Content-based retrieval from image databases using sketched queries

Abdolah Chalechale

University of Wollongong

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:

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.

Recommended Citation


Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au
Content-Based Retrieval from Image Databases Using Sketched Queries

A thesis submitted in partial fulfilment of the requirements for the award of the degree

Doctor of Philosophy

from

THE UNIVERSITY OF WOLLONGONG

by

Abdolah Chalechale
Master of Engineering

SCHOOL OF ELECTRICAL, COMPUTER AND TELECOMMUNICATIONS ENGINEERING

MARCH 2005
Abstract

Searching multimedia databases using features extracted from the content is currently an active research area. This thesis presents novel feature extraction approaches for content-based image retrieval when the query image is a hand-drawn black and white sketch. To facilitate robust man-machine interfaces, we accept query images with no color and texture attributes. Special attention is given to the scale and rotation invariance properties since the query and database images may vary in size and rotation angle. Several applicable techniques within the literature are studied for these conditions.

The goal is to present the user with a subset of images that are more similar to the sketched query. New affine transform invariant feature extraction techniques are proposed to improve retrieval performance, and reduce the extraction and search times. The techniques are tested both generally for multi-component images and particularly for isolated shapes. The solutions are discussed for each specific application. Finally, signature-based document retrieval, which explores document retrieval from databases using human signatures, is investigated on an individual basis.

Two different approaches based on spatial distribution of edge pixels are proposed for general sketch-based image retrieval. Here, the database images consist of multiple complex objects within an inhomogeneous background. One of the methods is an improved version of another, which increases retrieval performance. Both techniques exhibit scale invariance property resulting from size normalization. Rotation invariance property is achieved by employing the magnitude of the Fourier transform coefficients in two different ways. A database including 4000 artwork and photograph images is used for the experiments.
One of the most important aspects of the proposed methods is that the image segmentation is not needed. This significantly improves the feature extraction process and enables the methods to be used for other computer vision applications. Edge image matching is studied as an example.

Sketch-based shape retrieval, which deals with images containing an isolated shape, is studied next. This is to emphasize on the existing differences between general images and isolated shapes for sketch-based retrieval. A new approach is proposed for this task, which outperforms alternative approaches. In this approach a chain code differentiation of contour shapes is applied for contour polygonization. The geometric properties of the resulting polygon are used to extract hybrid and efficient features.

A signature region extraction method is proposed as a preprocessing stage in signature-based document retrieval. Several feature extraction techniques are adapted and examined for this application and the results are discussed.

In all the experiments, the Average Normalized Modified Retrieval Rank (ANMRR), which was developed in the MPEG-7 standardization process, is used as the retrieval performance criterion. Feature extraction and search times are computed for time-based comparisons. The proposed methods exhibit significant improvements in retrieval accuracy. There is also marked improvement in feature extraction and search speed for the proposed methods.
Statement of Originality

This is to certify that the work described in this thesis is entirely my own, except where due reference is made in the text.

No work in this thesis has been submitted for a degree to any other university or institution.

Signed

Abdolah Chalechale
2 March, 2005
Acknowledgments

I would like to express my gratitude to my wife, Nargess and my two boys, Amir and Ali for their veritable patience in the duration of this PhD. Without their support non of this would have been possible.

Next, I wish to profoundly thank my supervisor, Associate Professor Golshah Naghdy for her academic advice and very helpful hints in doing the research and in writing the thesis. Her exceptional guidance made this work to be completed.

I would also like to thank my previous supervisor, Professor Alfred Mertins for initiating the topic and for his outstanding technical comments in the first two years. Even after he moved to the University of Oldenburg, Germany, we had precious communication regarding the work presented in this thesis.

I sincerely appreciate my co-supervisor Dr. Prashan Premaratne for his advice, friendly manner, encouragement and time devoted for this work.

I acknowledge Associate Professor Ian Burnett for providing access to the MPEG-7 image databases and documents used in this research. Thanks also go to the World Art Kiosk at the California State University for providing image databases for the research purposes.

I would like to acknowledge the Razi University, Kermanshah, Iran, which provided me a scholarship to pursue my studies at Ph.D. level.

Finally, I would like to thank all the people who made sketched images for the tests or gave permission to use their personal signatures in this study.
Contents

1 Introduction 1

1.1 Main Trends in Image Retrieval 1

1.2 Problem Statement 2

1.3 Goals of the Research 6

1.4 Thesis Organization 8

1.5 Contributions 10

1.6 Publications 11

2 Literature Review 15

2.1 Introduction 15

2.2 Visual Information Retrieval 16

2.3 Image Retrieval 17

2.3.1 Attribute-Based Retrieval 19

2.3.2 Annotation-Based Retrieval 20

2.3.3 Object Recognition-Based Retrieval 21

2.3.4 Low-Level Image Features 21

2.4 Content-Based Image Retrieval 22

2.4.1 Pixel Domain Techniques 25

2.4.2 Compressed Domain Techniques 38
## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.5 Different Queries</td>
<td>47</td>
</tr>
<tr>
<td>2.6 Similarity Measures</td>
<td>48</td>
</tr>
<tr>
<td>2.7 Retrieval Performance</td>
<td>54</td>
</tr>
<tr>
<td>2.7.1 Exact Match</td>
<td>55</td>
</tr>
<tr>
<td>2.7.2 Similar Matches</td>
<td>56</td>
</tr>
<tr>
<td>2.8 Semantics in Image Retrieval</td>
<td>58</td>
</tr>
<tr>
<td>2.9 Content-Based Image Retrieval Systems</td>
<td>62</td>
</tr>
<tr>
<td>2.9.1 General Structure</td>
<td>63</td>
</tr>
<tr>
<td>2.9.2 Representative Systems</td>
<td>65</td>
</tr>
<tr>
<td>2.10 Chapter Summary and Conclusion</td>
<td>73</td>
</tr>
<tr>
<td>3 Sketch-Based Image Retrieval (SBIR)- Applicable Methods</td>
<td>74</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>74</td>
</tr>
<tr>
<td>3.2 Query by Visual Example (QVE)</td>
<td>75</td>
</tr>
<tr>
<td>3.2.1 Abstraction Algorithm</td>
<td>76</td>
</tr>
<tr>
<td>3.2.2 Retrieval Algorithm</td>
<td>77</td>
</tr>
<tr>
<td>3.3 Histogram of Edge Directions (HED)</td>
<td>78</td>
</tr>
<tr>
<td>3.4 Invariant Moments</td>
<td>80</td>
</tr>
<tr>
<td>3.5 MPEG-7 Edge Histogram Descriptor (EHD)</td>
<td>82</td>
</tr>
<tr>
<td>3.6 Angular Radial Transform (ART)</td>
<td>87</td>
</tr>
<tr>
<td>3.6.1 ART Extraction Algorithm</td>
<td>88</td>
</tr>
<tr>
<td>3.7 Polar Fourier Descriptors (PFD)</td>
<td>89</td>
</tr>
<tr>
<td>3.7.1 Polar Fourier Transform Extraction Algorithm</td>
<td>90</td>
</tr>
<tr>
<td>3.8 Chapter Summary and Conclusion</td>
<td>91</td>
</tr>
</tbody>
</table>
CONTENTS

4 SBIR Using Image Abstraction and Angular Partitioning ................................................. 92

  4.1 Introduction ................................................................................................................. 92
  4.2 Background .................................................................................................................. 93
  4.3 Angular Partitioning of Abstract Image (APAI) .......................................................... 94
    4.3.1 Statistical Image Abstraction ................................................................................ 96
    4.3.2 Angular Partitioning (AP) .................................................................................... 99
  4.4 Experimental Results and Efficacy Evaluation .............................................................. 102
    4.4.1 Similarity Measurement and Retrieval Performance Criteria .............................. 102
    4.4.2 The Effect of Parameter Variations ...................................................................... 105
    4.4.3 Comparative Results ............................................................................................ 108
  4.5 Chapter Summary and Conclusion ................................................................................ 113

5 Advanced Features Using Angular Radial Partitioning .................................................... 116

  5.1 Introduction ................................................................................................................. 116
  5.2 Background .................................................................................................................. 117
  5.3 Angular Radial Partitioning (ARP) ................................................................................ 118
  5.4 Experimental Results, Comparison and Discussion ...................................................... 122
    5.4.1 The ARP Methods for Sketch-Based Image Retrieval .................................... 122
    5.4.2 The ARP Method for Edge Image Matching ..................................................... 125
  5.5 Chapter Summary and Conclusion ................................................................................ 131

6 Sketch-Based Shape Retrieval Using Line Segment Distribution ....................................... 133

  6.1 Introduction ................................................................................................................. 133
  6.2 Background .................................................................................................................. 134
  6.3 Contour Polygonization Using Chain Code Differentiation (CPCD) ......................... 136
    6.3.1 A Hybrid Shape Similarity Measure ...................................................................... 139
CONTENTS

