives. So for a query-concept that is mixed with others in a number of clusters, single-query based image retrieval lacks information to clearly identify the target query-concept. So it can not achieve satisfactory query results.

[37] listed the drawbacks of traditional IR technologies and explained why multi-query based IR is more effective. The cluster hypothesis [12](p. 45) states that “closely associated documents tend to be relevant to the same requests.” Most existing CBIR approaches are based on the assumption that relevant images are physically near the query image in some feature space. This is the basis of the cluster hypothesis mentioned above. Therefore retrieval can be performed by getting the images in the neighborhood of the given query in the visual feature space. However, not all the images with the same semantics will be close to each other in the visual space. Semantically related images are often scattered across several visual clusters. Figure 2.7 shows an example to explain this.

The red rose and the white rose are quite different in their colour histogram. Therefore there is not a single cluster for the semantic term “flowers”. Actually there are many clusters in the visual space. The following figure shows such an example. If there are different colour of flowers in the image database, they might form several clusters naturally in the visual space. Each cluster contains several images with similar visual features (colour), as shown in Figure 2.8.

Multi-query image retrieval has been proven to be very effective in providing feedback for enhancing single-query image retrieval results.

In some multi-example image retrieval systems, image retrieval is carried out using

Figure 2.7: Semantically related images are usually different in visual features [37]

Figure 2.8: Semantically related images are scattered in several visual clusters [37]

a single image example, and then the relevance of the output retrieved images is scored by users for training a classification model, say neural network, Bayesian model and so on, to modify the similarity measure to match users' expectations. Then the model is used to perform the retrieval.

Figure 2.9: Retrieval by query center of multiple queries may achieve different effects when the queries are located in one or more than one cluster [37]

Traditional CBIR approaches often assume that there is only one cluster. They tried to find the “best” representative of the user’s intention in the visual space and get images in the neighborhood of this optimistic query. But if there exists more than one cluster, as shown in Figure 2.8, it does not work well. It is unrealistic to cover all the scattered clusters in a region with a small size.

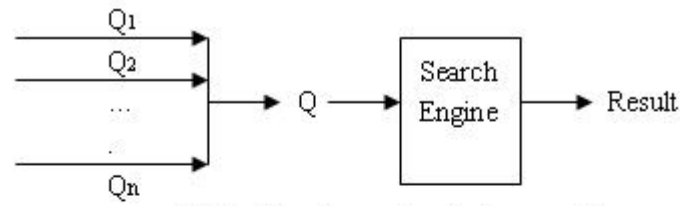
Some traditional CBIR feedback approaches try to move the query center by linear combination of the positive feedback images and try to adjust the neighborhood

shape by relevance feedback. This approach may also not be effective if more than one cluster exists. Figure 2.9 shows such an example. In (a), all positive queries happen to be located in one cluster and adjusting the query center may improve retrieval effectiveness. This is the theoretical foundation of traditional relevance feedback technology in CBIR. However, this will not work when the feedback queries are located in several clusters. In (b), if the feedback queries are located in several clusters, the query center may not be in any cluster. This results in the neighborhood of the query center containing nothing the user wanted.

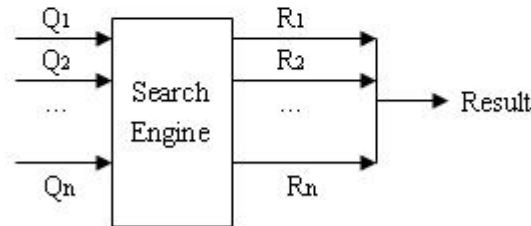
This CBIR search problem occurs because many CBIR approaches, no matter what feedback strategies, what visual features, and what distance formulations are used, all tend to find a single query center to perform retrieval task. This single query center does not sufficiently utilize the information contained in several feedback queries. In order to solve this problem, approaches using multiple queries have been developed.

In relevance feedback, after each iteration of retrieval, we need to combine the feedback images and use them to compare with database images. Similar to Relevance Feedback approaches, we will need to combine the queries at the first step to compare with database images. Two approaches have generally been used to capture an information need more precisely by a diversity of queries[4]. The first one is to combine the queries before searching, and submit the synthetic query to the search engine to get the result after that. The other one issues the searches individually and then merges the results of each query afterwards into a synthetic list. Figure 2.10 illustrates the basic procedure of these two approaches.

[37] proposed a multi-query retrieval technique based on the second approach. It can return semantically related images in different visual clusters by merging the result sets of multiple queries. For a given retrieval task, the user may pick different queries, which are all semantically related to the images the user desires. These queries will



(a) Combine the queries before searching



(b) Combine query results after searching

Figure 2.10: Two strategies for multi-query based IR

generate different retrieval results by the same CBIR system. These different result lists can be thought of as different viewpoints regarding the retrieval task in user's mind. The retrieval results in these two viewpoints have the potential to be merged into a better synthetic channel[22]. Merging the channels of multiple queries has two advantages. First, merging the channels of queries in one cluster, called intraccluster merging, can help filter some irrelevant images in the list, because the likelihood that an irrelevant image is similar to all the given queries will tend to be very low, although it may be quite similar to some query. This effect can improve retrieval precision. Second, merging the channels of queries in different clusters, called inter-cluster merging, can help to insert images from different clusters into the synthetic channel. Therefore images in different clusters will get the chance to appear in the synthetic list fairly. This effect can improve retrieval recall. Consequently, both retrieval precision and recall can be improved if we merge the channels of multiple queries. Channels used in this approach are formed by the original colour image (C+) together with the grayscale image (B+) and both the colour negative (C-) and the grayscale negative (B-). In the



merging part, an intuitive but effective merge strategy called mid-rank merge has been used [22]. All channels are treated as having the same importance, which means they have the same weight.

In multiple query image retrieval, each image has several types of features including colour, texture and shape. The question arises as to how to use those features from multiple images simultaneously [77]. Two approaches have been proposed to use the features. One is to combine different features from different query images. The other is to use the same feature from different query images.

An example of Multi-Example Based IR is multi-component image retrieval in which the components are features from multiple images. Assuming that we have two query images, one image may contain a texture of interest and the other image may contain colours of interest. We can use the texture features of the first image and the colour information of the second image to retrieve images with similar texture to that of the first image and similar colour information similar to that of the second image.

Figure 2.11 illustrates the general procedure of component based image retrieval.

[77] proposed an image retrieval algorithm which uses multiple query images. The proposed algorithm is based on multi-histogram intersection techniques. They calculate the colour histogram and texture histogram for each query image. Then the query image and the database images could be compared and the similarity calculated by multi-histogram intersection.

In 2002, Munehiro et al.[58] proposed an image retrieval system which uses a new concept called Query-by-Group. This is an extension of Query-by-Examples. The major difference is that Query-by-Example handles the images individually, but Query-by-Group considers a group of images as the basic unit of query. When doing the query, the user could select any number of image groups as either positive clusters or negative clusters. The groups can overlap with each other, allowing images to belong to multiple

Figure 2.11: Example of Multi-Query Based IR [77]

groups. The system will retrieve images similar to the positive group images and avoid images similar to negative group images.

[88] developed a geographical data retrieval system which supports multi-example queries. The users can assign different weights to each query image based on their own interests. We can get the difference between each query image and database images. The paper focused on how to merge the partial results to form the final result list. Several combination methods such as linear combination and non-linear combination are discussed and compared in the paper.

## 2.6 Formulation of the problem

While a considerable amount of work has been done in constructing queries for image retrieval systems, the literature indicates that multiple image queries has not been

---

studied extensively. The problem of how to combine the information provided by multiple images in the context of image retrieval queries needs further study. This problem should be addressed from the viewpoint of the extent to which the information provided is able to represent user intention. In essence the question relates to the correlation between the information content of multiple image query and human perception and expectation.

This thesis will explore the formulation of the problem in information theoretic terms and use the model derived to compute weights for the visual features employed in multiple image query retrieval.

## Chapter 3

# Perceived Similarity and Visual Descriptions in Image Retrieval

This chapter will investigate the human performance in image retrieval and evaluate performance of some visual features of images for image retrieval against perceived similarity of images. A Web-based experiment system is designed and implemented to collect data on how humans perceive the similarity of images. Then the similar images judged by humans are used as ground truth sets for image retrieval using MPEG-7 visual descriptors to evaluate their performance. The effect of weights in combining multiple visual descriptors for image retrieval is investigated. The experimental results are presented and analyzed.

### 3.1 Introduction

The goal of image retrieval systems is to find a set of relevant images which are similar to the query image that the user sets as an example. Depending on the intention of the user, similarity might be described by objects, colours, shapes, semantic aspects, or any combination of the above. In practical applications, it is not explicitly expressed.

In the ideal case, relevant features of images that a human uses to search for similar images are extracted and a similarity measure of these features is used to rank the images. However, how humans perceive the similar images is not well understood. Current technology of content-based image retrieval based on some visual features of images that can be extracted with computer algorithms. It is observed that current retrieval schemes based on low-level visual features can perform well for some categories of images while poorly for others.

Colour, in particular, has proved to be one of the most efficient features for the calculation of image similarity, since an object's colour is independent of the viewing position or viewing distance. Other features such as texture, edges and shapes are also used very often. All these approaches are based on physical, low-level features. But in the end the similarity of a given image with a target image will always be judged by a human observer. Therefore, the goal of image retrieval is to construct retrieval algorithms which are based on some of the low-level features of images and the retrieved images can better match the images judged by humans as similar images.

Research into content-based image retrieval has progressed in the last decade to the point where multi-image query is being proposed as a better alternative to the "single image query plus relevance feedback" paradigm. However, this new thinking presupposes that there is a "picture alphabet" that humans can use to provide a "picture description" of another picture. This hypothesis has not been tested in content-based image retrieval tasks. It is not clear how humans use multiple pictures to describe another picture or set of pictures they have in mind, and how different descriptors contribute to the image retrieval. Furthermore, to evaluate a proposed image retrieval system, we need to have a quantitative and objective approach to measure the resulting image metric by comparing it with measurements of perceived image similarity. A set of psychological experiments designed to provide answers to some of the questions.

The experimental results and their analysis are presented in the following sections.

## 3.2 Psychological Experiments on Perceived Similarity of Images

A web-based system is designed and implemented for the experiment. This allows the experimental set to be computer platform agnostic. The windows will be normalized to work effectively on the minimum screen size available in the laboratories where the experiment will be conducted.

The participant, or subject, will be presented with a window partitioned appropriately for each experiment. The subject will be shown one example image at the upper part of the window and  $n$  candidate images at the lower part of the window. They are required to select  $k$  images from the candidate images which they think are similar to the example image by clicking on them. One thing we need to consider is what is the optimum value of the number  $n$  and  $k$ ? After a discussion with the consultant from the School of Psychology, we decide to use 20 as the number of candidate images and use 5 as the number of selected images in our experiment, which means the subjects need to pick up 5 images from a list of 20 images in each experiment. The selected values for  $n$  and  $k$  are believed to be adequate for human subjects to view the images on the screen and conduct the experiment. There are 12 people who participated in these experiments. As an example, the ground truth set of “bush” is shown in Figure 3.3.

The experiment repeats 10 times for each subject. Each time an example image and a set of candidate images are presented to the subject. The sample image and the candidates are selected from a categorized image database. The categories include:

- Beach

- Bushes
- Cars
- Flowers
- Horses
- Mountains
- Opera House
- Party
- Ships in the ocean
- Sunset

Each set of candidate images contains 20 images from one category, which may have various visual features. For example, the 20 images in the category of “car” may contain cars with different shapes, different colours or different sizes. The human subjects are required to choose 5 images from them that are believed to be similar to the example image. Figure 3.1 shows the user interface of the experiment system.

The subjects produce their results in 2 steps. In Step 1, the subjects are required to select 5 images which are similar to the example. In Step 2, the subjects can re-order the 5 selected images so that the image on the left is the most similar to the example image and the image on the right is the least similar to the example image.

Figure 3.2 shows an example of the results for the category “bush”. The image order is determined by the number of the subjects who have selected the image as a similar image to the example image. The most selected image by all subjects is displayed first, which is considered as a similar image by majority voting. Therefore,

Figure 3.1: User Interface of the Experiment System



Figure 3.2: Experimental Results: “Bushes”

the displayed images in the order of voting are collectively considered as the retrieval results for the example image as the query image by human beings.

In order to compare the results to the results by computer retrieval systems and evaluate the visual descriptors in the following sections, a set of relevant images to the query image is determined by the majority votes. The rest are considered as irrelevant images. The relevant images determined by human subjects are used as ground truth sets.

### **3.3 Perceived Similarity and Similarity Measured by MPEG-7 Descriptors**

Generally speaking, people search for similar images based on the perceived visual content of images. It is called perceived similarity. However, visual features perceived and used by people to judge the similarity are still unknown today. Although some

Figure 3.3: Ground Truth Set: “Bush”

low-level visual descriptors are used in CBIR systems to represent the features of the images such as colour, texture and shape as specified in the MPEG-7 standard, these descriptors calculated by computers may not be the same as the features used by human beings. The ultimate goal of the CBIR system design is to rank the target images so that they have similar rankings to human subjects.

Common visual features used by machines, such as MPEG-7 descriptors, are designed to describe visual characteristics of the image content to a certain degree. The effectiveness of these descriptors is to be evaluated against the results by human subjects. The ground truth sets for all queries are created as described in the previous section. Similar images in the ground truth sets are ranked according to perceived similarity. In this section, the similar images ranked according to visual similarity based on visual descriptors are evaluated against the ground truth images. 4 MPEG-7 visual descriptors: CLD, CSD, EHD and HTD are used individually and in combinations as the features for similarity measurement. Precision and recall graphs are used as the performance indicator.

The results for various categories are presented and discussed in the following section.

### **3.3.1 Bushes**

Figure 3.4 shows the curves for the category “Bushes”. The performance of descriptors CSD and EHD are much better than that of the other two descriptors. Therefore, features described by CSD and EHD can be used to rank the images in this category. They are considered more important than others for this category.