6.4 Experimental Results ............................................ 141

6.4.1 Discussion .................................................. 145

6.5 Chapter Summary and Conclusion ............................. 147

7 Case Study: Signature-Based Document Retrieval 148

7.1 Introduction .................................................. 148

7.2 Background .................................................. 149

7.3 Document Image Retrieval Using Signature Image ......... 150

7.3.1 Signature Region Extraction ............................ 151

7.4 Feature Extraction and Comparative Results ............... 153

7.5 Chapter Summary and Conclusion ............................. 158

8 Summary, Conclusion and Further Work 159

8.1 Introduction .................................................. 159

8.2 Thesis Summary and Conclusion ............................. 160

8.3 Further Work ................................................ 163

Bibliography .................................................. 167
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Image retrieval approaches</td>
<td>18</td>
</tr>
<tr>
<td>2.2</td>
<td>Content-based image retrieval techniques</td>
<td>25</td>
</tr>
<tr>
<td>2.3</td>
<td>An example of two different images with similar color histograms [42]</td>
<td>27</td>
</tr>
<tr>
<td>2.4</td>
<td>Using 2D strings to represent spatial relationships; (a) an image example, containing three objects, and (b) the corresponding symbolic picture</td>
<td>34</td>
</tr>
<tr>
<td>2.5</td>
<td>Different contours for constant metrics in 2D space; (a) Chebychev or $\ell_\infty$, (b) Manhattan or $\ell_1$, (c) Euclidean or $\ell_2$, and (d) Mahalanobis</td>
<td>50</td>
</tr>
<tr>
<td>2.6</td>
<td>General structure of image retrieval systems</td>
<td>64</td>
</tr>
<tr>
<td>2.7</td>
<td>(a) A typical VisualSEEk query interface page, and (b) fifteen top results for “lion” keyword, retrieved by the WebSEEk system</td>
<td>67</td>
</tr>
<tr>
<td>2.8</td>
<td>An example of MARS retrieval result, the upper-left shape is the input query</td>
<td>68</td>
</tr>
<tr>
<td>2.9</td>
<td>Two examples of the Photobook system retrieved images; (a) search by texture, and (b) search by shape; the query images are those in the upper-left in each case</td>
<td>70</td>
</tr>
<tr>
<td>2.10</td>
<td>An example of Blobworld query result</td>
<td>72</td>
</tr>
<tr>
<td>3.1</td>
<td>Query by visual example using local and global correlation [9]</td>
<td>78</td>
</tr>
<tr>
<td>3.2</td>
<td>Definition of sub-image and image-block [155]</td>
<td>83</td>
</tr>
<tr>
<td>3.3</td>
<td>Sub-blocks and their labelling scheme [63]</td>
<td>84</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3.4</td>
<td>Five different edges and corresponding filter coefficients for edge detection: (a) vertical, (b) horizontal, (c) 45° diagonal, (d) 135° diagonal, and (e) isotropic (non-directional) [63, 155]</td>
<td>85</td>
</tr>
<tr>
<td>3.5</td>
<td>An edge classification flowchart without using the isotropic edge-filter for computing the MPEG-7’s EHD descriptors [155]</td>
<td>86</td>
</tr>
<tr>
<td>3.6</td>
<td>MPEG-7 edge histogram descriptor (a) thirteen clusters of sub-images for semi-global histogram, and (b) overall histogram structure [63]</td>
<td>87</td>
</tr>
<tr>
<td>4.1</td>
<td>Two procedures for image abstraction; (a) statistical approach for full-color images, and (b) morphological thinning for hand-drawn sketches</td>
<td>95</td>
</tr>
<tr>
<td>4.2</td>
<td>Edge points distribution for (a) three different images, (b) corresponding histograms, and (c) corresponding cumulative histograms</td>
<td>97</td>
</tr>
<tr>
<td>4.3</td>
<td>The effect of parameter $\beta$ on edge maps.</td>
<td>98</td>
</tr>
<tr>
<td>4.4</td>
<td>(a) A sketched image, and (b) morphologically thinned and isolated pixels removed (black pixels correspond to value 1 and white pixels correspond to value 0 in the image matrix)</td>
<td>99</td>
</tr>
<tr>
<td>4.5</td>
<td>Angular partitioning splits the abstract image into $K$ successive slices.</td>
<td>100</td>
</tr>
<tr>
<td>4.6</td>
<td>Bounding box of the abstract image $P$ or $Q$ is size normalized and supplied to the angular partitioning process. The final result is an associated feature vector $\Psi$ used for similarity measurement</td>
<td>101</td>
</tr>
<tr>
<td>4.7</td>
<td>(a) An image example, (f) the 90° rotated version, (b,g) corresponding abstract images before size normalization, (c,h) abstract images superimposed with angular partitions, (d,i) number of pixels in different slices, and (e,j) invariant features extracted using the Fourier transform, respectively.</td>
<td>102</td>
</tr>
<tr>
<td>4.8</td>
<td>The ANMRR (Average Normalized Modified Retrieval Rank) measure of the APAI method with (a) $\beta = 1$, (b) $\beta = 2$, and (c) $\beta = 3$ for a varying number of slices and four different normalized sizes.</td>
<td>107</td>
</tr>
<tr>
<td>4.9</td>
<td>The ROC curve of the APAI method showing the true positive ratio (TPR) and the false positive ratio (FPR) for different thresholds. The threshold (Thr) is applied on the normalized distance $d_n(P,Q)$.</td>
<td>108</td>
</tr>
<tr>
<td>4.10</td>
<td>Retrieval example 1: The top 16 retrieved images, in row order, when posing a sketched image (upper-left) with 8 similar images in the ground truth, NMRR=0.2400.</td>
<td>111</td>
</tr>
</tbody>
</table>
4.11 Retrieval example 2: The top 16 retrieved images, in row order, when posing a sketched image (upper-left) with 8 similar images in the ground truth, NMRR=0.3600.

4.12 Retrieval example 3: The top 16 retrieved images, in row order, when posing a sketched image (upper-left) with 8 similar images in the ground truth, NMRR=0.4200.

4.13 The ANMRR (Average Normalized Modified Retrieval Rank) measure of different approaches based on (a) rotation invariance property, and (b) scale invariance property. The abbreviations stand for the following approaches: angular partitioning of abstract image (APAI), angular radial transform (ART), polar Fourier descriptors (PFD), Zernike moment invariants (ZMI), histogram of edge directions (HED), MPEG-7’s edge histogram descriptor (EHD), and query by visual example (QVE) respectively. The lower the ANMRR the better the performance.

4.14 The feature extraction time (FET) and the search time (SET) for different methods. The vertical axis is discontinued, which shows that the search time for the QVE method is more than 10 times longer than for the other methods.

5.1 Angular Radial Partitioning of an image to $N$ angular and $M$ radial sectors where $k = 0, 1, 2 \ldots M - 1$ and $i = 0, 1, 2 \ldots N - 1$.

5.2 (a) An image example, (b) its $90^\circ$ rotated version, (c,d) corresponding edge images superimposed with angular-radial partitions, (e,f) number of pixels in the individual partitions, and (g,h) invariant features extracted using the Fourier transform.

5.3 The Manhattan ($\ell_1$) distance calculated from all database images using the ARP method for an arbitrary input query.

5.4 Using APR method; the top 16 retrieved images, in row order, when posing a sketched image (upper-left) with 8 similar images in the ground truth, NMRR=0.2.

5.5 Image examples: the first column shows the original images and the next columns are the variants (rotated, small translated and eroded).

5.6 The Manhattan ($\ell_1$) distance using the ARP method to the database images for (a) the $q_{10}$, and (b) the $q_{52}$ query examples.

5.7 Retrieval results of different methods with the ANMRR.
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.8</td>
<td>The feature extraction time (FET) and search time (SET) for the angular partitioning (AP) and angular radial partitioning (ARP) methods</td>
<td>131</td>
</tr>
<tr>
<td>6.1</td>
<td>(a) An example thinned hand-drawn sketch, (b) the corresponding chain code, (c) median filtered code, (d) shifted code, (e) smoothed representation, (f) derivative, and (g) extracted line segments</td>
<td>138</td>
</tr>
<tr>
<td>6.2</td>
<td>The effect of variable $\tau$: (a) an original database image, (b) the contour, (c) polygonized contour with $\tau = 0.1$, NoS=112, (d) $\tau = 0.3$, NoS=64, (e) $\tau = 0.4$, NoS=41, (f) $\tau = 0.5$, NoS=18, (g) $\tau = 0.7$, NoS=8, (h) $\tau = 0.9$, NoS=3, (i) $\tau = 1$, NoS=2</td>
<td>139</td>
</tr>
<tr>
<td>6.3</td>
<td>Shapes from 70 different classes of the MPEG-7, CE-1 database</td>
<td>142</td>
</tr>
<tr>
<td>6.4</td>
<td>Examples of variations within the deer class</td>
<td>143</td>
</tr>
<tr>
<td>6.5</td>
<td>Examples of sketched shapes</td>
<td>144</td>
</tr>
<tr>
<td>6.6</td>
<td>Retrieval performance of different methods expressed by ANMRR</td>
<td>145</td>
</tr>
<tr>
<td>6.7</td>
<td>Average feature extraction time for different methods using both database and sketched shapes on a Pentium-III, 1000 MHz machine</td>
<td>146</td>
</tr>
<tr>
<td>7.1</td>
<td>Signature-based document retrieval structure</td>
<td>151</td>
</tr>
<tr>
<td>7.2</td>
<td>Examples of document images</td>
<td>154</td>
</tr>
<tr>
<td>7.3</td>
<td>(a) The ANMRR and (b) the FET parameters in descending order for different methods in the case of signature-based document retrieval</td>
<td>157</td>
</tr>
</tbody>
</table>
## List of Tables