Figure 3.5 - 3.8 show the retrieval results using the 4 descriptors. The ground truth has been shown in Figure 3.3

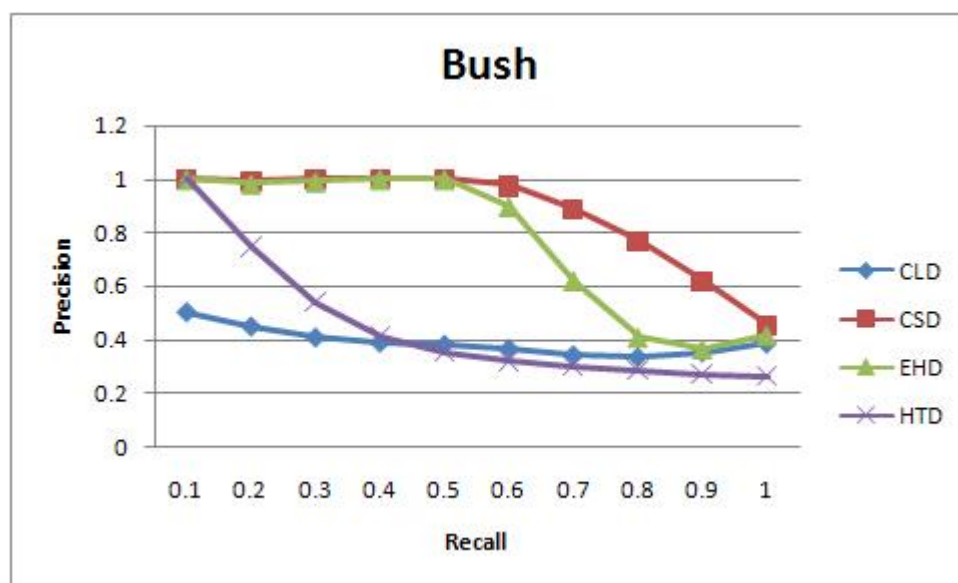


Figure 3.4: Precision-Recall Curves - “Bush”.

Figure 3.5: Retrieval Results by CLD for “Bush”

Figure 3.6: Retrieval Results by CSD for “Bush”

Figure 3.7: Retrieval Results by EHD for “Bush”

Figure 3.8: Retrieval Results by HTD for “Bush”

### 3.3.2 Party

Figure 3.9 shows the performances for “Party”. None of the the four descriptors exhibit good performance for this category. No descriptor could describe images well from this category. It is difficult for humans to judge the similarity in this case, as shown in Figure 3.10. The retrieval results by human subjects have diverse selections of images with perceived similarity, which results in a larger number of ground truth images as shown in Figure 3.11. The retrieval results by the 4 descriptors are shown Figures 3.12 - 3.15. They not only result in poor performances but also very different rankings. This query is considered as a difficult query for both humans and machines.

### 3.3.3 Other categories

Results for other categories are presented in Figures 3.16 to 3.23, which show that different features are believed to have been used for different queries and different

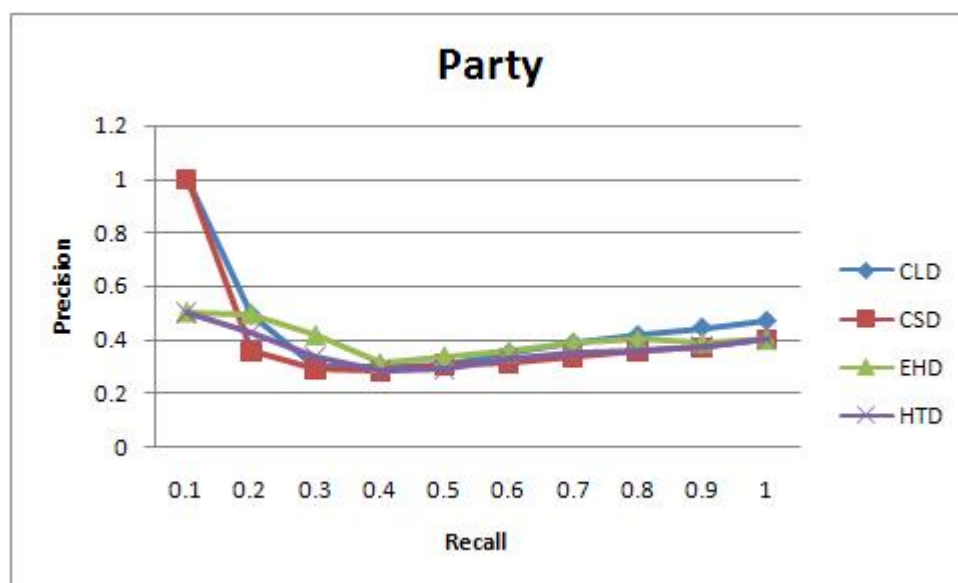


Figure 3.9: Precision-Recall Curves - "Party".

Figure 3.10: Retrieval Results by Human Subjects for "Party"

Figure 3.11: Ground Truth Set - Parties



Figure 3.12: Retrieval Results by CLD for “Party”

Figure 3.13: Retrieval Results by CSD for “Party”

Figure 3.14: Retrieval Results by EHD for “Party”

Figure 3.15: Retrieval Results by HTD for “Party”

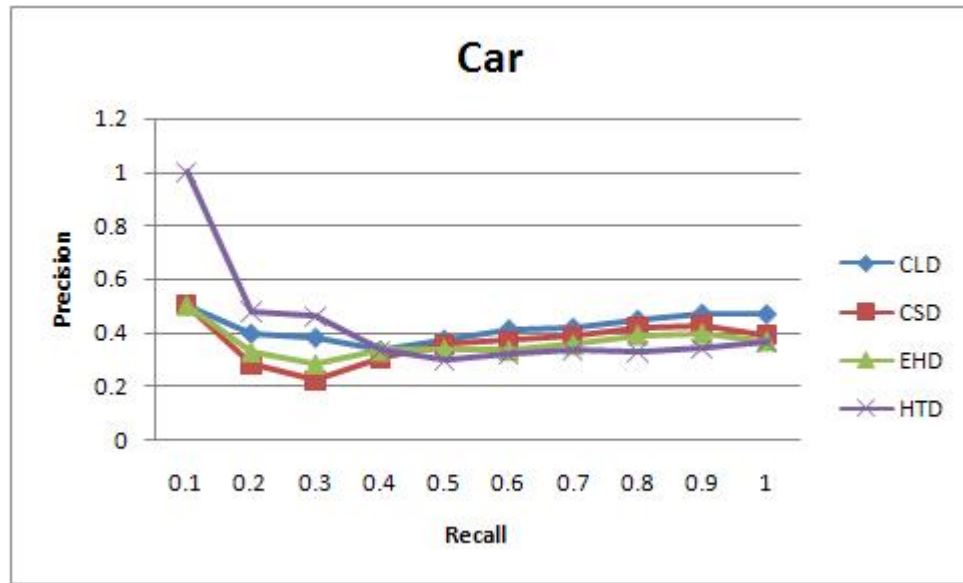


Figure 3.16: Precision-Recall Curves - Car.

descriptors describe some queries well while others poorly.

### 3.4 Further Analysis of Experimental Results

To further analyze the data, the mean values and standard deviations of dissimilarity or distances of features are calculated for 4 different descriptors for all the categories. The results are shown in the following tables: Table 3.1, 3.2, 3.3 and 3.4.

As shown in Table 3.1, the standard deviation of the CLD distances of the category “Flowers” is the greatest and that of the category “Party” is the smallest. The average distance of the category “Horse” is the greatest and the average distance of the category “Opera House” is the smallest.

As shown in Table 3.2, the standard deviation of the CSD distances in category “Ship in the Ocean” is the greatest and that of the category “Sunset” is the smallest. The average distance of the category “Car” is the greatest and the average distance of the category “Sunset” is the smallest.

In Table 3.3, the standard deviation of the EHD distances of the category “Sunset”

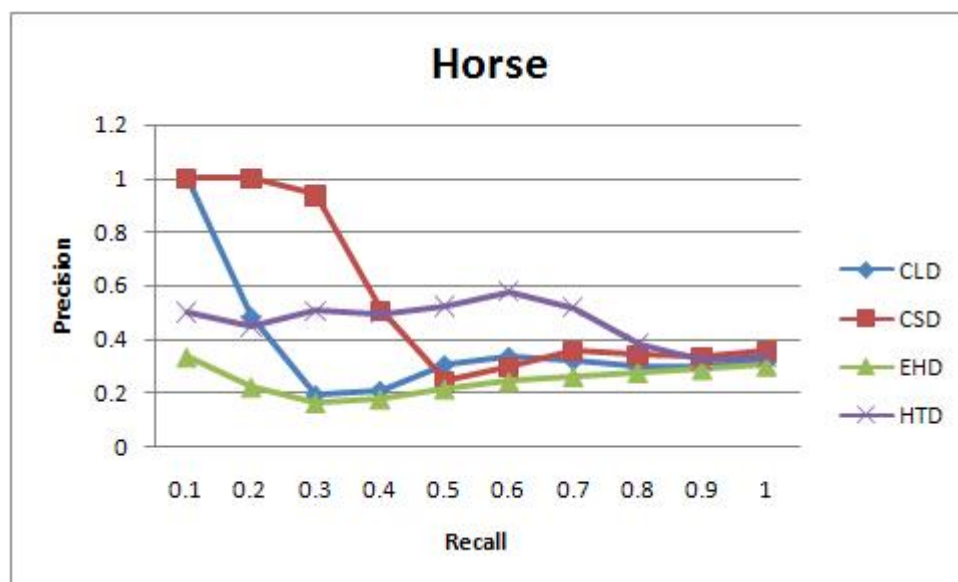


Figure 3.17: Precision-Recall Curves - Horse.

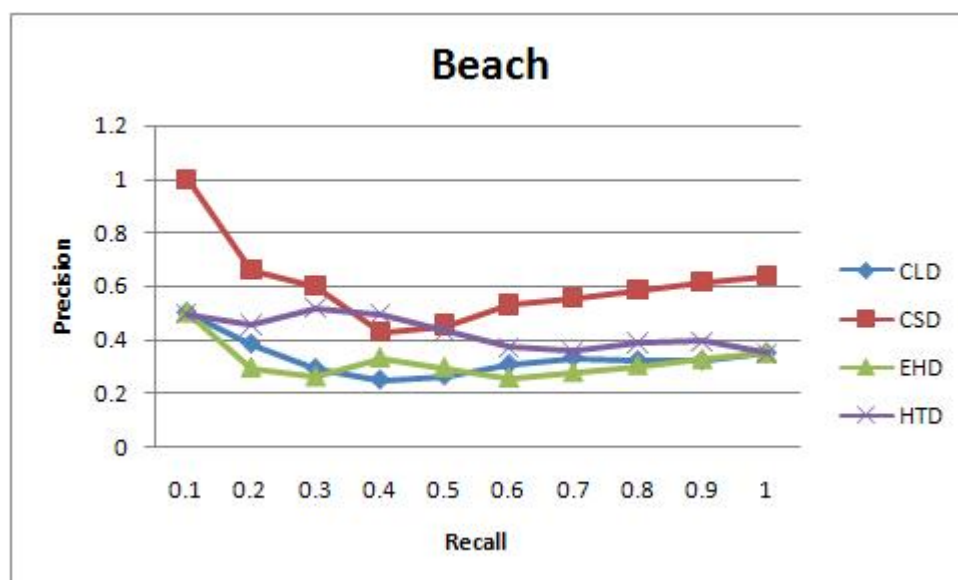


Figure 3.18: Precision-Recall Curves - Beach.

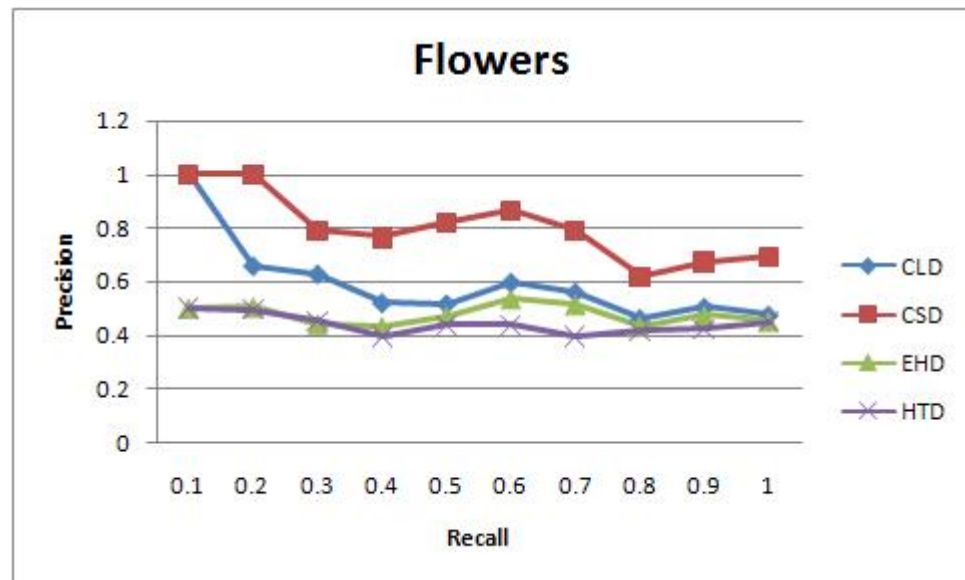


Figure 3.19: Precision-Recall Curves - Flower.

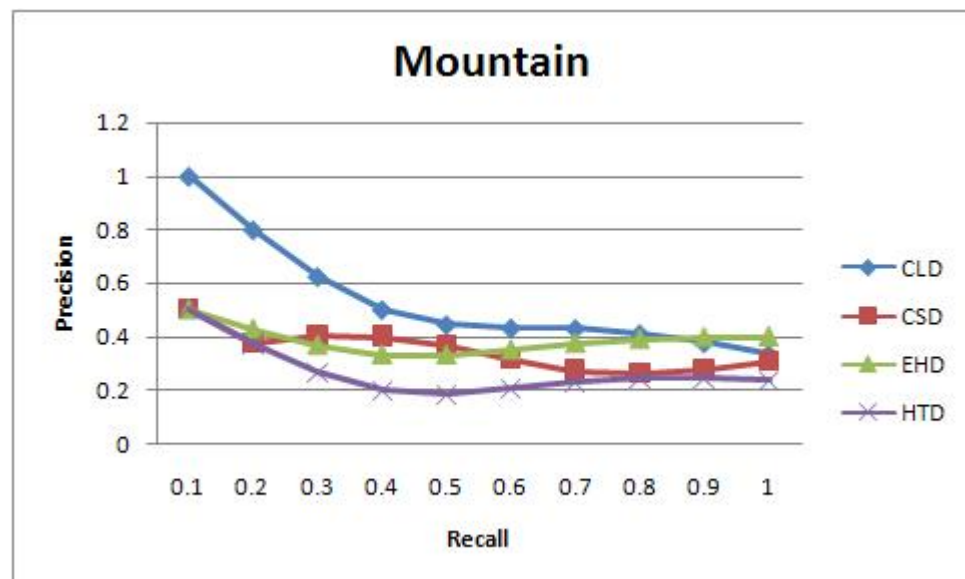


Figure 3.20: Precision-Recall Curves - Mountain.



Figure 3.21: Precision-Recall Curves - Opera House.