5.1 The ANMRR of the ARP method with 3 radial and varying angular partitions using the movie derived database (4320 images) .................................................. 127

5.2 The ANMRR of the ARP method with 12 angular and varying radial partitions using the movie derived database (4320 images) ........................................... 128

5.3 The ANMRR of the ARP method using selected data sets (2160, 1440, 720 and 480 images) ................................................................. 128

7.1 The ANMRR and FET of the AP method using different number of slices ...................................................................................... 155

7.2 The ANMRR and FET of the ARP method using different number of angular radial partitions .......................................................... 155

7.3 The ANMRR and FET of the CPCD method using different resolutions for length and angle distributions ........................................ 156
List of Abbreviations

ANMRR  Average Normalized Modified Retrieval Rank
ANN    Artificial Neural Networks
AP     Angular Partitioning
ARP    Angular Radial Partitioning
AUC    Area Under the Curve
CBIR   Content-Based Image Retrieval
CLD    Color Layout Descriptor
COM    Center Of Mass
CPCD   Contour Polygonization using Chain code Differentiation
CSD    Color Structure Descriptor
CSS    Curvature Scale Space
DBMS   Data Base Management System
DCT    Discrete Cosine Transform
DFT    Discrete Fourier Transform
DROC   Differential Receiver Operation Characteristic
DWR    Department of Water Resources
DWT    Discrete Wavelet Transform
FEM    Finite Element Method
FET    Feature Extraction Time
FP     False Positive
EHD    Edge Histogram Descriptor
EPNH   Edge Pixel Neighborhood Histogram
EPNI   Edge Pixel Neighborhood Information
ETM    Elastic Template Matching
FD     Fourier Descriptors
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>HED</td>
<td>Histograms of Edge Directions</td>
</tr>
<tr>
<td>HSV</td>
<td>Hue Saturation Value</td>
</tr>
<tr>
<td>HTD</td>
<td>Homogeneous Texture Descriptor</td>
</tr>
<tr>
<td>IBM</td>
<td>International Business Machines</td>
</tr>
<tr>
<td>IR</td>
<td>Information Retrieval</td>
</tr>
<tr>
<td>JPEG</td>
<td>Joint Photographic Experts Group</td>
</tr>
<tr>
<td>KLT</td>
<td>Karhunen-Loeve Transform</td>
</tr>
<tr>
<td>LAB</td>
<td>Lightness-A-B (color model)</td>
</tr>
<tr>
<td>MPEG</td>
<td>Moving Pictures Expert Groups</td>
</tr>
<tr>
<td>NMRR</td>
<td>Normalized Modified Retrieval Rank</td>
</tr>
<tr>
<td>NoS</td>
<td>Number of Segments</td>
</tr>
<tr>
<td>PCA</td>
<td>Principal Component Analysis</td>
</tr>
<tr>
<td>PFD</td>
<td>Polar Fourier Descriptors</td>
</tr>
<tr>
<td>PFT</td>
<td>Polar Fourier Transform</td>
</tr>
<tr>
<td>PSQL</td>
<td>Pictorial Structural Query Language</td>
</tr>
<tr>
<td>QBE</td>
<td>Query By Example</td>
</tr>
<tr>
<td>QBIC</td>
<td>Query By Image Content</td>
</tr>
<tr>
<td>QPE</td>
<td>Query by Pictorial Example</td>
</tr>
<tr>
<td>QVE</td>
<td>Query by Visual Example</td>
</tr>
<tr>
<td>ROC</td>
<td>Receiver Operation Characteristic</td>
</tr>
<tr>
<td>ROI</td>
<td>Region Of Interest</td>
</tr>
<tr>
<td>SBIR</td>
<td>Sketch-Based Image Retrieval</td>
</tr>
<tr>
<td>SCD</td>
<td>Scalable Color Descriptor</td>
</tr>
<tr>
<td>SET</td>
<td>Search Time</td>
</tr>
<tr>
<td>SQL</td>
<td>Structural Query Language</td>
</tr>
<tr>
<td>SWP</td>
<td>State Water Project</td>
</tr>
<tr>
<td>TP</td>
<td>True Positive</td>
</tr>
<tr>
<td>UI</td>
<td>User Interface</td>
</tr>
<tr>
<td>VI</td>
<td>Visual Information</td>
</tr>
<tr>
<td>VIR</td>
<td>Visual Information Retrieval</td>
</tr>
<tr>
<td>VQ</td>
<td>Vector Quantization</td>
</tr>
<tr>
<td>ZMI</td>
<td>Zernike Moment Invariants</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Main Trends in Image Retrieval

The rapid expansion and ease of acquisition of images in digital form have resulted in the availability of extensive image databases. Swift and effective utilization and manipulation of these databases depend greatly on the intuitive user interfaces and algorithms facilitating search, matching and retrieval. Currently, there are two main trends in image retrieval: text-based and content-based.

Text and alphanumeric symbols are used to describe images in the text-based approach. The main advantage of this approach is the potential use of data modelling, multidimensional indexing and query evaluation methods, which have matured in the text-based databases. Consequently, the speed of search is improved by exploiting such techniques. Furthermore, visual databases can be accessed using standard query languages such as the SQL.

However, there are some major difficulties in the text-based approach that have led the researchers to pay more attention to the alternative content-based approach. First, according to the adage “an image is worth thousands words”, the rich content of images makes the annotation to be a conceptually difficult task. Describing, for example, a landscape or a country map by keywords is hard (if not impossible) and the underlying visual meaning could not be conveyed completely. Another problem is that manual annotation is a toilsome and laborious task specifically in large-scale
databases (i.e. tens to hundreds of thousands images).

Furthermore, different people may perceive a specific image differently. Human perceptual judgment is essential for many applications and textual approaches may cause unexpected and unrecoverable mismatches in the final retrieval stage, because language mismatch can occur when the user and the domain expert use different keywords. If the image database is to be shared globally, then linguistic barriers will render the use of keywords ineffective. Extra storage and large amount of manual processing make the text-based approach impractical and obsolete for large-scale image databases [1]. Many of the limitations of this approach are overcome by employing the content-based approach. Nevertheless, as the text-based approach has the ability to describe high-level concepts such as “laugh” or “vehicle”, some commercial content-based image retrieval systems are still offering this technique as an option for image querying.

The term “content-based image retrieval” (CBIR) is used when the image content is utilized for indexing and retrieval. Object recognition methods and the low-level features (i.e. color, texture, and shape) are used for automatic feature extraction in this technique. Moreover, middle-level features including spatial layout and sketch, are also considered in this approach. Due to the aforementioned difficulties in the text-based technique and the emergence of large-scale image databases, the content-based approach was proposed in the early 1990’s and research interests in this area have grown rapidly. Since then, the advances in this topic are mainly contributed to the image processing and computer vision areas. There has been some promising progress in this field, and many commercial image retrieval systems have been built based on this approach. To date, it remains an active research topic.

1.2 Problem Statement

Providing an image example as the input query is a well known approach in the content-based approach. The way in which an input query is submitted has a great influence on the overall efficiency and effectiveness of the system. There exist several techniques to submit the query. Most often, the query image is an ordinary full-color
image provided by the user. Sometimes, it could be a selected image from an existing album provided by the system. The query image can be generated using primitives such as circles and rectangles. It may be black and white or colored. Another way to submit a query is to provide a hand-drawn sketch via a scanner or a computerized tool. This latter approach is referred to as the *sketch-based image retrieval* (SBIR).

Each querying technique has advantages and disadvantages due to the degree of user satisfaction, accuracy and speed that should be compromised based on the application at hand. It might be simple to let the user choose from a set of images already in the database the ones that more closely match the image to be retrieved. This would allow pre-computation on these images and feature extraction off-line, which speeds up the query time. This technique, unfortunately, limits the users’ versatility. If full freedom is to be given to the users, the only practical mechanism for a query input is a sketch. A sketch consisting of black and white is a more versatile way of letting the users express their mental picture of desired image. Although more attention has been given to the general problem of the CBIR in the recent years, there are very few approaches centered around the SBIR. Furthermore, those dealing with sketched queries mostly use color and texture attributes for feature extraction.

Having had the query, the system will search the database and retrieve most similar images based on the adapted features. Many existing systems do image retrieval using the image content. Most of these systems, however, employ features based on color and texture attributes. Some do not exhibit rotation or scale invariance or they inherently rely on image segmentation as a priori. A significant drawback in some other approaches is lack of indexing. This arises from calculating the similarity measure by a correlation scheme which overlays two images at the search time. Thus, reducing the speed of the retrieval task remarkably.

In the following we highlight the most important aspects of an efficient approach for the CBIR in general and for the SBIR in particular.

1. **Ease of query interface:** this makes the approach more attractive and convenient for not only a general but also a specific user. Moreover, implementing the approach on the web needs an easy, fast and simple but effective user inter-
face.

2. **Provide indexing:** an approach which supports indexing leads to efficient memory usage since the indexes are always smaller than images themselves. Indexing speeds up the search process significantly which is profoundly influential in the on-line and web-based applications.

3. **Need for segmentation:** it is always desirable (if possible) to avoid image segmentation since it is, by and large, computationally intensive. Segmentation-free approaches are preferred in image retrieval due to ease of operation and higher speed. Although the image segmentation may improve the overall performance in some applications, the existing limitations in the current segmentation methods make them applicable for some specific areas rather than more general and wide-range utilizations.