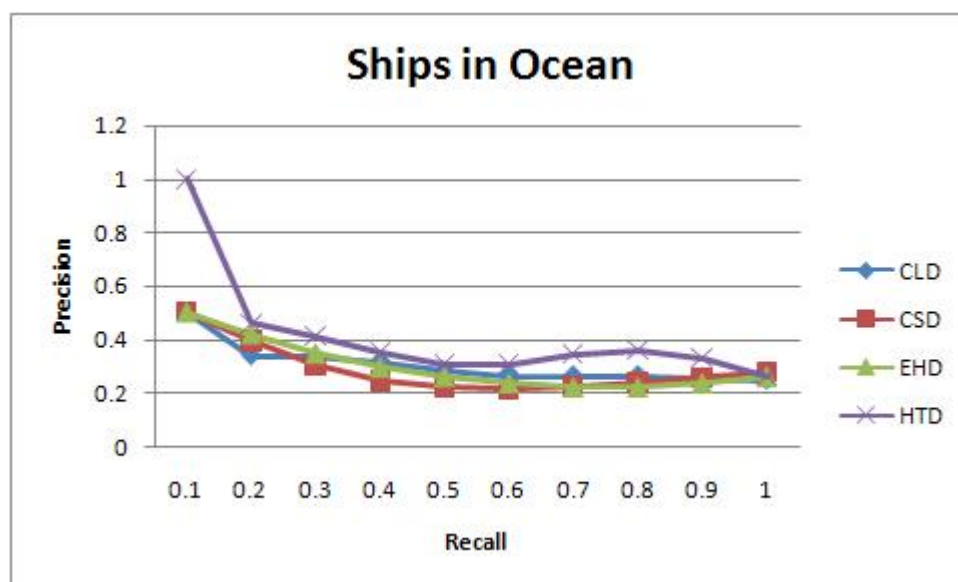


Figure 3.22: Precision-Recall Curves - Ship in Ocean.

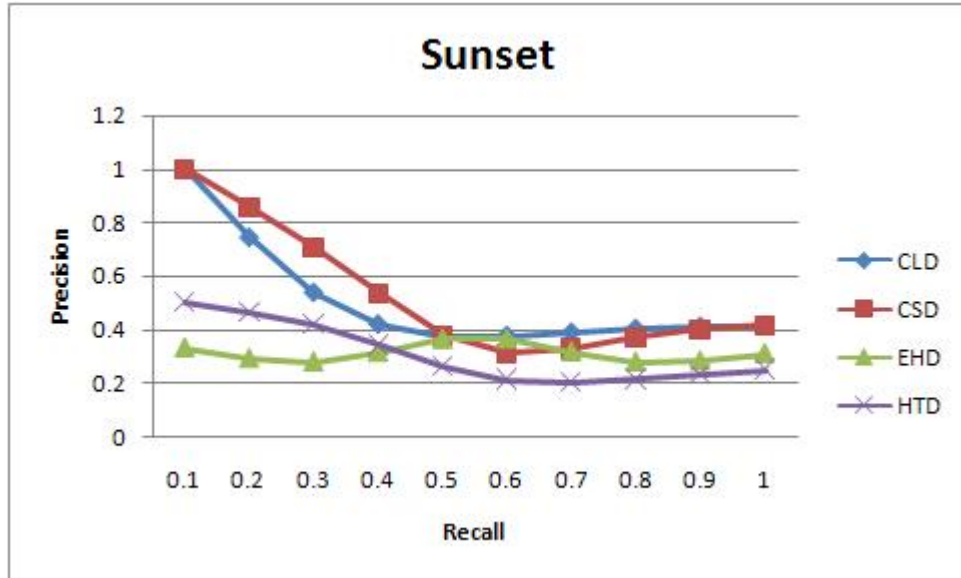


Figure 3.23: Precision-Recall Curves - Sunset.

	Mean Value	Standard Deviation
beach	0.3204	0.1383
bush	0.2718	0.1324
car	0.3020	0.1499
flowers	0.3274	0.1593
horses	0.3653	0.1433
mountains	0.3438	0.1339
Opera House	0.2685	0.1347
Party	0.2759	0.1069
Ship in Ocean	0.2996	0.1357
Sunset	0.3461	0.1479

Table 3.1: Mean values and standard deviations for normalized distances on CLD feature space

	Mean Value	Standard Deviation
beach	0.2100	0.0875
bush	0.3183	0.1320
car	0.3398	0.1394
flowers	0.243	0.1280
horses	0.3252	0.1295
mountains	0.2635	0.1023
Opera House	0.2581	0.1138
Party	0.3182	0.1106
Ship in Ocean	0.2821	0.1401
Sunset	0.1724	0.0824

Table 3.2: Mean values and standard deviations for normalized distances on CSD feature space

	Mean Value	Standard Deviation
beach	0.2771	0.1274
bush	0.1901	0.1605
car	0.2874	0.1240
flowers	0.2903	0.1266
horses	0.2863	0.1181
mountains	0.3040	0.1399
Opera House	0.2996	0.1364
Party	0.2611	0.1110
Ship in Ocean	0.2736	0.1444
Sunset	0.3040	0.1668

Table 3.3: Mean values and standard deviations for normalized distances on EHD feature space



	Mean Value	Standard Deviation
beach	0.3205	0.0558
bush	0.3305	0.0681
car	0.3201	0.0465
flowers	0.3411	0.0611
horses	0.3329	0.0580
mountains	0.3277	0.0560
Opera House	0.3450	0.0616
Party	0.3188	0.0543
Ship in Ocean	0.3281	0.0684
Sunset	0.3483	0.0777

Table 3.4: Mean values and standard deviations for normalized distances on HTD feature space

is the greatest and that of the category “Party” is the smallest. The average distance of the category “Sunset” and “Mountain” is the greatest and the average distance of the category “Bush” is the smallest.

As shown in Table 3.4, the standard deviation of the HTD distances of the category “Sunset” is the greatest and that of the category “Car” is the smallest. The average distance of the category “Sunset” is the greatest and the average distance of the category “Party” is the smallest.

### 3.5 Effect of Weights in Combining Visual Descriptors

In a CBIR system using multiple features, it is important to understand the roles that individual descriptors play in various queries and how they would impact on the overall retrieval results.

As presented in the previous section that performances of individual descriptors are limited. This section will investigate the effects of weights of individual descriptors on the retrieval performance. To get higher performance, multiple features are combined.

It is expected that a proper combination of features used in retrieval systems could result in improved performances. The general idea is to assign higher weights to a feature that is more important to the query. But the question is how important needs to be assessed and relevant to the query.

Experiments are designed to evaluate the effects of weight assignments based on the performance of individual descriptors.

To get a higher system performance, methods of combining multiple features are proposed. We tested the system performance by combining all the 4 MPEG-7 descriptors used when generating the curves. There are many methods to combine different descriptors. One is to use equal weights, this is to assume that different descriptors takes the same importance during the searching. But in most cases, they don't play the same roles. We need to assign weights to the descriptors. In the image retrieval based on multiple queries, the multiple query images are used to derive proper weights for each feature.

In our experiment, we assign higher weights to descriptors which are considered more important for the category, and lower weights to the descriptors which do not play important roles during the retrieval. The system performance is also evaluated using precision and recall curves. These curves are compared with the curves generated by using single descriptors. Different weighting methods are tested and the result curves are shown in the following paragraphs.

The curves in Figure 3.4 show that the descriptors CSD and EHD are considered more important than CLD and HTD for the category "Bush", which means CSD and EHD contribute more than the other two descriptors. Based on the discussion above, if we combine the 4 descriptors together, the latter should be assigned more weights than the others. Since we need to calculate the distance between query and database images, we assign weights to the distances of different descriptors and use the combined

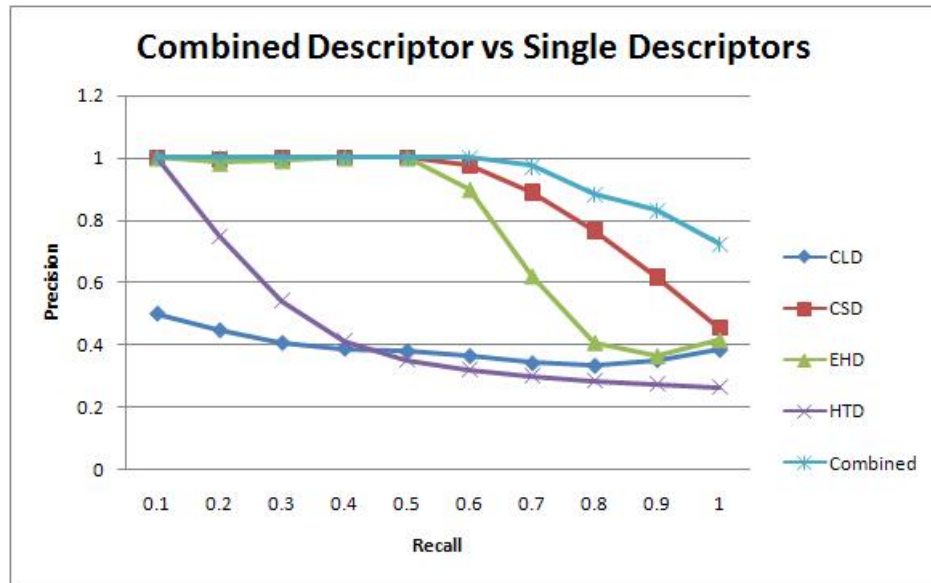


Figure 3.24: Combine the features by putting greater weights on more important features in category “Bush”.

distance to rank the images. We set the weights of CSD and EHD distances to 0.3 and the weights of CLD and HTD distances to 0.2, which makes the sum of all the weights be 1. Then we do the retrieval using the combined distances. The precision and recall curve of the new retrieval are compared with the curves of using single descriptors in Figure 3.24. As we can see from the figure, the retrieval using combined features has a higher performance than using any of the single descriptor.

This demonstrates that proper combination of descriptors can improve the performance. Figure 3.25 shows the retrieval result that can be compared to the ground truth set, as shown in Figure 3.3 to see the improvement of the retrieval results.

Another experiment has been conducted on the same category to evaluate the equal weight assignments. The precision and recall curves are compared in Figure 3.26. The curve of the retrieval using combined features is not the highest this time. Although it is better than using a single descriptor of CLD or HTD, it is not as good as using a single descriptor of CSD or EHD. This means using equal weights will not improve over the performance of a good descriptor.

Figure 3.25: Retrieval results with combined descriptors for “Bush”

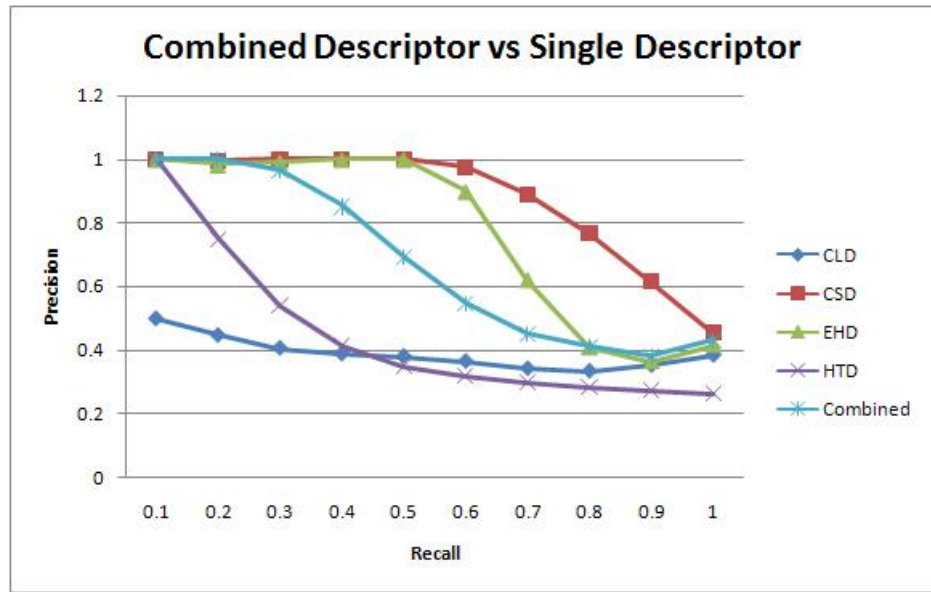


Figure 3.26: Combine the features by putting equal weights to the descriptors in category “Bush”.

One more case has been investigated. An experiment is designed to test the effects of weights that are assigned in the way opposite to that in the previous experiment. We set the weights of CSD and EHD to be 0.2 and the weights of CLD and HTD to be 0.3. This means we assign the higher weights to the less important features during the retrieval. The curves are shown in Figure 3.27. The result shows that if we combine the features with assigning higher weights to less important features, the retrieval performance may be even worse than using single descriptors. This also raises the question of how to design a good weighting method to assign proper weights to each of the descriptor.

The discussion above focuses on the situations where there are significant features to describe the images of the category. However, for some of the categories, there are no significant features to describe the images therein. For example, from the standard deviations of the distances calculated for each category, we can find out that the standard deviation value for category “Party” is very small for all the 4 descriptors. This indicates that no matter in which feature space, images in category “Party” are

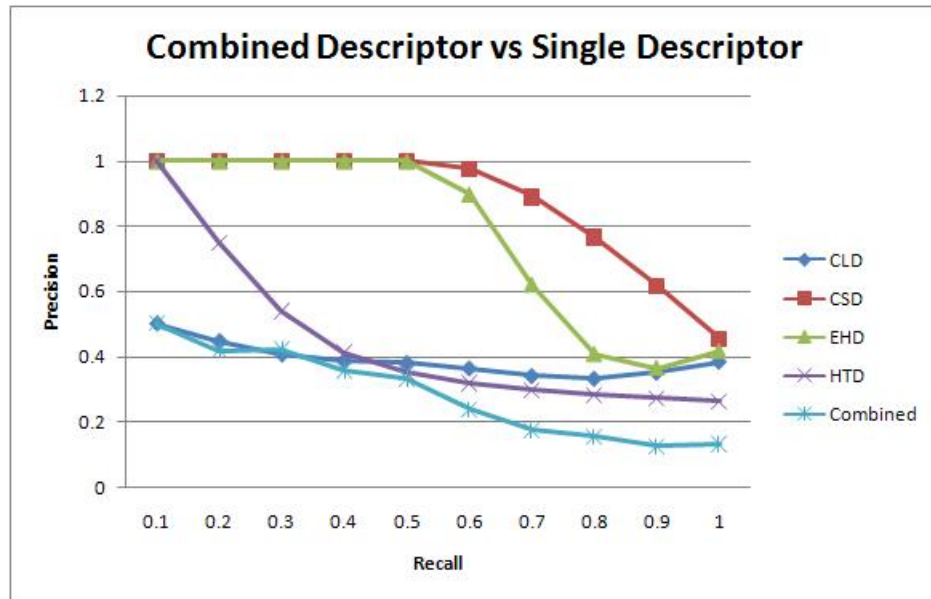


Figure 3.27: Combine the features by putting greater weights on less important features in category “Bush”.

very similar to each other. This causes the picked up images by the human subject are very different. In this case, using any single descriptor can not give good performance.