4. **Scale and rotation invariance:** a good approach should be invariant to the scale and rotation angle. The query image provided in the CBIR and in the SBIR often has a different size compared to the size of the similar image(s) in the database. The system is required to be scale invariant and consequently able to find conceptually similar images regardless of the size differences. In addition, the query image may be posed in a different angle compared to the angle of the target image(s) in the database. Rotation and scale invariance properties enforce the ability of finding similar images regardless of the angle and size characteristics.

5. **Feature extraction time (off-line):** the time spent to extract features from the images in the database is an important off-line criterion. In spite of the fact that the off-line process is carried out at the time of database population, the feature extraction time (FET) should still be kept low. It comes into consideration whenever the database is updated or reorganized. The FET parameter is significant in the on-line process as well. This belongs to the feature extraction procedure applied on the query image during the on-line phase. For these reasons adequate attention needs to be given to the FET parameter in both CBIR and SBIR design.
6. **Search time (on-line):** the time consumed to generate the results and show them to the user in the on-line process of any retrieval system is dependent on the feature extraction, searching algorithm, and the indexing structure that have been used. Obviously, the shorter the time is, the better would be the user satisfaction. The search time’s (SET) dependance on the feature extraction process is usually negligible. This is due to the fact that only the query feature extraction is computed for the SET parameter while for the FET the feature extraction times of all images in the database are added up. The most important factors in the SET is the dimension of the selected features (i.e. the length of the feature vector), the similarity measure used for comparing the features and finally the indexing structure employed to organize the database.

7. **Retrieval performance:** the ultimate goal of image retrieval systems (CBIR and SBIR) is to present the user with a subset of images from those in the database that are conceptually and/or visually similar to the input query. The retrieval performance can be considered from two different perspectives: *efficacy* which reflects the speed of the retrieval task, and *effectiveness* which refers to correctness of the retrieved images. The FET and SET criteria express the *efficiency*. There are several quantitative parameters with quasi similar basis in the literature to declare the *effectiveness*. The criterion employed for the effectiveness of the retrieval task not only has to consider the number of correct images in the presented subset of images but also it should acknowledge the rank of the similar images in the output list.

This thesis addresses the problem of content-based retrieval from image databases containing full-color and textured images using hand-drawn sketches as queries. Here, the query is a simple black and white sketch of the desired shape, image, or scene similar to the one sought. To promote ease of use, the queries have no color and texture attributes. Moreover, special attention is devoted to the scale and rotation invariance properties. These are crucial issues since the input query usually is provided with a different size and rotation angle compared to the database images. The proposed techniques support indexing which makes them useful for large-scale image databases. The indexing is accomplished by the extraction of efficient features
indexed as the representatives of the original images. This enables us to search for similar images in a compressed data rather than uncompressed images. The existing similarity between images is measured using the corresponding indexes.

### 1.3 Goals of the Research

The main idea of this project is to provide effective, fast, and accurate CBIR using easy and robust man-machine interface by means of sketches. In other words the aim is to find content-based image features that are effective enough to enable us to do sketch-based image retrieval where a user-drawn sketch is applied as the starting point to search in an image database for conceptually similar item(s). This is a non-trivial task since users usually do not have a good example at hand when they begin to search for images. The combination of the following goals/aspects of this work sets it apart from other image retrieval tools:

- **Greater user versatility:** using a simple hand-drawn sketch as the input query which allows to search in a database of normal images containing color and texture attributes is a unique feature of this work. Here, we put no restriction on the images in the database. They are ordinary images with inhomogeneous background consisting of one or several objects such as an art painting or an outdoor photograph. The mechanism by which we do search in such a database is using a simple sketch containing the outline of the main objects in the image.

- **Avoiding image segmentation:** image segmentation to extract objects is a pre-processing step in many CBIR systems. It forces a computationally intensive task which usually takes place with help of an operator. Full-automatic segmentation is still an open problem in computer vision. Current methods work in an specific area which restricts their use in general applications. We propose methods that do not force the images to be segmented for object extraction. The features are extracted from the whole image and the cost of segmentation is saved.
• **Scale and rotation invariance:** the input query is often supplied with a size greater or smaller than the similar one(s) in the database. Beside, the query image may pose in a different or even opposite direction compared to the images in the database. This is a frequent event when a scanner is used to supply the query. Thus, special attention is given to the scale and rotation invariance properties in this project. The extracted features need to be scale and rotation invariant in the case of general multi-component images, and have translation invariance property as well in the case of isolated shape retrieval (one of the topics studied apart).

• **Indexing capability:** we look for compact and efficient features to serve as an index for an image. The indexes should be as close to identical as possible for similar images irrespective of color and texture attributes and regardless of the scale and rotation angles. The extracted features (index) are used to search in a compressed data which are the representatives of the images in the database. This makes the retrieval task efficient and fast. Indexing ability of the proposed methods ensures their suitability in large-scale image databases.

Three main categories in the sketch-based image retrieval (SBIR) have been considered in this thesis. First, searching in a database containing full-color and textured images using hand-drawn black and white sketches which are consisting of main objects in a nearly right positions. This is, for example, useful when we search in an art gallery for a specific painting or look for a desirable landscape in a photo album. Searching for a given frame in a video sequence is another application of this category. We have exploited a database created from art works and photographs (Chapter 4) and another database containing video frames (Chapter 5) for these applications. Second, we study isolated object retrieval using sketched shapes. This is to investigate object-based matching using user sketches and to compare different techniques in both global and local searches. The MPEG-7 contour shape database is used for this category. The third category is related to retrieving document images using the human signature within the image. Signature region detection and feature extraction procedures are discussed for this application. A document image database containing images of mixed text, logo, headline, and cursive signature has been employed for
the conducted tests.

In each of the aforementioned categories the goal is to find the most appropriate image features first, and then to extract the features efficiently. The extracted features are used as an index for the database search. A suitable test bed has been built for each application to conduct authentic experiments as best as possible.

The thesis presents several experiments conducted to evaluate the efficiency and effectiveness of the proposed methods. The experiments are built around the general sketch-based image retrieval, edge image description, sketch-based shape retrieval, and signature-based document retrieval. In each case, we have implemented our proposed techniques and some other well known approaches for comparison. The results have been derived in tantamount circumstances for all methods. The experimental results confirm the robustness and efficacy of the proposed methods.

Although our goals of the thesis do not include building a complete image retrieval system, we have designed many necessary components of an image retrieval system. Making an end-to-end system requires some more consideration beyond the goals of this thesis including graphical user interface (GUI). Moreover, several other algorithms need to be included for completeness of the usage, which in fact, by and large, is a commercial task. The main focus in this thesis is placed on proposing sketch-based image retrieval algorithms, which can be used efficiently and effectively within different systems with different goals in different circumstances.

1.4 Thesis Organization

The chapters that comprise the thesis are organized as follows. Most chapters include an introductory section. A brief background is presented next, and a chapter summary is given at the end. The main ideas of the chapters are explained below.

- Chapter 1 introduces the research topic, its main goals, and provides contributions and publications related to the thesis.
• **Chapter 2** is a wide-range literature review of the key concepts and outstanding works in the areas of image retrieval, content-based indexing, image querying, and similarity and retrieval performance measuring. Semantics in image retrieval is briefly discussed in this chapter. Several existing content-based image retrieval systems are further discussed. Readers interested only in SBIR can skip this chapter as it contains general concepts of CBIR.

• **Chapter 3** presents six different approaches within the literature which can be used directly or can be adapted for the SBIR. The main concept of each approach and associated algorithms are provided in this chapter with more details.

• **Chapter 4** introduces a novel approach for image feature extraction. The approach is based on image abstraction and spatial distribution of edge pixels in the abstract images. Theoretical basis is explained mathematically and invariance properties are discussed. The extracted features are scale and rotation invariant and robust against translations. A database containing paintings and photographs is created using images in the World Art Kiosk at the California State University and set S3 of the MPEG-7 database. The proposed approach creates a kind of ambient intelligence in terms of the evaluation non-precise easy to input sketched information. The proposed method is compared to six other approaches within the literature.

• **Chapter 5** improves the approach proposed in Chapter 4 by applying a more reliable spatial partitioning on the abstract images. Although the feature extraction scheme is slightly slower, the retrieval performance is significantly improved. As another application of the proposed method, edge image description is investigated in particular. Searching in a sequence of movie frames using an edge map of a given frame is examined in this chapter. The database which has been used for this new application is generated from "Animals have young" movie of the MPEG-7 content set V14.

• **Chapter 6** presents the concept of shape-based image retrieval. This is to evaluate existing techniques for isolated shape retrieval using hand-drawn shapes.
A new technique based on chain code differentiation is proposed for feature extraction. The new features result in better retrieval performance due to scarcity of edge pixels in this case. The MPEG-7 contour shape database CE-1 is used for the tests. Four other well known methods within the literature are also included in the experiments.

- **Chapter 7** is a case study of signature-based document retrieval. A special case of sketch-based image retrieval is introduced, which is dealing with document images. Each document is an official or business letter consisting of a cursive signature together with normal text. The aim is to retrieve those documents having the signature similar to the query signature. The signature region is determined first, and then the features are extracted for the retrieval task. Connected component analysis and labelling are employed for the signature region extraction. Four different feature extraction approaches are evaluated using a document image data set containing 425 documents.