We combine the descriptors to do the retrieval in the category “Party”, and compare the result with retrieval using single descriptors. As shown in the previous results, we could not find any descriptors which are more significant than the others. Thus we assign equal weights to all the descriptors. Since 4 descriptors are used, we set all weights to be 0.25. The curves are shown in Figure 3.28. As we can see, using combined features in the category “Party” does not give a good result as well. So we draw the conclusion that in categories that do not have significant features, combining the features does not provide any improvement to the system performance.

We also try to assign the same weights as we used in previous category to features in this category. Firstly, we set the weights for CSD and EHD to be 0.3, while the weights for CLD and HTD are set to 0.2, then in the opposite way, weights for CLD and HTD are set to be 0.3 and weights of CSD and EHD are set to 0.2. The system performance of using combined features are compared with that of using single descriptors in Figures

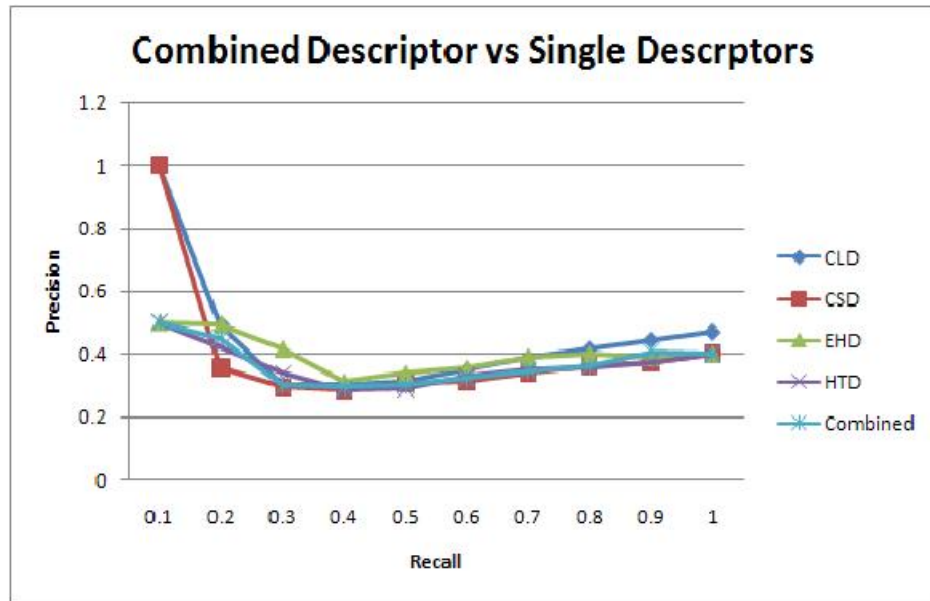


Figure 3.28: Combine the features by putting equal weights to all descriptors in category “Party”.

3.29 and 3.30.

As we can see from these curves, assigning different weights to different features does not help to improve the performance of the image retrieval system in this category. This also demonstrates that for categories without significant features, the performance of the retrieval system can hardly be improved by using weights on the features. We can also find out that although putting higher weights to more important features can improve the system performance to a certain degree, there is no fixed set of weights which is suitable for all situations. The weights of features changes from one category to another. To improve the performance of the image retrieval system, a weighting method which can assign different weights to the features based on categories is needed. With a single query image, we can hardly know the category information. But if we use multiple images as the query, the query set can reflect the attributes of images of their category much better than using a single query. Our proposed weighting method is based on multiple query images. We will introduce it in the next chapter.

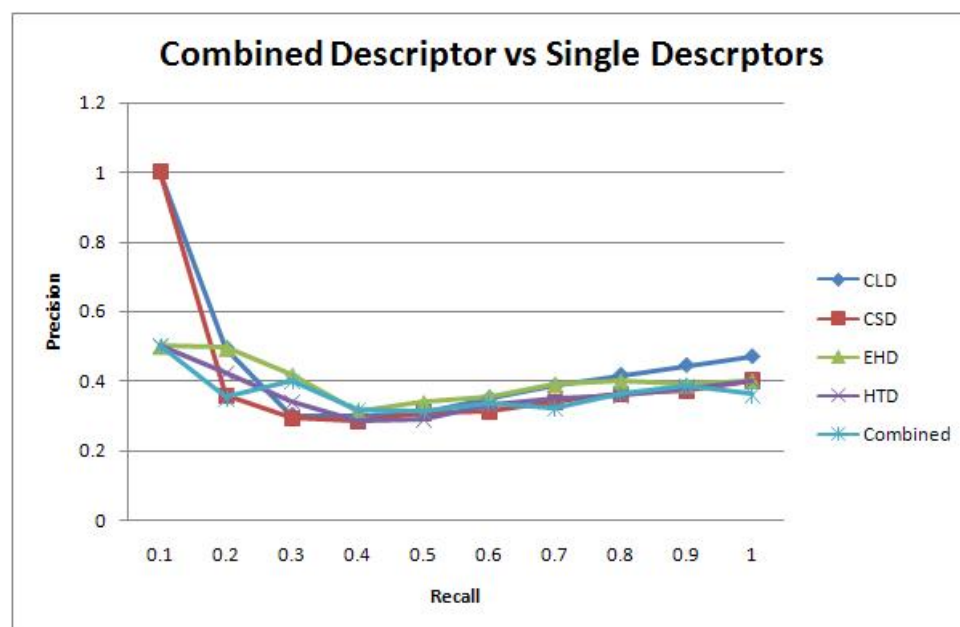


Figure 3.29: In category “Party”, weights of CSD and EHD are set to 0.3 and weights of CLD and HTD are set to 0.2

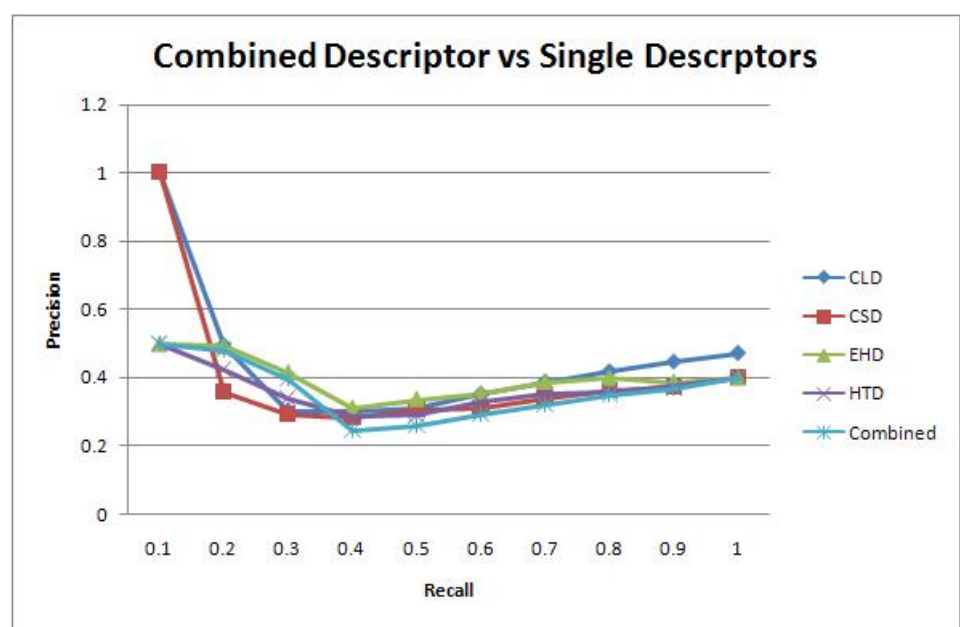


Figure 3.30: In category “Party”, weights of CSD and EHD are set to 0.2 and weights of CLD and HTD are set to 0.3



## 3.6 Summary and Conclusions

A series of psychological experiments has been conducted to collect data on how human subjects judge 'similarity'. Based on these experiment results, some common visual descriptors are evaluated against the results by human subjects. It is found that various descriptors play different roles in different queries and proper combinations of them can improve the performance. There are no fixed weight assignments for all queries or categories. The weights of different descriptors change from one category to another.

To improve the performance of image retrieval systems, we can combine the different features by assigning weights to the features that are derived from each individual query. Generally speaking, to get good performance, we should assign higher weights to more important features. There are many methods to assign the feature weights. Using multiple images as the query to derive the weights is one of the solutions. An approach of image retrieval which uses multiple queries to decide the weights is proposed in the following chapters.

# Chapter 4

## A New Weighting Method for QBMI

This chapter presents two basic interpretations of how users may relate a set of example images with the images they desire to search for in a Query By Multiple Images (examples) (QBMI) system. One is based on a *compositional* view that users utilize the objects contained in the example images to compose the desired images in their mind. The other is based on a *descriptive* view in which the example images describe a cluster of images to the user's mind. In the descriptive view, one of the key challenges in QBMI is to calculate the intra- and inter-weights of the features that are employed to describe the images. Inspired by the work of mutual information (MI) based feature selection and the concept of distance entropy, a new method is proposed to calculate inter-feature weights for QBMI.

### 4.1 Interpretations of QBMI

Let  $Q = \{q_1, q_2, \dots, q_n\}$  be  $n$  query images (examples) and  $\Omega$  be the image database that contains  $N$  images. The task of QBMI is to find  $K$  target images,  $T = t_1, t_2, \dots, t_K$ ,

that are most “relevant” to  $Q$ , where  $T \in \Omega$ . A common approach is to rank the  $N$  images in  $\Omega$  in terms of their “relevance” to the query set  $Q$  and select the  $K$  most “relevant” images. Therefore, the measurement of the “relevance” between  $Q$  and each image in  $\Omega$  is essential for QBMI. Ideally, the relevance measurement should capture the user’s intention to relate  $Q$  with the wanted images. In other words, the relevance measurement should reflect the user’s interpretation of the query set  $Q$ . Assume  $\tilde{I}$  is the image or a set of images wanted by the user. There are two basic views of interpreting the relationship between  $\tilde{I}$  and  $Q$ : compositional and descriptive

#### 4.1.1 Compositional View

In principle, an image is composed of a set of semantic objects. For instance, an image may consist of animals, flowers and cars in a background of a green field. Let  $O_p^i$  be the set of objects in  $q_i$  that are expected to appear in  $\tilde{I}$  and  $O_n^i$  be the set of objects in  $q_i$  that are optional to  $\tilde{I}$ ,  $q_i = O_p^i \cup O_n^i$ . The compositional view says that, ideally,

$$\tilde{I} = \bigcup_{i=1}^n O_p^i \quad (4.1)$$

where  $\cup$  is a composition operator. In other words, a user holding this view relates  $\tilde{I}$  with  $Q$  through the objects that appear in the query images and interest the user. For example, if a user wants to find a white ship in ocean, he/she may select one or more example images that contain ships and one or more examples that contain ocean as their query set. [88] took the compositional view and implemented an image retrieval system for geographic images. In their system, users may query for images with residential areas and grasslands by using example images from the category of residential areas together with images from the category of grasslands.

Compositional view based QBMI requires extraction or segmentation of objects and identification of the objects in which users are interested from the query images. In the

process of retrieval, the system needs to extract the objects from every database image so that objects from the query images can be compared with the objects contained in the database images. The performance of such a system highly relies on how well the objects can be segmented. It is well known that image segmentation is an ill-posed problem. Reliable segmentation algorithms that are applicable to a wide range of images are yet to be devised and none of the existing segmentation algorithms would give satisfactory object segmentation in many cases. Nevertheless, under certain restrictions such as the interested objects are dominant in each query images and/or users being interested in all objects in the query images, i.e.  $O_n^i = \phi, \forall i$ , compositional view based QBMI may not require explicit segmentation.

#### 4.1.2 Descriptive View

Instead of using the objects in  $Q$  to *compose* the image  $\tilde{I}$ , users may use  $Q$  to *describe*  $\tilde{I}$ . In this case,  $\tilde{I}$  often represents a cluster of images that users are interested in and each query image  $q_i$  may describe  $\tilde{I}$  from one or more perspectives. In this descriptive view, construction of the cluster  $\tilde{I}$  either explicitly or implicitly from the query set  $Q$  is required to measure the relevance between  $Q$  and images in  $\Omega$ .

Descriptive view based QBMI may be considered as a special case of relevance feedback (RF) in which only positive examples are available. All positive examples in RF are usually thought to describe a cluster of images that users are searching for. However, QBMI does not have the opportunity of iterative refinement through interactions as RF has.

#### 4.1.3 Discussion

While both compositional and descriptive views have their own roles in QBMI, most existing systems are based on a descriptive view. Our perceptual experiments have

also demonstrated that users tend to prefer descriptive view to composition view. In one of the experiments, subjects were given a set of images containing ships, ocean or both, and asked to select images to form a QBMI query to search for the image of “Ships in ocean”. Only 20% of the subjects chose images with only ships or ocean to form the query set (compositional view). 80% subjects chose images having both ships and ocean (descriptive view). This thesis focuses on the descriptive view based QBMI.

## 4.2 Descriptive View Based QBMI

In a descriptive view, we assume that each query image  $q_i \in Q$  describes  $\tilde{I}$  from one or more perspectives, i.e.  $q_i$  will reveal partial information about  $\tilde{I}$ . None of the query images alone will be able to fully describe  $\tilde{I}$ . “Relevance” between two images is measured as a distance between the two images in multiple feature spaces. Distances are calculated with respect to individual features and combined together based on how well the corresponding feature describes the image cluster formed by the query set  $Q$ . The smaller the combined distance, the higher the relevance of the two images.

Let  $f_i, f_2, \dots, f_F$  be a set of visual features to describe the images and feature,  $f_i, i = 1, 2, \dots, F$ , has  $F_i$  components and is represented as a vector,  $f_i = (f_{i,1}, f_{i,2}, \dots, f_{i,F_i})$ . The major challenge in the descriptive QBMI is to compute from the the query set  $Q$  the importance of every feature,  $f_i$  and its components,  $f_{i,j}$ , with respect to  $\tilde{I}$ .

Let  $u_i = (u_{i1}, u_{i2}, \dots, u_{iF_i})$  be the weights of the components of the  $i$ 'th feature, where  $\sum_{j=1}^{F_i} u_{ij} = 1, i = 1, 2, \dots, F$ , and  $w = (w_1, w_2, \dots, w_F)$  represents the weights of the  $F$  features, where  $\sum_{i=1}^F w_i = 1$ . The higher the weight, the more important the corresponding feature or feature component.  $u_i$  and  $w$  are often referred to respectively as intra-feature and inter-feature weights or, briefly, intra- and inter-weights. The relevance of the  $r$ 'th image,  $I_r$ , in the database  $\Omega$  and the query set  $Q$  is calculated as

a combination of the distances between the two images in the  $F$  feature spaces.

**Step One** Calculate the distances between  $I_r$  and  $q_i, i = 1, 2, \dots, n$  with respect to feature  $f_j, j = 1, 2, \dots, F$

$$d_{ir}^j = g^j(I_r, q_i, u_i), \quad (4.2)$$

where  $g^j(\cdot)$  is a metric to measure the distance or similarity between  $I_r$  and  $q_i$  in the feature space of  $f_j$ . Depending on the nature of  $f_j$ , generalized Euclidean distance [70] or weighted block distance may be adopted.