- **Chapter 8** summarizes the thesis and draws the conclusions. Some new directions for further work are alleged in this final chapter.

### 1.5 Contributions

The main contributions of this thesis are:

1. A similarity measure for comparing image features extracted from edge pixel neighborhood histograms is proposed. This measure is used for ranking images in a database for the retrieval task.

2. A retrieval performance criterion for the case of exact match is proposed. This technique is based on a scoring scheme which enables us to evaluate different retrieval algorithms. Each retrieving algorithm is assigned with a percentage value showing its ability to retrieve the requested item. The proposed criterion is not restricted to image retrieval and can be used in evaluation of any ranking algorithm.
3. A dynamic image abstraction algorithm, which uses statistical distribution of edge pixels is suggested for full-color and textured images. The resulting image consists of strong edges and is comparable to the morphologically thinned sketched image.

4. Angular partitioning (AP) technique, which is based on spatial distribution of strong edges in the image slices is proposed and used for feature extraction. Applying the 1D Fourier transform on the resulting values yields a novel features which are scale and rotation invariant and robust against translations.

5. Angular radial partitioning (ARP) method which is an enhanced version of the AP technique is proposed. Here, strong edge points are overlayed with several concentric sectors in the image plane. Applying multiple 1D Fourier transforms on concentric sectors generates rotation invariance while size normalization is used to gain scale invariance.

6. A novel chain code differentiation approach is proposed for contour shape representation. This generates a set of line segments which efficiently describes a planar curve. Geometric properties of the set are used to extract hybrid features. The features are employed for the shape-based retrieval.

7. A signature region detection method based upon connected component analysis and labelling is proposed. The method uses heuristic rules to distinguish the region of interest, that is, where the cursive signature in a document image resides. This region is then used for feature extraction and the extracted features are used for the signature-based document retrieval.

1.6 Publications

The following publications have been the result of the research presented in this thesis:

**Journal papers and book chapters**

1. A. Chalechale, G. Naghdy, and A. Mertins, “Sketch-based image matching


### Peer reviewed conference papers


16. A. Chalechale, G. Naghdy, and A. Mertins, “Visual investigation using circular partitioning of abstract images,” in Proc. 7th Australian Pattern Recognition...


Chapter 2

Literature Review

2.1 Introduction

Visual information processing is becoming increasingly important with the advent of broadband networks, high power workstations, and advanced imaging tools including digital cameras and scanners. Managing huge number of images and video clips in multimedia databases and on the web needs more efficient organization for fast navigation and retrieval. There are several application domains including: remote sensing, telemedicine, interactive television, authentication, virtual university, military and art, involved in image and video indexing, search, and retrieval. Because of the large amount of memory and computational intensity needed in visual data storing and processing, enhanced methods and tools are required to manage multimedia databases more efficiently.

This chapter provides a survey of the most important issues in the current literature in image and video indexing and retrieval. Section 2.2 presents major similarities and differences between two kind of visual information, i.e. image and video. Different approaches in image retrieval are reviewed in Section 2.3, and Section 2.4 explains content-based image retrieval (CBIR) techniques both in pixel and compressed domains. This is followed by a discussion of different types of queries in Section 2.5. Sections 2.6 and 2.7 present similarity measures and retrieval performance issues, respectively. Semantics issues in image retrieval are addressed in Section 2.8 and...
some representative image retrieval systems are explored in Section 2.9. Finally, a chapter summary is given in Section 2.10.

2.2 Visual Information Retrieval

Visual information retrieval (VIR) is a methodology that searches and retrieves images and videos in databases. Video is primarily defined as a series of individual still images and the associated audio. In other words, Video, in an elementary view, is a long sequence of still images, ignoring the audio attribute. Therefore, image retrieval can be considered more fundamental to video retrieval as images are the building blocks of videos. In another aspect, video retrieval could be considered simpler than image retrieval since video discloses its objects more easily as the points, corresponding to one object, move together [2].

In still pictures, the creator’s narrative expression on intention is in scene selection, illumination, and composition of objects. Moreover, video has a linear time-line, which is as important to the narrative structure of video as it is in text. On the other hand, the image does not have such attributes and it expresses all information in only one frame. Thus, methods developed in image retrieval could be extended to video retrieval. Consequently, more research is being carried on image retrieval than video retrieval [2–4].

Image and video retrieval fall into the category of visual information retrieval (VIR) which itself is the successor to the information retrieval (IR). Although traditional IR models and methods could be adapted for use in the VIR, it is more reasonable to propose new exclusive approaches for image and video.

This thesis concentrates on image retrieval but, the proposed methods can be adapted to video retrieval as well. An image databases extracted from a movie has been used in Chapter 5. The subsequent sections review the most important concepts related directly to image retrieval.
2.3 Image Retrieval

Image database is one of the centerpieces for distributed multimedia systems. In general, an image database system is an intelligently combination of the following three major components:

- image processing
- storage, retrieval, and management
- user interface

The first component deals with processing for the extraction of information from original images whilst the second component is providing efficient tools for storing, retrieval, and managing of image data. Querying the database needs a simple and user friendly interface which is the responsibility of the third component.

Image retrieval has been influenced by the language support from conventional database management systems (DBMS), which are based on relational, network, and hierarchical models [5]. Pictorial Structural Query Language (PSQL) [6] and Query by Pictorial Example (QPE) [7] are two primary examples. The PSQL language is an extension of SQL (Structural Query Language), which supports user-defined abstract data types employed for definition of pictorial domains. Functions for computing attributes and comparison operators are defined on each domain. Here, the association between the pictorial and alphanumeric domains is used for retrieval purposes. In QPE, queries are specified using tables with a style similar to QBE (Query By Example), a well known method in traditional database systems.

To make an image retrieval system more flexible and more intelligent, visual information embedded in images should be preserved by exploiting an efficient data structure to store them. The symbolic description of graphical information such as shape, sketch, or spatial relations, using traditional approaches is a very difficult task [8]. Attempts to describe such information textually can lead to representations that are either too general or too complex.
In traditional database systems, the use of indexing to facilitate accessing databases, is well established. Similarly, image indexing techniques have been studied during the last two decades to elucidate image information retrieval from an image database. The use of icons as pictorial index is the basis of the iconic indexing methodologies developed to this aim. In this approach, the index of an image is the picture itself, which is best suited to hold visual information and to allow different levels of abstraction and management. The Query by Visual Example (QVE), developed by Hirata and Kato [9], is one of the earliest systems developed for image retrieval using image content. It defines pictorial indexes for database and for query images and compares them for retrieval task. As this approach is one of the well known methods in sketch-based image retrieval (SBIR), it will be discussed in detail in Chapter 3 (Sketch-Based Image Retrieval).

Different approaches to image indexing and retrieval can be classified into four categories: attribute-based, annotation-based, object recognition-based, and low-level image feature-based (see Figure 2.1). Each category is explained in the following subsections.
2.3.1 Attribute-Based Retrieval

This approach uses traditional database management system (DBMS) methods for image indexing and retrieval. Here, image contents are modelled as a set of structured attributes extracted manually. They are managed within the same framework of conventional DBMS’s. Queries are specified using the extracted attributes, and images, almost always, have predefined properties (attributes) that can be stored with them. Examples of these properties are the image file name, category, author, subject, image source, date of creation, and location. Therefore, images can easily be indexed and retrieved employing a powerful relational database model based on selected attributes [10, 11]. As well as the standard relational database features, some other features, not found in traditional DBMSs, could be used. For example, classes can be defined for objects in an image database and attributes can be inherited as in object-oriented paradigm. Moreover, some extensive metadata could be defined and employed for image description as attributes. The following is a sample entry for one arbitrary image selected from the State of California Department of Water Resources (DWR) database used in the Chabot project [11].

"0162 A-9-98 6/1/69 SWP Lake Davis Lahontan Region (6) Grizzly Dam, spillway and Lake Davis, a scenic image, DWR 35mm slide Aerial 2013055618"

This includes the first four digits of the compact disc number (0162) followed by the DWR identification number (A-9-98) and the date the photo was taken (6/1/69). The category SWP, which stands for State Water Project, the image description (Grizzly Dam...image), the source of the image (DWR), the type of film used (35mm slide), the perspective of the photo (Aerial), the last eight digits of the photo CD, and finally the image number on the photo CD are the selected image attributes.

The major drawback of this approach is that the predefined properties may not properly describe the image content. The queries are also limited to those predefined attributes. The attribute-based retrieval approach is advocated and advanced primarily by researchers in the database management domain.
2.3.2 Annotation-Based Retrieval

The annotation-based approach utilizes traditional information retrieval (IR) methods for image indexing and retrieval [12,13]. In this approach, free text is used to describe and annotate images. Queries are presented in form of keywords or free text, usually with Boolean operators. Techniques, based on similarity between the query and the annotating text, and clustering algorithms are used for retrieval. Text annotation is normally a manual process because automatic high-level image understanding is not possible (except in highly domain-specific applications). Two comprehensive review papers, surveying fundamental issues on this approach, are [14] and [15].

The main advantage of the annotation-based image retrieval is in its ability to capture high-level abstraction. Since text can describe the high-level abstraction contained in images, this method is widely used in many public domain search engines. Because there is no structure or attributes to limit the description, in contrast with the attribute-based approach, it is less domain specific. The text description can also be added incrementally [10]. General concepts, such as “flower”, “tree”, and ”smile” are easily annotated.