**Step Two** Calculate the distance between  $I_r$  and the query set  $Q$  as the distance to  $\tilde{I}$

$$d_r^j = \Lambda(d_{1r}^j, d_{2r}^j, \dots, d_{nr}^j), j = 1, 2, \dots, F \quad (4.3)$$

where  $\Lambda(\cdot)$  is a function that aggregates the distances to individual query images as a distance to the query set  $Q$ .

**Step Three** Combine distances with respect to individual features according to their importance,  $w_j$ , to form the overall distance  $d_r$  between  $I_r$  and  $Q$ .

$$d_r = \sum_{j=1}^F w_j * h(d_r^j), \quad (4.4)$$

where  $h(\cdot)$  is linear or non-linear function. A simple linear combination is to set  $h(d_r^j) = d_r^j$ .

In retrieval, images in the database,  $I_r, r = 1, 2, \dots, N$ , are ranked according to their distances,  $d_r$ , to the query set  $Q$  and the top  $K$  images are output as the retrieval result. It is obvious that obtaining the intra- and inter-weight  $u_i, i = 1, 2, \dots, F_i$  and  $w_i, i = 1, 2, \dots, F$  from the query set  $Q$  becomes a key issue in QBMI.

In our project, we choose MPEG-7 visual descriptors [57, 8] as the feature set,  $f_i, i = 1, 2, \dots, F$ . For each feature or descriptor, the similarity metric proposed in

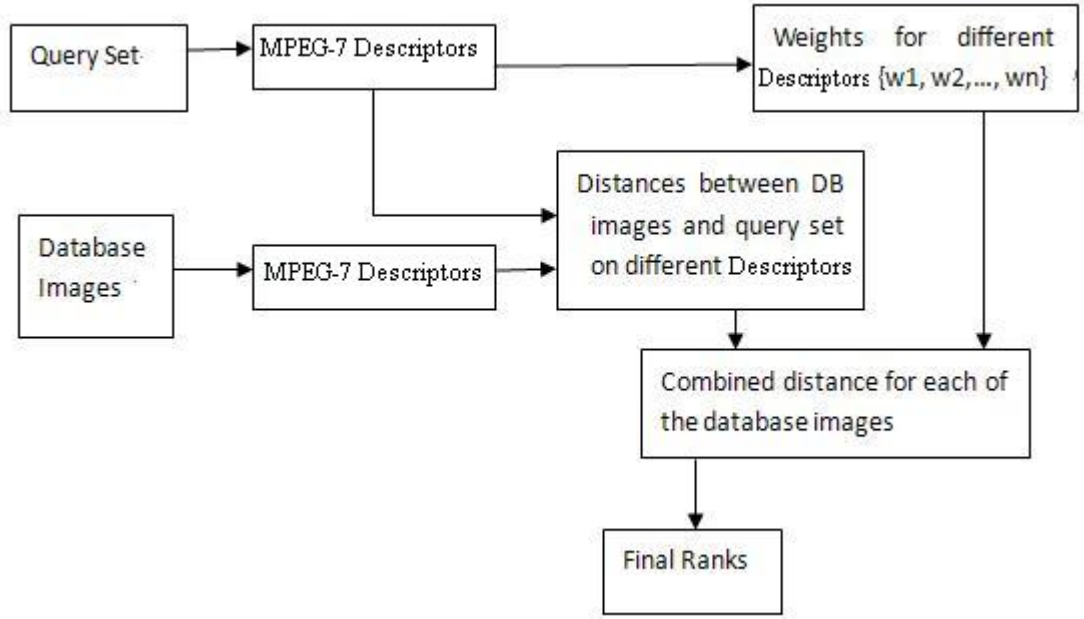


Figure 4.1: The retrieval procedure of a QBMI using MPEG-7 visual descriptors

MPEG-7 is used to calculate the distance  $d_{ir}^j$ . Since feature component weight,  $u_i$ , has been well investigated and incorporated in the similarity measurement of MPEG-7 descriptors to match human visual perception, our focus is on the estimation of the inter-weights. Figure 4.1 shows the procedure of our QBMI based on MPEG-7 descriptors. Notice that the terms *visual descriptor*, *descriptor* and *feature* have the same meaning in the rest of the paper and shall be used interchangeably hereafter.

The principle of calculating the inter-weights  $w_i$  is to assign higher values to the descriptors that well describe the cluster represented by the query set  $Q$ . In the following section, we formulate the estimation of  $w_i, i = 1, 2, \dots, F$  as a feature selection problem and propose a new method to calculate  $w_i$  from the query set  $Q$ .

### 4.3 Calculation of Inter-weights Using MI

Let  $C$  denote the image cluster represented by the query set  $Q$ . We assume there is more than one query image in  $Q$ , i.e.  $n > 1$ . The importance of each feature or descriptor,  $f_i$ , can be measured based on its discriminative power with respect  $C$ . Therefore, the calculation of inter-weights for the  $F$  features or descriptors,  $f_i, i = 1, 2, \dots, F$ , can be considered as a problem of feature selection for  $C$ , i.e. to rank the features with respect to  $C$ .

Many feature selection algorithms (FSAs) have been proposed in the past and thorough literature reviews can be found in [15, 47, 29]. Broadly, FSAs are divided into three categories [48, 47]: filter model [43], wrapper model [40] and hybrid model [79, 48].

The filter model exploits the general characteristics of the data and selects features without involving any classification. One popular technique for filter model is mutual information (MI) [13]. MI is a measure of correlation between two variables. It is often used to calculate the relevance of a feature with respect to a target class and redundancy among features. The fact that MI is independent of the chosen coordinates permits a robust estimation of the relevance [65, 3]. The wrapper model selects features by directly using classification results as a criterion for ranking the features or selecting the minimum subset of features. Although the subset found by a wrapper algorithm is best suited to the employed classifier, wrapper model tends to be more computationally expensive than the filter algorithms [40, 47]. The hybrid model is a combination of both approaches, taking the advantages of each model at different search stages [48].

The characteristic of not involving explicit classification in the filter model and the effectiveness of MI to measure the correlation between two random variables make the MI based filter model more appealing than other models for the inter-weight calculation.



### 4.3.1 Mutual Information

Formally, the mutual information of two discrete random variables  $X$  and  $Y$  can be defined as:

$$I(X; Y) = \sum_x \sum_y p(x, y) \log \frac{p(x, y)}{p(x)p(y)} \quad (4.5)$$

where  $x, y$  represent the discrete states of two discrete random variables  $X$  and  $Y$  respectively,  $p(x, y)$  is the joint probability distribution function of  $X$  and  $Y$ , and  $p(x)$  and  $p(y)$  are the marginal probability distribution functions of  $X$  and  $Y$  respectively.

Mutual information quantifies the distance between the joint distribution of  $X$  and  $Y$  and what the joint distribution would be if  $X$  and  $Y$  were independent. Intuitively, mutual information measures the information that  $X$  and  $Y$  share: it measures how much knowing one of these variables reduces the uncertainty about the other. For example, if  $X$  and  $Y$  are independent, then knowing  $X$  does not give any information about  $Y$  and vice versa,  $p(x, y) = p(x)p(y)$ , so their mutual information is zero. At the other extreme, if  $X$  and  $Y$  are identical then all information conveyed by  $X$  is shared with  $Y$ : knowing  $X$  determines  $Y$  and vice versa. The higher the mutual information's value, the more correlation exists between  $X$  and  $Y$ .

Calculation of the mutual information  $I(X; Y)$  requires the joint probability  $p(x, y)$  and marginal probability  $p(x)$  and  $p(y)$ . Alternatively,  $I(X; Y)$  can be calculated from entropies which measure the uncertainty of random variables.

$$I(X; Y) = H(X) - H(X|Y), \quad (4.6)$$

where  $H(X)$  is the entropy of  $X$  and  $H(X|Y)$  is the conditional entropy of  $X$  given

$Y$ .

$$H(X) = \sum_x p(x) \log p(x) \quad (4.7)$$

$$H(X|Y) = \sum_x p(x) \sum_y p(x|y) \log p(x|y) \quad (4.8)$$

### 4.3.2 Calculation of inter-weights from MI

We consider both  $C$  and  $f_i$  as random variables and use the mutual information  $I(C; f_i)$  as a measurement of the discriminative ability of  $f_i$  for  $C$ . The higher the mutual information value, the more discriminative is the feature and, hence, the more important is the feature. From an information-theoretic viewpoint, we can say that the amount of information regarding  $C$  is revealed by  $f_i$ . Thus, for a set of features or descriptors, we want to quantify the amount of information it conveys about the cluster  $C$ . We can then ascribe a high degree of relevance to the feature that has most information, i.e.

$$w_i = \frac{I(C; f_i)}{\sum_{i=1}^F I(C; f_i)}, \quad (4.9)$$

and

$$I(C; f_i) = H(f_i) - H(f_i|C) \quad (4.10)$$

Calculation of  $I(C; f_i)$  requires a large number of samples of  $f_i$  taken from  $C$  so that the probability functions can be estimated properly. In our application, the number of query images in  $Q$  is usually small and the feature or descriptor,  $f_i$  is usually a high dimensional vector. Computation of the marginal probabilities and conditional or joint probabilities from such a small set of samples becomes problematic.

## 4.4 Proposed Weighting Method

[14] observed that data with clusters has very different point-to-point distance histogram than data without clusters. They proposed distance entropy to measure how well the data samples are clustered in a feature space. The entropy is low if data has distinct clusters and high otherwise. By analogy, if feature  $f_i$  describes the cluster of the query set  $Q$  well, the distances between two images from the query set in this feature space would form a distinct cluster. How well the distances are clustered is an indication of the importance of the feature. Inspired by the concepts of the distance entropy and mutual information described above, we propose the following method to calculate the inter-weights instead of using Eq.( 4.9) directly.

Let  $D_{ab}^i$  be the normalized distance between the query images,  $q_a$  and  $q_b$ , in feature space of  $f_i$  and  $D^i = \{D_{ab}^i, a = 1, 2, \dots, n; b = a + 1, a + 2, \dots, n\}$ . For simplicity, we rewrite  $D^i = \{D_s^i, s = 1, 2, \dots, S\}$ , where  $S = \frac{n(n-1)}{2}$ . We propose to assign  $w_i$  to feature  $f_i$  as below

$$w_i = \frac{J(C; D_i)}{\sum_{i=1}^F J(C; D_i)} \quad (4.11)$$

$$J(C; D^i) = E(D^i) - E(D^i|C) \quad (4.12)$$

where  $E(\cdot)$  is the distance entropy function.  $E(D^i)$  is distance entropy of  $D^i$  and  $E(D^i|C)$  is the conditional distance entropy of  $D^i$  on  $C$ .

The distance entropy can be calculated as

$$E(D^i) = -\sum_{s=1}^S \left( \frac{1}{S} \log \frac{1}{S} \right) \quad (4.13)$$

$$= -\log \frac{1}{S} \quad (4.14)$$

Considering all query images in  $Q$  form one cluster  $C$ , the conditional entropy  $E(D^i|C)$  becomes

$$E(D^i|C) = - \sum_{s=1}^S p(D_s^i|c) \log p(D_s^i|c) \quad (4.15)$$

where  $p(D_s^i|c)$  is a function that is subject to

$$p(0|c) = 1.0 \quad (4.16)$$

$$p(1.0|c) = 0.5 \quad (4.17)$$

$p(1.0|c)$  is 0.5 which has maximum uncertainty when the distance reaches the largest.

A simple linear function that satisfies the conditions is

$$p(D_s^i|c) = 1 - D_s^i/2; \quad (4.18)$$

#### 4.4.1 Normalization of the distances

We choose MPEG-7 visual descriptors as features to describe images and the distance between two images with respect to a particular descriptor is calculated according to the corresponding similarity measurement defined in MPEG-7. The distance for each descriptors varies within a wide range. For instance, the CLD distance ranges from 0 to 470.4 and CSD distance ranges from 0 to 65536. The distance has to be normalized before it can be mapped to probability using Eq.( 4.18). Let  $d$  be the distance calculated with respect to a descriptor and  $D$  is the normalized distance, i.e.

$$D = \text{norm}(d), D \in [0, 1] \quad (4.19)$$

where  $\text{norm}(\cdot)$  is a function to normalize the distance  $d$ .

A common method is to use the linear function:

$$norm(d) = \frac{d - min}{max - min + 1} \quad (4.20)$$

where  $min$  and  $max$  are respectively the minimum and maximum possible values of  $d$ .

However, linear normalization has been found problematic when applied to MPEG-7 descriptors. This is mainly due to two factors. First, the maximum possible distance varies significantly among the MPEG-7 descriptors. Second, for a particular database, the distances of each descriptor usually falls into a small subrange of the entire possible range. As a result, the linear normalization will possible compact the distances into a very narrow and indiscriminate range within  $[0, 1]$  and distances of different descriptors will be mapped to different subranges within  $[0, 1]$  which makes the distances of different descriptors incomparable.

We employ Gaussian normalization [72] that puts equal emphasis on the distances in each of the feature spaces. By doing this we normalize the distances of each descriptor into a normal distribution with standard deviation being equal to 1.0.

$$norm(d) = \frac{1}{2} \left( 1 + \frac{d - \mu}{3 * \sigma} \right) \quad (4.21)$$

where  $\mu$  is the mean value of the distances and  $\sigma$  is the standard deviation of the distances, both are calculated from the image database.

Eq.( 4.21) will normalize 99.7% of the distances to  $[0, 1]$ . For the remaining 0.3% percent, we simply set them to the nearest 0 or 1.

The advantage of this normalization processes over (4.20) is that the presence of a few abnormally large or small values does not bias the importance of a component in computing the similarity between two images.

## 4.5 Conclusion

Descriptive view is often adopted or implied by users in retrieving images with multiple image examples. When MPEG-7 descriptors are used to describe the visual content of the images in QBMI, the key issue is to compute the importance of every descriptor in relation to the ideal images in the users' mind. This chapter presents a new method to measure the importance of the descriptors given the query set which aims to match human perception. Experiment results will be presented in the next chapter.

# Chapter 5

## Experimental Results

This chapter describes a QBMI system based on the proposed new weighting method and reports its performance on an image database that consists of about 9,000 diverse images. The performance is measured in precision and recall and compared with other weighting methods. Experimental results have demonstrated that the proposed method outperforms the others.