There are two important issues in image annotation that should be considered carefully: (a) how to do annotation efficiently, and (b) how to describe the image content completely and consistently. Domain knowledge or an extended thesaurus needs to be used to overcome the completeness and consistency problems. Relationships between words or terms should also be considered. For example, “webcam”, “scanner”, “monitor” and “keyboard” are subclasses of more general terms “computer devices” and “apparatus”. Consider that a user provides a query using keyword “apparatus”, intending to retrieve all images containing machinery devices. Without using an appropriate thesaurus, images annotated with “webcam”, “scanner”, “monitor” or “keyboard” are not retrieved, even though they are actually what the user is looking for. Using the thesaurus, images with these terms are also retrieved. Therefore, domain-specific knowledge is introduced to further improve retrieval performance when building such domain-specific systems [16].

Moreover, text annotation may not be complete and may be subjective. Conse-
quently the use of a knowledge base and relevance feedback is extremely important for annotation-based image retrieval. Relevance feedback gradually improves user’s satisfaction once the image is determined to be relevant or not. In addition, as the original text description may not be complete, relevance feedback enables us to modify the description to make it more complete and accurate [10, 17].

2.3.3 Object Recognition-Based Retrieval

Object recognition (or more precisely, automatic object recognition) is an endeavor that uses software tools to accomplish tasks which would be termed intelligent activity if performed by humans. Whether it is attempting for example to (a) match a query image to appropriate similar images in a given image database, (b) identifying cursive signatures in a text image, or (c) classifying handwritten characters, or fingerprints, it is an activity that is often difficult to describe precisely. But humans can do these tasks very well. The pattern recognition discipline was born in the late 1950’s, primarily to address optical character recognition, but has, over the years, grown to cover other areas. Applications of object/pattern recognition cover a broad scope of activities including military (analysis of aerial photography or automatic target recognition), medicine (analysis of electrocardiograms and electroencephalograms), agriculture (crop analysis), and engineering (character recognition and fault detection in manufactured products) [18, 19].

The object-recognition approach in unrestricted image retrieval is not mature yet as it relies on automatic and domain-specific object recognition [10]. Improvements in this approach greatly depend on semantic-based non-manual segmentation which is now in its early stages. This approach would be employed semi-automatically and in specific areas.

2.3.4 Low-Level Image Features

Low level image feature extraction and its usage in image retrieval is a well-established approach. The most common image features used in the literature are: color, texture, and object shape (spatial layout) [20–22]. These content features are discussed with more details in Section 2.4. Indexing and retrieval are carried out automatically in
this approach. This is an important issue, specifically when we are dealing with
large-scale image databases. Moreover, implementation can be easily managed us-
ing feature vectors and a similarity/distance measure. It has been shown that this
approach produces reasonably good retrieval performance and outperforms the other
approaches in many applications [10, 21, 22].

Attribute-based and annotation-based techniques are referred to as text-based ap-
proaches since they utilize text and alphanumerics symbols to describe images. De-
spite the advantages of using data modelling and multidimensional indexing tech-
niques which have matured in the text-based databases, the main challenges are the
tremendous amount of manual labor required to annotate the images and the inade-
quacy of few keywords to describe an image. This has led to more attention being
given to the content-based approach as a powerful and convincing alternative.

2.4 Content-Based Image Retrieval

Valuable contribution of many researchers in the last decade has led to the design
and develop of techniques which use the image content for indexing and retrieval.
Some survey papers of various content-based image retrieval (CBIR) approaches are
[2, 23–27]. The object recognition and the low-level features (color, texture, and
shape) described in previous section, can be considered as the main approaches of
the CBIR. Moreover, middle-level features including spatial layout and sketch, are
assigned to this category.

The first group of techniques (i.e. the object recognition) is currently not practical
for general applications [10] but the second and the third groups of techniques (i.e.
the low-level and middle-level features) have resulted in many practical algorithms.
The visual content of the image, instead of manually extracted annotations, is used
for the indexing and retrieval processes. Different techniques in this approach have
been developed and many special issues of leading journals have been dedicated to
this topic. The representative review papers are [2, 23–27].

In the content-based approach, the query provided by the user is usually an image
example, and the goal is to find similar images in the relevant database(s). The query and the database images need to be feature extracted first and then compared with each other. The prominent content features are color, texture, object shape, spatial relationship and sketch [1]. MPEG-7 standard suggests descriptors for color, texture [21], and visual shape of an object [22]. Computer programs can automatically extract and analyze these features and construct the corresponding indexes. Consequently, image storage and retrieval systems using this approach are more powerful and reliable.

The content-based approach can be summarized as follows:

1. Computer vision and image processing techniques are used to extract low-level content features from the image.

2. Images are represented as collections of their prominent features. For a given feature, an appropriate representation of the feature and a notion of similarity are determined.

3. Image retrieval is performed based on computing similarity in the feature space, and results are ranked based on the similarity measure employed.

Regardless of whether we use single or multiple features and the representation exploited for the features, the technique used for measuring the similarity plays an important role in finding the final results. The choice of this similarity function is critical and may be general or domain-dependent.

The general image retrieval problem could be formulated as follows:

\[ X = \{x_1, \ldots, x_M\}, \]

represents the image database with \( M \) members. Each image \( x_i \) will have an associate feature vector \( f_i \), which contains the relevant information extracted from the image and required for measuring the similarity between images. \( F = \{f_1, \ldots, f_M\} \) represents \( M \) feature vectors associated with \( M \) images. Let

\[ T : x \rightarrow f \]

(2.1)

represents a transformation from image space onto the \( n \)-dimensional feature space \( f \), where \( x \subset X \) and \( f \subset F \). Now, the similarity between two images \( x_i \) and \( x_j \)
could be measured using any distance function $d(f_i, f_j)$, which describes the distance between the feature vectors. The problem of image retrieval can then be posed as: Given a query image $q$, retrieve a subset of images $Y$ from the image database $X = \{x_1, \ldots, x_M\}$, $Y \subset X$ such that

$$d(T(q), T(y)) \leq t, \quad y \in Y$$

(2.2)

where $t$ is a user-defined threshold. Alternatively, as the user is usually unfamiliar with $t$ and $d$, they can ask the system to output, say, the top-20 images which are most similar to the query image [28]. Therefore, the main concepts in image retrieval can be summarized as: (a) the selection of appropriate feature vector(s) $f$, (b) the study of applied transform $T$, and (c) the definition and evaluation of distance function $d$.

The average dimension of feature vectors in CBIR is of the order of $10^2$ [29]. Although this dimension is high, the embedded dimension is low [30], so a dimension reduction process may be applied. The main techniques of dimension reduction are Karhunen-Loeve transform (KLT) and clustering [29]. Furthermore, any known indexing methods such as $R$-trees, $R^+$-trees and $R^*$-trees in DBMS literature can be used for feature vector indexing [4].

From a practical point of view, content-based image indexing techniques can be divided into two main domains: pixel and compressed domain techniques. Figure 2.2 exhibits the hierarchy of the techniques. In the pixel domain, the values of individual pixels in the image matrix are used directly for making visual indexes. On the other hand, in the compressed domain, transformed data, which is the result of mapping the original image matrix into another domain, is employed for feature extraction and indexing.

Different approaches for visual content analysis, representation, and their applications to indexing and retrieval have been studied in [24]. However, the survey on indexing and retrieval is brief. Pixel domain techniques, advantages, and disadvantages have been reviewed in [1]. Mandal et al. [20] present a critical review of image indexing methods in compressed domain. Image retrieval techniques, promising directions, and open issues are surveyed in [29]. The authors discuss feature extraction
methods and high dimensional indexing. A review of several image retrieval systems and future research directions are also given in this study. Shanbehzadeh et al. [23] have investigated image retrieval and indexing schemes in pixel domain. They review different approaches in color, shape, and texture categories. The authors anticipate that image coding algorithms will provide the capability of producing image features in compressed domain. Pixel domain and compressed domain techniques are explained with more details in the following subsections.

2.4.1 Pixel Domain Techniques

Typically, visual indexing techniques are based on features which are extracted directly from the pixel domain [20]. These include visual features that the human visual system can easily recognize. Working in this domain is more intuitive and does not require us to transform and re-transform the images at the store and retrieving stages, respectively.
Here, we review the most important image clues in the pixel domain, which are color, texture, shape, spatial relation, and sketch.

2.4.1.1 Color

Color is often used as a major feature for indexing in color images because of the important role it plays in vision in general and in the identification and discrimination of objects in particular. Some representative studies of color indexing and color spaces can be found in [4, 21, 31–34]. There are several color models used in the literature, but HSV (Hue, Saturation, Value), RGB (Red, Green, Blue) and YUV models are more popular. The HSV model correlates well with human color perception while the RGB color model is used in many imaging systems [35].

The YUV color space is usually used in video signal processing. This model, in a similar way, decomposes the color attribute into the three following components [36]:

- **Y**: luminance
- **U, Cb**: chroma channel, U axis, blue component
- **V, Cr**: chroma channel, V axis, red component

Some other definitions, depending on the context, are exist. The YUV decomposition model is used for the following reasons. First, converting an RGB signal (such as the one used on computer monitors) to the YUV model is just a linear transform, which is easily implemented in an analog circuitry. Second, the YUV color space allows to separate the color information from the luminance (which is our perception of brightness). For this reason the YUV space is used in the European television standard PAL and for the JPEG compression. In the television, the luminance channel is modulated separately, which allows old televisions to work even without color support (i.e. the “color” information is sent apart). Since the human eye is also much more responsive to the luminance attribute, the JPEG technique compresses more heavily the chroma channels which leads to a minor perceptible difference in the resulting image. Other names for the YUV model are YCC (sometimes referred to
as analog YUV, which ranges from 0 to 1), YCbCr (i.e. digital YUV, 8 bit unsigned data), and YPbPr.

The color distribution of an image is typically represented using a color histogram. For an image $X$, the histogram is an $N$-dimensional vector $\{H^X_i ; i = 1, 2, \ldots, N\}$, where $N$ is the number of colors in the image and $H^X_i$ is the number of pixels having color $i$ in the image $X$. The histogram is invariant to image rotation, translation and viewing axis. It is widely used as the image feature vector in many applications [37–41]. For a given image $X$, the color histogram $H^X$ is a compact summary of the image. A database of images can be searched to find the most similar images to $X$, and then the image $Y$, with the most similar color histogram $H^Y$ to $H^X$, is returned. However, there are some examples where two different images have very similar color histograms. See Figure 2.3 for an example.

Color histograms are typically compared using Manhattan or Euclidean distances [42]. More complex methods for measuring the similarity between color histograms have been studied in the literature [34,43,44]. Section 2.6 discusses different similarity metrics. Pass and Zabih [42] describe a technique for comparing images called histogram refinement. This technique imposes additional constraints on histogram-based matching by splitting the pixels in a given bin, employing local properties, into several classes. It has been shown that histogram refinement could be used to distinguish between images whose color histograms are indistinguishable. Ratio-based [45] color indexing for efficient search in image and video databases, and
using the first three moments of color distribution of images [46] show advantages in robustness over simple color histograms. Color histograms at different color depths (resolutions) are studied in [47] for possible use in filtering or matching between image features.

The MPEG-7 standard suggests four kinds of color descriptors. The descriptors, are generally defined considering high efficiency, low complexity, and other criteria such as applicability to a broad range of applications. They could be treated as different histograms with different definitions. The domain color descriptor gives the distribution of the salient colors in the image while the color layout descriptor (CLD) captures the spatial layout of the dominant colors on a grid superimposed on the region of interest. The latter is a very compact color descriptor and effective in fast browsing and search applications. The scalable color descriptor (SCD) addresses the interoperability issue by fixing the color space into HSV, with a uniform quantization of the space to 256 buckets. The bucket values are then uniformly quantized into an 11-bit value. Finally, the color structure descriptor (CSD) expresses local color structure in an image using an $8 \times 8$-structuring element. It counts the number of times a particular color is contained within the structuring element as the structuring element scans the image [21, 48].

2.4.1.2 Texture

Texture is a property of a region resulting from homogenate visual patterns. It can be defined as the appearance characteristics of a surface having a tactile quality. It is a natural property of all surfaces such as fabric, clouds, bricks, trees, hair, and woods. The texture feature is illumination invariant [23]. A traditional representation used for texture is the co-occurrence matrix [29]. This approach explores the gray level spatial dependence of underlying image to represent its texture. An enhanced version of this statistical approach which experimentally determines contrast, inverse deference moment and entropy possesses better discriminatory power [49]. A computational approximation to the visual texture properties is developed based on psychological studies in human visual perception of texture [29]. The six prominent visual texture properties are coarseness, contrast, directionality, linelikeness,
regularity, and roughness.

One major difference between the psychological approach and the co-occurrence matrix representation is that all the texture properties in the former representation are visually meaningful, whereas some of the texture properties used in the latter method may not be meaningful (for example, entropy). These characteristics make the psychological texture representation approach very attractive in image retrieval. Moreover, it can provide more user-friendly interfaces. The QBIC system [37, 50] and the MARS system [51] further improve the psychological texture representation.

Following the introduction of the wavelet transform in the early 1990s and the establishment of its theoretical framework, many researchers began to study the use of the wavelet transform in texture representation. Its application in image retrieval has been studied extensively [29, 52–54]. The mean and variance extracted from the wavelet subbands can be used as the texture representation. Typically, textured images are decomposed into separate frequency and/or orientation bands and then features are extracted separately from each subband. Busch and Boles [55] show that features modelling the relationships between the subbands provide a better characterization of textured images than features extracted from individual bands alone. They developed a feature set for texture classification and demonstrated its effectiveness by applying the method on a set of images from the Brodatz texture album. A new method for modelling multivariate distributions of subband coefficients by considering spatially related coefficients is introduced in [56]. A statistical view of the texture retrieval problem by combining the feature extraction phase and similarity measurement phase is proposed in [57]. Pun and Lee [58] introduce a two-stage wavelet packet feature approach for classification of rotated textured images. Thyagarajan et al. [59] combine the wavelet transform with the co-occurrence matrix to take advantage of both statistics-based and transform-based texture analysis. Gabor filter banks are frequently used for measuring the similarity based on texture features [60–62].

The MPEG-7 standard defines three different texture descriptors. The texture browsing descriptor characterizes perceptual attributes such as directionality, regularity and coarseness of a texture. The homogeneous texture descriptor (HTD) provides a quantitative characterization of homogeneous texture regions for similarity retrieval.
It is based on computing the local-spatial statistics of the texture. The third descriptor, the *edge histogram descriptor* (EHD), is useful when the underlying region is not homogeneous in texture properties [21, 63].

The semantic power of the EHD method, in a logo matching process, is examined subjectively in [64]. The results show that the retrieval performance of the MPEG-7’s EHD descriptor is worse than the edge pixel neighborhood histogram and the correlation method [9], respectively. However, the EHD descriptor is useful in image-to-image matching (by example or by sketch), specifically when natural images, with non-uniform edge distribution, are compared [65]. Therefore, the EHD is one of the candidate methods in sketch-based image retrieval. The details of this descriptor are provided in Chapter 3 (Sketch-Based Image Retrieval) of this thesis.

### 2.4.1.3 Shape

Based on semantic information carried by shape, a large body of research has been devoted to object shape representation [66], recognition [67, 68], retrieval and indexing using shape information [69–72]. Depending on the applications, some users require the shape representation to be invariant to translation, rotation and scaling, while in some other applications these invariance properties are not important [29].

The shape representation can be divided into two broad categories: boundary-based, and region-based techniques. The first category uses only the outer boundary of the shape for feature extraction while the second one uses the entire shape region. The most popular representatives in the above categories are Fourier descriptors and moment invariants, respectively. The main idea of the Fourier descriptors is to use the Fourier transform of the surrounding contour as the boundary-based shape feature. Object’s moments, which are invariant to transformations, are used as the region-based shape features. Seven invariant moments originally identified by Hu are improved and used broadly in shape-based image retrieval [4]. The invariant moments are used in [73] for trademark matching, and in [74], together with the edge direction histogram, for a trademark registering process. Goshtasbi defines a shape descriptor which is invariant to translation, rotation, and scale, in the form of a matrix utilizing polar quantization [66]. In [71] a structural feature index, obtained from the object
boundary, is used for similar shape retrieval. Ankerst et al. [70] propose a distance measure for measuring the similarity of two shapes. The approach is based on pixel-to-pixel comparison in a neighboring rectangle using a user-defined weight.

A technique for retrieving similar shapes is proposed by Jagadish [69]. It works in two dimensions. The approach derives an appropriate object description from a rectilinear cover of the given object. The cover consists rectangles with parallel axes. The problem of detecting shapes anywhere in a given image that is similar to the objects in a database is addressed in [75]. The finite element method (FEM) for shape representation and matching is used in the Photobook system [76]. The FEM defines a stiffness matrix that describes how each point of the object is connected to the other points. The eigenvectors of the stiffness matrix are called modes and span the feature space. All shapes are first mapped onto this space and similarity is then computed based on the eigenvalues. In [68], Chuang and Kuo apply the wavelet transform to describe object shape. This approach takes advantage of wavelet properties such as multi-resolution representation, invariance, uniqueness, stability, and spatial localization for shape representation and retrieval.

Elastic template matching (ETM) method, proposed by Del Bimbo et al. [77, 78], uses the elastic deformation of the query-sketched shape for matching the shapes in the database to obtain a similarity measure. In this approach, the boundary of the query shape is progressively deformed to overlap the shape in the database. After a query shape has reached convergence over a shape, the similarity measure between the two shapes is obtained. The amount of energy spent in the deformation process models the perceptual distance between the template (query) and the target image. Indexing with this approach is generally complex or impossible [4]. The approach is also computationally expensive and can be used only with single objects. Where the image contains multiple objects, each object needs to be considered individually resulting in a more complex and slower process.

Sajjanhar and Lu [79, 80] developed a grid-based method for shape representation and measuring the similarity among shapes. A grid space is overlayed on a given shape. The grid space consists of fixed size square cells. The cells fully or partially covered by the shape are assigned 1 and the other cells 0. The cells are then read from
left to right and top to bottom to obtain a binary sequence for the shape. This binary value is used to represent the shape and also for matching purposes. The method is suitable for structured and simple shapes created by connecting rectangles and triangles. It needs to be normalized because of rotation and scale variations. The proposed approach is compared with the moment invariants method, the Fourier descriptors, the query by image content (QBIC), the query by visual example (QVE), and the elastic template matching (ETM) methods [81]. It has been shown that the overall performance of the grid-based method is comparable to the ETM method while it outperforms the QBIC and the QVE methods. The performance of the Fourier-based technique is reported to be very close to that of the grid-based technique. Since the test database is specific and small (22 images) with a limited number of queries (3 images), the results are not conclusive.