### 5.1 Overview of the QBMI system

We have implemented a QBMI (Query Based Multiple Images) System based on the proposed weighting method. The system supports both single image and multiple image queries. For single image query, it uses equal weights for different features. For multiple image queries, it applies our new weighting method described in Chapter 4 to calculate the weights for the selected set of features. Four MPEG-7 visual descriptors are employed as the candidate features to describe the images. They are Colour Layout Descriptor (CLD), Colour Structure Descriptor (CSD), Edge Histogram Descriptor (EHD) and Homogeneous Texture Descriptor (HTD). CLD and CSD are used to describe the colour information contained in the images, whereas EHD and HTD

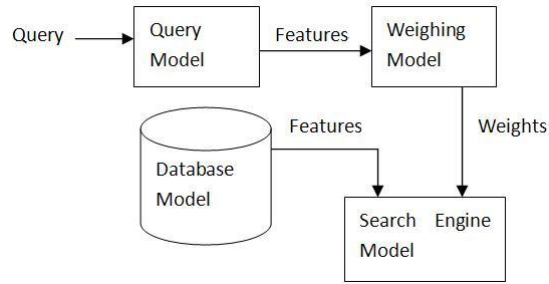


Figure 5.1: Four models of a QBMI system

are used to describe the texture information of the images.

The system consists of four modules: Query Module, Weighting Module, Database Module and Search Engine Module. Figure 5.1 shows how the models interact.

**Query Module** Users are required to select one or more sample images as the query set and a set of visual descriptors to perform the retrieval. For example, images of three cars in different colours may be chosen to form a query set and CLD and EHD may be employed for searching for images of cars. The query images will be passed to the Weighting Module to calculate the weights of the chosen descriptors. The resultant weights will be output to the Search Engine where images in the database will be compared with the query set and ranked accordingly.

**Weighting Module** This module is used to dynamically calculate the weights for the descriptors from the query image set selected in the Query Module. If the query set contains only one image, this module will assign equal weights to all the descriptors. If there is more than one image in the query set, the weights are calculated using the new weighting method described in Chapter 4.

**Database Module** To improve the system response time, the distances between all pairs of images in the database with respect to the four descriptors are precomputed and stored in the database. MPEG-7 eXperimentation Model (XM) is



employed to extract the descriptors and calculate the distances. The distances are loaded into the Search Engine Module and the Weighing Module whenever needed.

**Search Engine Module** The Search Engine Module takes the query set and calculated weights for the descriptors and computes the relevance (i.e. combined distance) between every image in the database and the query set. All images from the database are ranked based on their relevance. The top  $K$  images are then output as the retrieval result.

## 5.2 Experimental Setup

A collection of 8870 images was included in the image database. All images are stored in JPEG format. The collection consists of many categories of images including vehicles, landscapes, animals, plants and buildings. The size of the images varies from 170 x 128 pixels to 3721 x 3086 pixels.

The minimum function was adopted in calculation of the distances between a database image,  $I_r$ , and a query set,  $Q$ . For  $j$ 'th descriptor. Eq.( 4.3) becomes

$$d_r^j = \min_{1 \leq i \leq n} d_{ir}^j, r = 1, 2, \dots, N; j = 1, 2, \dots, F \quad (5.1)$$

The overall distance between  $I_r$  and  $Q$  is a linear combination of the distances with respect to the descriptors selected for retrieval.

$$d_r = \sum_{j=1}^F w_j * d_r^j \quad (5.2)$$

$d_r$  is used to rank all images ( $r = 1, 2, \dots, N$ ) for output.

We have run various queries on the system in order to evaluate its performance.

Results are partially reported below.

## 5.3 Experimental Results

### 5.3.1 Multiple descriptors vs. Single descriptor

Our perception experiments assert that using a single descriptor for retrieval has drawbacks and does not perform well in most situations. Multiple descriptors can improve the retrieval performance and give better results. In the following, we present two experiments to further confirm this assertion. In the first experiment, three sunset images were selected to form the query set for retrieving the image category of sunset. The query set and ground truth are shown in Figure 5.2.

Figure 5.3 shows the precision and recall curves of the retrieval based on CSD, EHD and combination of CSD and EHD respectively. In the case where both CSD and EHD are used, the weights for CSD and EHD were calculated using the proposed weighting method. It can be seen from the curves that CSD performed better than EHD when they were used alone, and therefore, colour information is more important than edge information in describing sunset images. When both descriptors were used, the weights calculated by our method for CSD and EHD were 0.565 and 0.435 respectively. This also indicates that colour information is more important

In the second experiment, a set of images containing towers was selected as query examples. The query images and ground truth are shown in Figure 5.4. We chose the query images having similar towers and different background (especially in colour). This meant that the user's intention was to search images with towers in any background. Again, CSD and EHD were used alone and together. The precision and recall curves are shown in Figure 5.5. As expected, EHD performed better than CSD when they were used alone. When CSD and EHD were employed together, retrieval per-

Figure 5.2: Query set and ground truth for “Sunset”

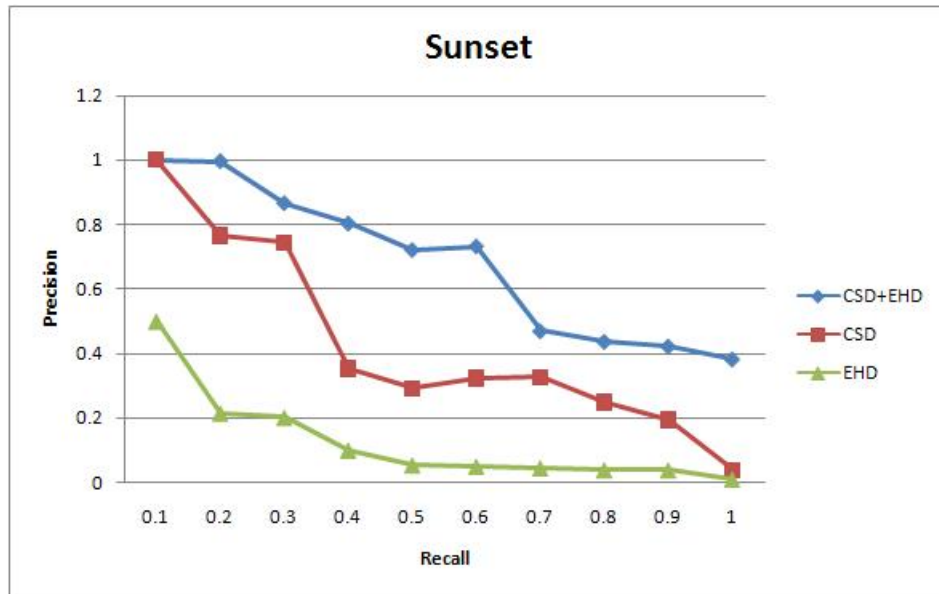


Figure 5.3: Single descriptor vs multiple descriptors for “Sunset”

formance was improved significantly. The weights calculated by the proposed method were 0.381 for CSD and 0.619 for EHD, which indicates that edge information is more important than colour information for this set of query images.

### 5.3.2 Proposed weights vs. equal weights

It is easy to understand that multiple descriptors usually perform better than a single descriptor since more information is used to characterize the cluster defined by the query set. In this section, we compare the retrieval performance by employing equal weights to the multiple descriptors with the performance achieved using the weights calculated from the proposed method.

We used both CSD and EHD to search the images containing sphinx. Figure 5.6 shows the query images and the ground truth. Our weighting method set the weights 0.542 and 0.245 for CSD and EHD respectively, as shown in Table 5.1. The weights indicate that colour information is more discriminative than edge information for sphinx. The precision-recall curves shown in Figure 5.7 demonstrates that this set of weights

Figure 5.4: Query set and ground truth for “Tower”

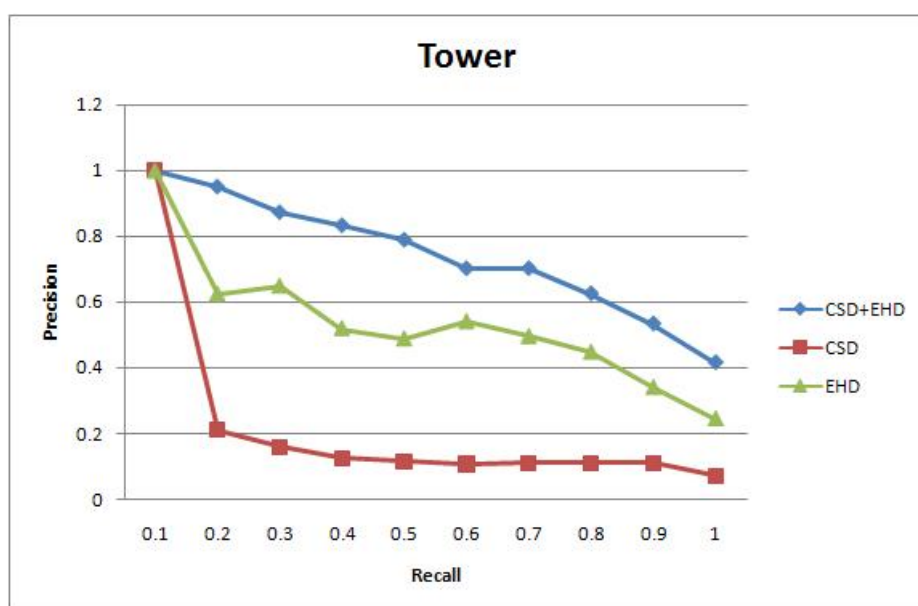


Figure 5.5: Single descriptor vs multi descriptors for “Tower”

Figure 5.6: Query set and ground truth for “Sphinx”

	CSD	EHD
Equal weights	0.5	0.5
Proposed weights	0.542	0.458

Table 5.1: Feature weights for “Sphinx”

performs much better than equal weights.

Experiments on other images including the Towers as shown in Figure 5.4 consistently verified that the proposed method is superior to the equal weights.

### 5.3.3 Proposed weights vs. heuristic weights

Heuristics are commonly used in relevance feedback (RF) to calculate the weights of features. The concept behind the heuristic method is that if a particular feature captures the user’s intention and preference, then its values for different query images should be consistent. In other words, the standard deviation of the feature values



Figure 5.7: Equal weights vs proposed weights for “Sphinx”

should be small. We map this concept to the distance spaces of the descriptors and make an analogy that an important descriptor that captures user’s preference would lead to a small average distance between query images and small standard deviation as well. Let  $\mu_i$  be the average distance for the  $i$ ’th descriptor and  $\sigma_i$  be the standard deviation. According to [68], the weight for the  $i$ ’th descriptor is defined as

$$u_j = \frac{1}{\sqrt{\sigma_j * \mu_j}}, (j = 1, 2, \dots, F) \quad (5.3)$$

$$w_j = \frac{u_j}{\sum_{j=0}^k u_j} \quad (5.4)$$

Figure 5.9 and Figure 5.10 compare the performance of the heuristic method and the proposed method in search for “Yellow Butterfly” and “Sphinx” respectively when CSD and EHD were employed. The query set and ground truth images are shown in Figure 5.8 and 5.6. Table 5.2 and 5.3 lists the weights assigned to the two descriptors respectively by heuristics and our method. Both methods indicate consistently which descriptor is more important in both cases. However, our method is more effective to

Figure 5.8: Query set and ground truth for “Yellow Butterfly”

	CSD	EHD
Heuristic Weights	0.336	0.664
Proposed Weights	0.471	0.529

Table 5.2: Feature weights for “Yellow Butterfly”

reveal the relative importance of the two descriptors and, consequently, leads to better performance.

#### 5.3.4 Overall System Performance

To evaluate the overall performance of the system, experiments were conducted over query sets formed from twenty different categories of images, including cars, butterflies, beach, building, flowers, mountain, sunset, etc. Figure 5.11 shows the average precision and recall curve. From the curve we can see that the precision is above 90% when

	CSD	EHD
Heuristic Weights	0.507	0.493
Proposed Weights	0.542	0.458

Table 5.3: Feature weights for “Sphinx”



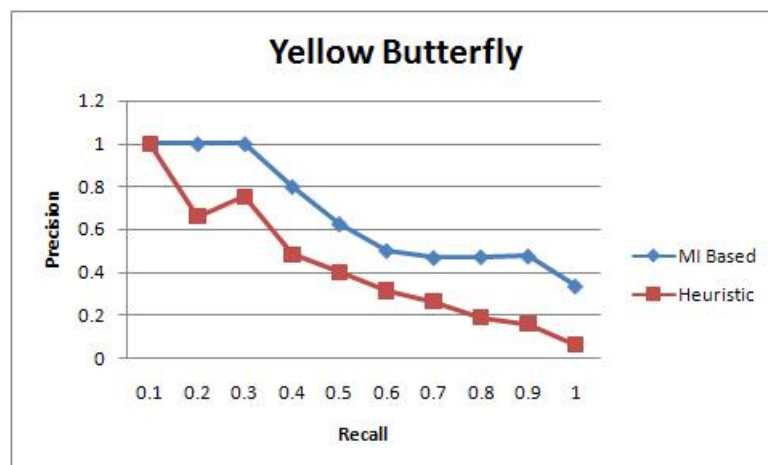


Figure 5.9: Proposed weights vs heuristic weights in category “Yellow Butterfly”

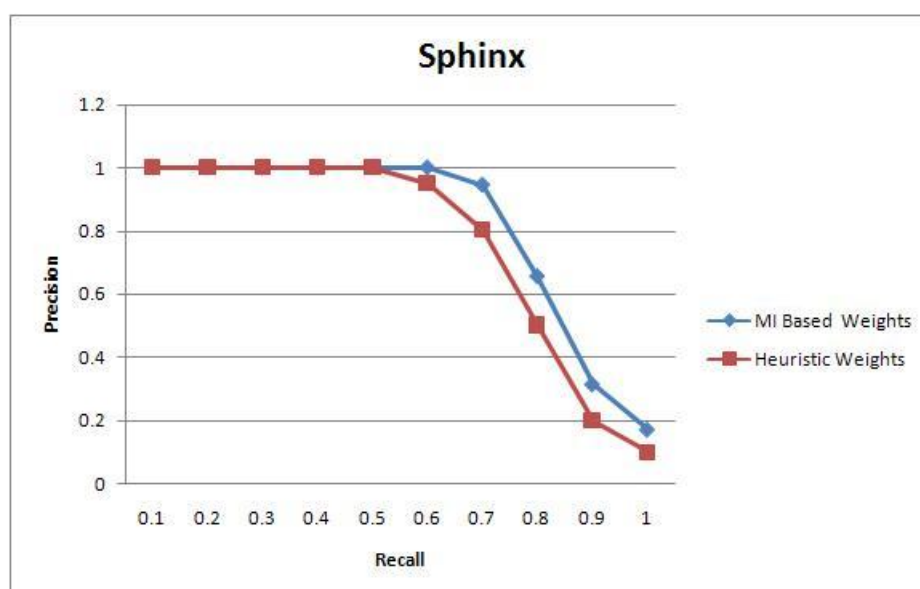


Figure 5.10: Proposed weights vs heuristic weights in category “Sphinx”

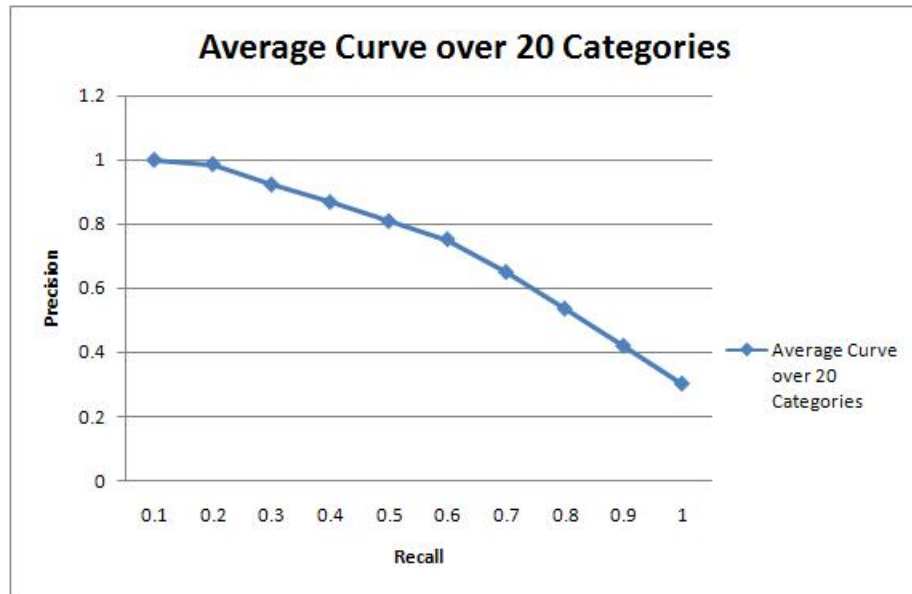


Figure 5.11: Average precision and recall curve over twenty image categories

30% of the relevant images are retrieved. It reduces slowly and reaches 80% when half of the relevant images are retrieved. Then the precision drops quickly. When all the relevant images are retrieved the precision falls to around 30%.

Following are a few example query sets and ground truths from the 20 selected categories. The precision and recall curves were computed and are shown as well. Figure 5.12 shows the query set and the ground truth in the category of “Lilies”. Three images with Lilies in a background of green leaves were selected to be the query set. The purpose of this query was to find out all the images which contain Lilies in a green background. As we can see from the precision-recall curve shown in Figure 5.13, for the first 70 percent of retrieved relevant images, the precision is over 80%. It remains high (over 60%) until 90% of the ground truth were retrieved.

Figure 5.14 shows the query set and ground truth of “Flowers”. There were 3 images contained in the query set. They have similar shape but different colours. The aim of this query was to find out all the images of flowers of similar shape to the flowers in the query set regardless of the colour. Figure 5.15 is the retrieval precision-recall

Figure 5.12: Query set and ground truth for “Lilies”



Figure 5.13: Precision and recall curve for “Lilies”

Figure 5.14: Query set and ground truth for “Flowers in Different Colour”

curve. As indicated, the precision is almost 100% up to 50% of the ground truth images being retrieved.

In all, we have verified that the image retrieval system built upon the proposed weighting method performs well for a wide range of images.

### 5.3.5 Semantic retrieval

One of the challenges in image retrieval is the well-known “semantic gap” that exists between the semantic meaning of the images and their low-level features or descriptors. Both relevance feedback (RF) and QBMI aim to narrow the gap. It would be interesting to see how well the proposed weighting method is able to differentiate images that are considered to be at different semantic levels, for instance, images containing “green objects” vs. images containing “green cars”. We refer to this type of retrieval as *semantic retrieval* and conducted a number of preliminary experiments. Two examples are reported below.

The first example was about the concepts of “flowers” and “red flowers”. Fig-



Figure 5.15: Precision and recall curve for “Flowers in Different Colour”

Figure 5.16: Query set for “Red Flowers”

Figure 5.14 and 5.16 contain respectively the query sets for the two concepts. Their ground truth images were chosen by human subjects and are shown in Figures 5.14 and 5.17 respectively.

CSD and EHD were used for the retrieval. Table 5.4 shows the weights assigned to each descriptor. It is interesting to see that CSD was assigned with a higher weight in search for “Red flowers” whereas EHD was assigned with a higher weight in retrieving “Flowers”. This appears consistent with human perception. For “Red flowers”, colour information which is described by CSD is more prominent than texture and shape information described by EHD. The retrieved images are shown in Figure 5.18 and 5.19.

Figure 5.20 shows the precision-recall curves. As noticed, the retrieval of “Red Flowers” had higher precision than that of “Flowers” at all recall levels. This may

Figure 5.17: Ground truth for “Red Flowers”

	CSD	EHD
flowers	0.425	0.575
red flowers	0.551	0.449

Table 5.4: Feature weights for “Flowers” and “Red Flowers”

Figure 5.18: Retrieval result of “Flowers in Different Colour”

Figure 5.19: Retrieval result of “Red Flowers”

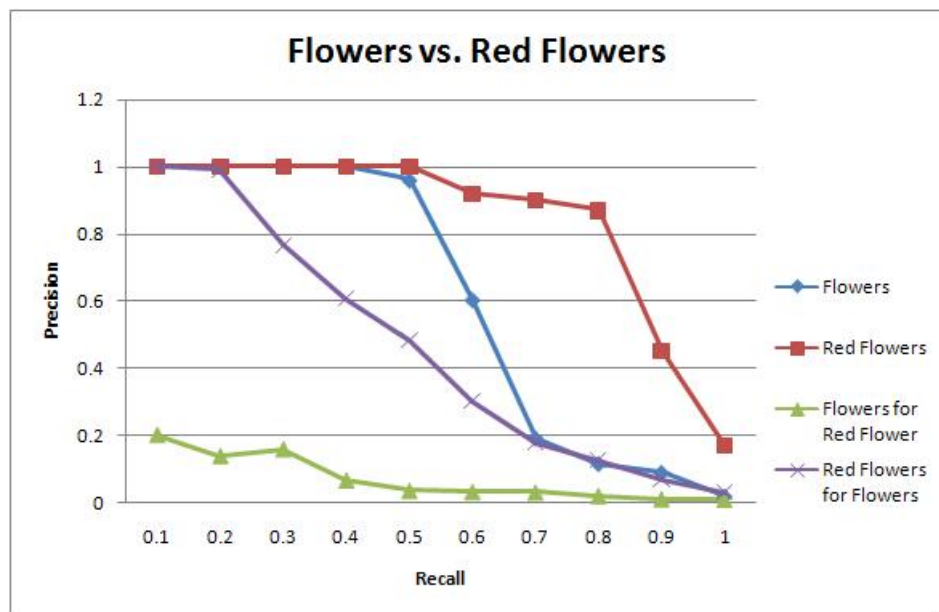


Figure 5.20: Semantic retrieval: “Flowers” vs “Red Flowers”

Figure 5.21: Query set for “Building in a desert”

be due to the query set for “Flowers” and/or the descriptors being not discriminative enough for the concept of flowers. For a given concept, how to choose examples and descriptors remains as an open question for future research.

We also conducted experiments of cross-retrieval which used the query set of flowers to retrieve red flowers and the query set of red flowers to retrieve flowers. Their precision-recall curves are shown in Figure 5.20. As expected, the performance is worse than the cases when a correct query set was used.

The second example is about the concepts of “Building in a desert” and “Sphinx”. Figure 5.21 and 5.22 show the query set and ground truth of “Building in a desert”. Figure 5.6 shows the query set and ground truth of “Sphinx”, which is a subset of “Building in a desert”.

Similar results to the first example were obtained as evidenced by the precision and recall shown in Figure 5.23.

## 5.4 Conclusion

We present a QBMI system in this Chapter. The system is based on MPEG-7 visual descriptors and the weighting method proposed in Chapter 4. Experimental results have shown that our weighing method is more effective than both equal weights and heuristic weighting method. In addition, we demonstrate preliminarily that our weighting method is able to narrow the “semantic gap”.



Figure 5.22: Ground truth for “Building in a desert”

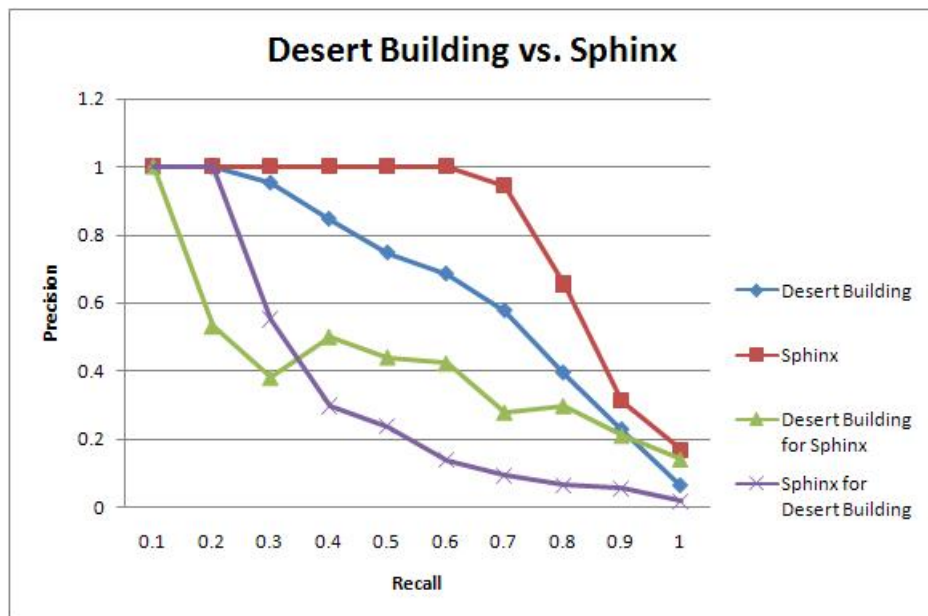


Figure 5.23: Semantic retrieval: “Desert Buildings” vs “Sphinx”

# Chapter 6

## Conclusion

A brief summary and conclusion of this thesis are presented in this chapter, as well as some suggestions on future work.

### 6.1 Summary and Conclusion

Content based image retrieval remains a vibrant research area especially because of the difficult task of inferring users' intention from the query posed. The “semantic gap” resulting from the inability of low-level features including color, texture and edge to describe images on a semantic level remains unbridged. However limited success has been achieved through combinations of these features in order to improve the system performance and to match human perception during retrieval. In this thesis, a set of experiments are designed to gauge how human subjects use the low-level features to describe their target images and complete retrieval tasks. MPEG-7 descriptors are used to select similar images to the given query. The descriptors are Color Layout Descriptor (CLD), Color Structure Descriptor (CSD), Edge Histogram Descriptor (EHD) and Homogeneous Texture Descriptor (HTD). The selection results are compared with the corresponding images selected by human subjects. It was found

that using one descriptor is not good enough to retrieve target images in all the cases. To get a better retrieval performance, combining the descriptors is necessary. The use of equal weights proved inadequate in retrieval tasks. The salient features need to be determined and assigned higher weights to improve the precision.

A CBIR approach based on multiple query images is proposed in this thesis. Query composed by multiple images are considered to be able to describe users' target image more clearly. In essence query by multiple images can offer the possibility to capture the query concept underlying the users' intention. Weights of the features are generated from the query images based on information theoretic concepts. From the viewpoint of information theory, the feature along which the most mutual description of the query set is obtained is considered to be the most important feature. Entropies and mutual information are used to represent the information contained by the feature. By doing this, features which are considered to be important should be assigned higher weights. An image retrieval system based on the proposed approach is implemented and its performance compared with systems employing other approaches. By comparing the precision and recall curves of the systems, it is shown that the proposed approach is able to improve the system performance and the retrieval precision.

## 6.2 Future Work

The approach taken in this thesis and results obtained have generated ideas for new directions and improvements to this research. Possible improvements and further studies on the proposed methods are addressed below:

- The human perception experiments introduced in Chapter 3 required the user to pick up a group of images that provides best description of the example images. This can be used to test if people can use a set of images to describe the target image. Further experiments might be designed to gain more insight about this

notion. For example, the images picked up by human subjects can be ranked by the number of people who selected them. The top 5 images are chosen to be the example images. The old example images could then be put back into the database. When the subjects are given 5 example images and required to pick up one image which can be best described by them, it will be interesting to see if the user will pick up the old example image.

- The proposed image retrieval approach focuses on query by multiple images. The relevance feedback approach could be combined with the current approach. The user is required to pick up positive feedback from a given image set. Also the proposed weighting method is used to get the weights from the feedback images in each iteration. The negative feedback may also be considered in relevance feedback model. Negative feedback might also be used to adjust the weights by applying information theory.
- In the image retrieval system implemented using the proposed approach, the distance between a database image and the query set on feature  $j$  is considered to be the smallest value in distance set  $D_j$ .  $D_j$  contains all the distances between any query image and the database image. This method considers only one image from the query set and may not be the best way to represent the distance between query set and database image. Using average value of the set  $D$  takes all the query images into account. This might be able to improve the performance of the retrieval system.
- Information theory can not only be used to decide the weights of the features, but also the weights of the query images. By calculating the mutual information for each image, we can assign weights to each image in the query set. The query image with a higher weight is considered to be more important than other query

---

images. By applying this method, the distance between query set and database image on feature  $j$  could be calculated by summing all the distances in  $D_j$  with assigned weights.

# References

- [1] I. Ahmad and T. Jang. Old fashion text-based image retrieval using FCA. *Image Processing, 2003*, 2:III–33–6, September 2003.
- [2] J. Andrade and J. May. *Cognitive Psychology*. BIOS Scientific Publishers, London and New York, 2nd edition, 2004.
- [3] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. on Neural Networks*, 5(4):537–550, 1994.
- [4] N. J. Belkin, C. Cool, W. Bruce Croft, and J. P. Callan. The effect multiple query representations on information retrieval system performance. In *SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 339–346, New York, NY, USA, 1993. ACM Press.
- [5] A. B. Benitez, S. Paek, S. Chang, A. Puri, Q. Huang, J. R. Smith, C. Li, L. D. Bergman, and C. N. Judice. Object-based multimedia content description schemes and applications for MPEG-7. *Signal Processing: Image Communication*, 16:235–269, September 2000.
- [6] A. P. Berman and L. G. Shapiro. A flexible image database system for content-based retrieval. *Comput. Vis. Image Underst.*, 75(1-2):175–195, 1999.

- 
- [7] T. E. Bjoerge and E. Y. Chang. Why one example is not enough for an image query. *2004 IEEE International Conference on Multimedia and Expo(ICME)*, 1:253–256, June 2004.
  - [8] ISO/IEC CD15938-3. Multimedia content description interface-part 3: Visual. Final committee draft. *ISO/IEC/JTC1/SC29/WG11, Doc. N4062*, March 2001.
  - [9] M.E. Celebi and Y.A. Aslandogan. Content-based image retrieval incorporating models of human perception. *Information Technology: Coding and Computing*, 2:241–245, April 2004.
  - [10] E. Y. Chang, B. Li, and C. Li. Toward perception-based image retrieval. In *CBAIVL '00: Proceedings of the IEEE Workshop on Content-based Access of Image and Video Libraries (CBAIVL'00)*, page 101, Washington, DC, USA, 2000. IEEE Computer Society.
  - [11] S.-C. Chen. Indexing and searching structure for multimedia database systems. In M. M. Yeung, B.-L. Yeo, and C. A. Bouman, editors, *Proceedings of SPIE: Storage and Retrieval for Media Databases 2000*, volume 3972, pages 262–270, 2000.
  - [12] V. Rijsbergern C.J. *Information Retrieval*. Butterworths, London, 2nd edition, 1979.
  - [13] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley & Sons, Inc., 1991.
  - [14] M. Dash, K. Choi, P. Scheuermann, and H. Liu. Feature selection for clustering - a filter solution. In *ICDM '02: Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM'02)*, page 115, Washington, DC, USA, 2002. IEEE Computer Society.



- 
- [15] M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1:131–156, 1997.
  - [16] R. Datta, J. Li, and J. Z. Wang. Content-based image retrieval: approaches and trends of the new age. In *MIR '05: Proceedings of the 7th ACM SIGMM international workshop on Multimedia information retrieval*, pages 253–262, New York, NY, USA, 2005. ACM Press.
  - [17] S. Deb and Y. Zhang. An overview of content-based image retrieval techniques. *Advanced Information Networking and Applications, 2004. AINA 2004. 18th International Conference*, 1:59–64, 2004.
  - [18] J. Driver, G. Davis, C. Russell, M. Turato, and E. Freeman. Segmentation, attention and phenomenal visual objects. *Cognition*, 80(1-2):61–95, June 2001.
  - [19] M. W. Eysenck and M. Keane. *Cognitive Psychology: A Student's Handbook*. Psychology Press, 5th edition, 2005.
  - [20] J. Feng, M. Li, H. Zhang, and B. Zhang. Region-based relevance feedback in image retrieval. *IEEE International Symposium*, 4:145–148, May 2002.
  - [21] M. Flickner, H. Sawhney, and W. Niblack. Query by image and video content: The QBIC system. *IEEE Computer*, 28(9):23–32, September 1995.
  - [22] J. French, J. Watson, X. Jin, and W. Martin. Using multiple image representations to improve the quality of content-based image retrieval. Technical Report CS-2003-10, Dept. of Computer Science, Univ. Virginia, Mar 2003.
  - [23] R. Gelman and T. K. Au, editors. *Perceptual and Cognitive Development*. Academic Press Limited, San Diego, California, 2nd edition, 1996.

- 
- [24] T. Gevers and A. W. M. Smeulders. The PicToSeek WWW image search system. In *Proceedings of IEEE ICMCS Multimedia Systems*, pages 264 – 269. IEEE Press, 1999.
- [25] T. Gevers and A.W.M. Smeulders. PicToSeek: Combining color and shape invariant features for image retrieval. *IEEE Transactions on Image Processing*, 9(1):102–119, Jan 2000.
- [26] A. Goodrum. Image information retrieval: An overview of current research. *Informing Science The International Journal of an Emerging Transdiscipline*, 3(2):63–66, 2000.
- [27] MPEG-7 Requirements Group. MPEG-7: Context, objectives and technical roadmap, v.12. *MPEG Vancouver Meeting*, July 1999. ISO/IEC SC29/WG11 N2861.
- [28] MPEG-7 Requirements Group. MPEG-7 requirements document v.9. *MPEG Vancouver Meeting*, July 1999. ISO/IEC SC29/WG11 N2859.
- [29] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.
- [30] N. Hiransakolwong, K. A. Hua, S. Koompaiojn, K. Vu, and S. Lang. An adaptive distance computation technique for image retrieval systems. In *SAC '05: Proceedings of the 2005 ACM symposium on Applied computing*, pages 1195–1199, New York, NY, USA, 2005. ACM Press.
- [31] S. Hochstein and M. Ahissar. View from the top: Hierarchies and reverse hierarchies. *Neuron*, 36:791–804, December 2002.
- [32] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

- 
- [33] Y. Huang, T. Change, and C. Huang. A fuzzy feature clustering with relevance feedback approach to content-based image retrieval. *Virtual Environments, Human-Computer Interfaces and Measurement Systems*, pages 57–62, July 2003.
- [34] 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(VISUAL 2003)*, pages 467–474, Miami, Florida, September 2003.
- [35] J. Li J. Z. Wang and G. Wiederhold. SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. *Pattern Analysis and Machine Intelligence*, 23(9):947–963, September 2001.
- [36] S. Jeong, K. Kim, B. Chun, J. Lee, and Y. J. Bae. An effective method for combining multiple features of image retrieval. In *TENCON 99. Proceedings of the IEEE Region 10 Conference*, volume 2, pages 982–985, Cheju Island, South Korea, Dec 1999.
- [37] X. Jin and J. C. French. Improving image retrieval effectiveness via multiple queries. In *MMDB '03: Proceedings of the 1st ACM international workshop on Multimedia databases*, pages 86–93, New York, NY, USA, 2003. ACM Press.
- [38] F. Jing, B. Zhang, F. Lin, W. Ma, and H. Zhang. A novel region-based image retrieval method using relevance feedback. In *MULTIMEDIA '01: Proceedings of the 2001 ACM workshops on Multimedia*, pages 28–31, New York, NY, USA, 2001. ACM Press.
- [39] T. Jost, N. Ouerhani, R. von Wartburg, R. Muri, and H. Hugli. Assessing the contribution of color in visual attention. *Computer Vision and Image Understanding*, 100(1-2):107–123, 2005.

- 
- [40] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):23–34, 1997.
  - [41] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer-Verlag, 3rd edition, 2001.
  - [42] M. Koskela, J. Laaksonen, and E. Oja. Self-organizing hierarchical feature maps. *Proc.IJCNN-90, International Joint Conference on Neural Networks*, 2:279–284, June 1990.
  - [43] N. Kwak and C.-H. Choi. Input feature selection by mutual information based on parzen window. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 24(12):1667–1671, 2002.
  - [44] J. Laaksonen, M. Koskela, and E. Oja. PicSOM-self-organizing image retrieval with MPEG-7 content descriptors. *IEEE International Conferences in Neural Networks*, 13:841–853, July 2002.
  - [45] J. Li, J. Z. Wang, and G. Wiederhold. IRM: integrated region matching for image retrieval. In *MULTIMEDIA '00: Proceedings of the eighth ACM international conference on Multimedia*, pages 147–156, New York, NY, USA, 2000. ACM Press.
  - [46] H. Lin, Y. Kao, S. Yen, and C. Wang. A study of shape-based image retrieval. In *ICDCSW '04: Proceedings of the 24th International Conference on Distributed Computing Systems Workshops - W7: EC (ICDCSW'04)*, pages 118–123, Washington, DC, USA, 2004. IEEE Computer Society.
  - [47] H. Liu. Evolving feature selection. *IEEE Intelligent Systems*, pages 64–76, Nov/Dec 2005.

- 
- [48] H. Liu and L. Yu. Toward integrating feature selection algorithms for classification and clustering. *IEEE Trans. on Knowledge and Data Engineering*, 17(4):491–502, 2005.
- [49] 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 *MULTIMEDIA '00: Proceedings of the eighth ACM international conference on Multimedia*, pages 31–37, New York, NY, USA, 2000. ACM Press.
- [50] J. Luo and M. A. Nascimento. Content based sub-image retrieval via hierarchical tree matching. In *MMDB '03: Proceedings of the 1st ACM international workshop on Multimedia databases*, pages 63–69, New York, NY, USA, 2003. ACM Press.
- [51] J. Luo and M. A. Nascimento. Content based sub-image retrieval via hierarchical tree matching. In *MMDB '03: Proceedings of the 1st ACM international workshop on Multimedia databases*, pages 63–69, New York, NY, USA, 2003. ACM Press.
- [52] W.Y. Ma and B.S. Manjunath. NETRA: A toolbox for navigating large image databases. *Multimedia Systems*, 7(3):184–198, May 1999.
- [53] B.S. Manjunath and W.Y. Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):837–842, 1996.
- [54] O. Maron and T. Lozano-Perez. *A Framework for Multiple Instance Learning. Advances in Natural Information Processing System 10*. MIT Press, 1998.
- [55] J. Martinet, Y. Chiaramella, and P. Mulhem. A model for weighting image objects in home photographs. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 760–767, New York, NY, USA, 2005. ACM Press.

- 
- [56] S. Mika, G. Ratsch, J. Weston, B. Scholkopf, and K. Muller. Fisher discriminant analysis with kernels. In IEEE, editor, *Proceedings of IEEE Neural Networks for Signal Processing Workshop*, pages 41–48. IEEE Press, 1999.
- [57] MPEG-7. MPEG-7 visual part of eXperimentation Model version 4.0. *ISO/IEC JTC1/SC29/WG11/N3068, Maui, Hawaii*, December 1999.
- [58] M. Nakazato and T. S. Huang. Extending image retrieval with group-oriented interface. *Multimedia and Expo, 2002*, 1:201–204, 2002.
- [59] M. Nakazato, L. Manola, and T. S. Huang. ImageGrouper: Search, annotate and organize images by groups. In *VISUAL '02: Proceedings of the 5th International Conference on Recent Advances in Visual Information Systems*, pages 129–142, London, UK, 2002. Springer-Verlag.
- [60] D. Neumann and K. R. Gegenfurtner. Image retrieval and perceptual similarity. *ACM Trans. Appl. Percept.*, 3(1):31–47, 2006.
- [61] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, P. Yanker, C. Faloutsos, and G. Taubin. The QBIC project: Querying images by content using colors, textures and shape. *Storage and Retrieval for Image and Video Databases*, 1908:173–187, February 1993.
- [62] H. Nishiyama, S. Kin, T. Yokoyama, and Y. Matsushita. An image retrieval system considering subjective perception. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 30–36, New York, NY, USA, 1994. ACM Press.
- [63] S. E. Palmer. Perceptual grouping: It's later than you think. *Current Directions in Psychological Science*, 11(3):101–106, June 2002.

- 
- [64] J.S. Payne and T.J. Stonham. Can texture and image content retrieval methods match human perception? In *Intelligent Multimedia, Video and Speech Processing*, pages 154–157, Hong Kong, China, 2001.
- [65] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 27(8):1226–1238, 2005.
- [66] A. Pentland, R. Picard, and S. Sclaroff. Photobook: Content-based manipulation of image databases. *International Journal of Computer Vision*, 18(3):233–254, June 1994.
- [67] A. Rao, R. K. Srihari, and Z. Zhang. Geometric histogram: A distribution of geometric configuration of color subsets. *Internet Imaging, Proc. of SPIE*, 3964:91–101, January 2000.
- [68] F. Ren. Multi-image query content-based image retrieval. Master’s thesis, School of Information Technology and Computer Science, University of Wollongong, 2006.
- [69] C. Roda and J. Thomas. Attention aware systems: Theories, applications and research agenda. *Computers in Human Behavior*, 22(4):557–587, 2006.
- [70] Y. Rui and T. Huang. Optimizing learning in image retrieval. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 01, pages 236–243, Los Alamitos, CA, USA, 2000. IEEE Computer Society.
- [71] Y. Rui, T. S. Huang, and S. Chang. Image retrieval: past, present, and future. In *Proc. of Int. Symposium on Multimedia Information Processing*, 1997.

- 
- [72] Y. Rui, T. S. Huang, M. Ortega, and S. Mehrotra. Relevance feedback: A power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits Systems for Video Technology*, 8(5):644–655, 1998.
- [73] S. K. Saha, A. K. Das, and B. Chanda. Image retrieval based on indexing and relevance feedback. *Pattern Recogn. Lett.*, 28(3):357–366, 2007.
- [74] S.K. Saha, A.K. Das, and B. Chanda. An efficient search technique for CBIR systems. *Proc. 3rd Internat. Workshop on Content Based Multimedia Indexing*, 3:151C156, September 2003.
- [75] J. R. Smith and S. Chang. VisualSEEk: a fully automated content-based image query system. In *MULTIMEDIA '96: Proceedings of the fourth ACM international conference on Multimedia*, pages 87–98, New York, NY, USA, 1996. ACM Press.
- [76] Y. Sun and S. Ozawa. A hierarchical approach for region-based image retrieval. *2004 IEEE International Conference*, 1(10-13):1117–1124, October 2004.
- [77] J. Tang and S. Acton. An image retrieval algorithm using multiple query images. *Signal Processing and Its Applications, 2003*, 1(1-4):193–196, July 2003.
- [78] A.M. Triesman. Features and objects: The fourteenth bartlett memorial lecture. *Quarterly Journal of Experimental Psychology*, 40A:201–237, 1988.
- [79] A. Tsymbal and M. Pechenizkiy abd P. Cunningham. Diversity in search strategies for ensemble feature selection. *Information fusion*, 6:83–98, 2005.
- [80] J. Z. Wang, G. Wiederhold, O. Firschein, and X. W. Sha. Content-based image indexing and searching using Daubechies' wavelets. *International Journal of Digital Libraries*, 1(4):311–328, 1998.



- 
- [81] P. Wilkins, P. Fergson, A. F. Smeaton, and C. Gurrin. Text based approaches for content-based image retrieval on large image collections. *EWIMT'05 London, U.K.*, pages 281–288, November 2005.
- [82] C. Zhang, J. Y. Chai, and R. Jin. User term feedback in interactive text-based image retrieval. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 51–58, New York, NY, USA, 2005. ACM Press.
- [83] C. Zhang, S.-C. Chen, M.-L. Shyu, and S. Peeta. Adaptive background learning for vehicle detection and spatio- temporal tracking. *Information, Communications and Signal Processing*, 2:797–801, December 2003.
- [84] C. Zhang and X. Chen. OCRS: an interactive object-based image clustering and retrieval system. In *MDM '05: Proceedings of the 6th international workshop on Multimedia data mining*, pages 71–78, New York, NY, USA, 2005. ACM Press.
- [85] R. Zhang, Z. Zhang, and J. Yao. A unified fuzzy feature indexing scheme for region based online image querying. *Web Intelligence, 2003.*, pages 421–424, October 2003.
- [86] X. S. Zhou and T. S. Huang. Comparing discriminating transformations and SVM for learning during multimedia retrieval. In *MULTIMEDIA '01: Proceedings of the ninth ACM international conference on Multimedia*, pages 137–146, New York, NY, USA, 2001. ACM Press.
- [87] X. S. Zhou and T. S. Huang. Relevance feedback in image retrieval: A comprehensive review. *Multimedia Systems*, 8(6):536–544, 2003.

- 
- [88] L. Zhu and A. Zhang. Supporting multi-example image queries in image databases. *IEEE International Conference on Multimedia and Expo (II)*, 2:697–700, 2000.