The 2D Fourier transform in polar coordinates is employed for shape description in [82]. Its supremacy over 1D Fourier descriptors, curvature scale space descriptors and Zernike moments is shown in [83]. The polar Fourier descriptors (PFD) are extracted from the frequency domain by exploiting a two-dimensional Fourier transform on polar raster sampled images. It is able to capture image features in both radial and spiral directions [83]. A contour-based approach using 1D Fourier transform and region-based approach using Zernike moments are combined to improve retrieval performance in [84].

The MPEG-7 standard supports different notion of similarity using region-based and boundary-based shape descriptors. Four shape descriptors have been defined in the standard [22]:

- region-based,
- boundary-based,
- 3D, and
- 2D-3D

The region-based shape descriptor expresses the pixel distribution within a 2D object
region. It belongs to the broad class of shape analysis techniques which are based on moments. The descriptor uses a complex 2D angular radial transformation (ART), defined on a unit disk in polar coordinates. The boundary-based shape descriptor is based on the curvature scale-space representation of the contour [85]. If a complex object consists of multiple disjoint regions, using this descriptor, each region of the component contours needs to be described separately. The MPEG-7 3D shape descriptor is based on the shape spectrum concept. It is an extension of the shape index, used in MPEG-4 as a local measure of 3D shape to 3D meshes. Finally, the 2D-3D descriptor can be used to combine 2D descriptors representing a visual feature of a 3D object seen from different viewing angles. This descriptor forms a complete 3D view-based representation of the object. Any 2D visual descriptor, such as the contour shape descriptor, the region shape descriptor, or color or texture descriptors can be used to extract the 2D-3D descriptor.

### 2.4.1.4 Spatial Layout

The spatial relation between objects (layout) is a criteria for image retrieval in many applications. For example, in geographical information systems (GISs), there are queries such as ”select images containing a lake to the left of a forest” or in general applications, “find images containing a chair in front of a computer”. For these types of queries, objects and their spatial relationships have to be determined. Here, the object should be extracted first and then their relations should be properly represented.

One of the best known approaches to represent the spatial content of an image is based on a 2D-string structure. This structure first introduced by Chang [86] and many other variations have been tried in the literature (e.g. those in [87–90]). In this approach, once the objects are segmented and recognized, they will be projected along the $x$ and $y$ axes using a set of symbols $S$ and a set of operators $A$. These symbolic projections expressed by text strings are used for comparing different scenes. For example, $S = \{a, b, c, d, e, f\}$ and $A = \{<, =:, \}\}$, where the alphabets in $S$ are used to represent a limited number of image components and operators in $A$ to describe the existing spatial relationships among the component. The operator “$<$” denotes, respectively on the $x$ and $y$ axes, the left-right and below-above spatial rela-
Figure 2.4 Using 2D strings to represent spatial relationships; (a) an image example, containing three objects, and (b) the corresponding symbolic picture.

An algorithm for computing the spatial similarity is proposed by Gudivada and Raghavan [91]. A symbolic image, which is a logical representation of the original image, is used in this approach. Here, the image objects are uniquely labelled with symbolic names at their centroid locations. The approach assumes that images have been (automatically or manually) segmented into meaningful objects. A framework for retrieving images by spatial similarity is designed in [92]. In this framework, similarity between two images is a function of the number of common objects and the closeness of directional and topological-spatial relationships between object pairs in both images. Spatial information and spatial relationships are employed together with color information in the VisualSEEk system to enhance retrieval performance [38]. Matching 2D strings is based on a ranking scheme for the objects in the string. This rank is calculated using the relative ordering of objects in the string [91].

The 2D B-string, which is an extension of the original 2D string, has been proposed by Lee et al. [93]. This is a technique for image indexing without the need for object partitioning (object segmentation is still required). The beginning and the end
boundaries of objects are kept in a set of object symbols $S$. The special symbol “=” denotes that the projections are at the same spatial location. A 2D B-string over $S$ is defined as $(r, c)$, where $r$ and $c$ are 1D strings over $\{=\}$. Each 1D string represents the relative ordering of the boundaries of objects projected along horizontal or vertical axes, respectively. For example, $(ababcc, b=cabca)$ is the 2D B-string representing Figure 2.4.

The object symbols are assigned a rank as follows:

$$ rank(s_i) = \sum_{k=1}^{i} order(s_k) $$

(2.3)

where

$$ order(s_k) = \begin{cases} 
1 & s_k \neq \text{"="}, \\
-1 & \text{otherwise}
\end{cases} $$

(2.4)

The rank of objects in companion with 2D B-strings is used for measuring the similarity between images. It is worthwhile to note that these techniques cannot be applied to images with a complex background. Images with several overlapped objects lead to very long strings which are difficult to handle and cause a long time overhead.

2.4.1.5 Sketch

The content of an image can be described using a sketch. A sketch is an abstract image which contains the outline of the main components in the image. To index an image database, a sketch image is generated for each image and used as index of the corresponding image. Typically, a sketch is created by using edge detection, thinning and shrinking algorithms and then used as a key to retrieve the desired image(s) from the database. Measuring the similarity of two images is done by comparing the corresponding sketches. Although there exists a large body of research on color, texture, and shape features, few have addressed sketch-based image retrieval (SBIR), specifically where the sketched query is a simple hand-drawn image with no color and texture attributes (rough sketched).

Hirata and Kato proposed a correlation-based technique for sketch-based image retrieval called query by visual example (QVE) [9]. A modified version of this method
Literature Review

is embedded in the QBIC system [94] by IBM. In this approach, the query and target images are resized to 64×64 pixels and then a proposed gradient operator extracts their edges. The resulting edge map is called *pictorial index*, which is used for image-to-image matching. The details of this approach are provided in Chapter 3. Although the method has a good ability to find similar images in small databases, its main shortcomings are orientation and scale variance properties. Similar images with different orientation or scale will not be retrieved when compared with the query image [1]. The convolution is made in the corresponding local blocks and consequently the global rotation of images is not recognized in this approach. Because of its expensive computational cost, using this method in large-size image databases is prohibitive [64].

Jacobs *et al.* [95] introduce a searching algorithm based on multi-resolution wavelet decompositions of the sketched query and the database images. The coefficients of these decompositions are employed to generate signatures for images. By applying a novel metric, devised for the signatures, the similarity of images is measured. The sketched and the database images both contain color and texture attributes. Sciascio *et al.* [96] have proposed a low-dimensional color fragmentation-weighted histogram for query by sketch approach. It outperforms non-fragmented color histograms. A spatial layout representation for query by sketch is proposed in [97], which needs objects in the image to be segmented a priori. The approach provides a technique to extract the image content starting from basic features and combining them in a higher-level description of spatial relations.

Agouris *et al.* [98] have proposed a prototype system for sketch-based image queries. Here, simple and generic objects with, e.g. circular shape, representing cooling towers, and rectangular shape, representing buildings, are used to simulate a topographic database. A metadata library, a semantic library, and a feature library are employed to retrieve similar images from the database. The system is an improvement of the Image Query (IQ) system [99], which is an image database environment for searching in satellite images using hierarchical arrangement of features. Edge maps are extracted in this method from target and query images and used for the matching process. A multi-layered architecture is reported in [100], which is designed for
sketch-based interactive engineering drawing.

Using histograms of edge directions (HED) for representing image information is one of the well known methods in the image retrieval field. The feature is appropriate also for sketch-based image matching as it compares the distribution of edge points in the edge map of the query image with the corresponding information in the database image. Abdel-Mottaleb [101] uses the approach by applying the Canny edge operator [102] to find strong edges in an image and then quantizes them into 4 directions. Jain and Vailaya [34] also propose edge directions as an image attribute for matching purposes [74].

Matusiak et al. [103] and Daoudi et al. [104] have previously reported using rough and simple hand-drawn shapes as the input queries for SBIR. These two approaches are similarly based on curvature scale space that is computationally expensive and has been shown to be less efficient than Fourier descriptors and Zernike moments techniques [83]. In [105] several dominant points are extracted for each sketched contour using information derived from the convex hull and the contour curvature.

Based on the assumption that using sophisticated edge representation and matching algorithms can eliminate some shortcomings in the SBIR [1], we have proposed a new approach for sketch-based image retrieval [106]. The approach is based on edge pixel neighborhood information (EPNI) and uses the overall structure of edge pixels in the image for similarity measurement. We define a neighboring diagram, an edge pixel neighboring histogram (EPNH), and a vicinity table for storing first and second order neighboring information, respectively. We have compared the semantic ability and the efficiency of the proposed approach with some other methods including invariant moments, correlation (QVE), edge angles and the MPEG-7’s edge histogram descriptor [64]. The comparative results show that the correlation method has the best semantic power, i.e. closest to the human judgment but it is extremely slow. The proposed method yields the second best performance at a reasonable speed. Although the retrieval performance of the EPNI method is acceptable, it is not rotation invariant and is therefore good only for situations where the sketched query and the database images have similar spatial directions.
Although the MPEG-7 standard has introduced sketch-based database searching as an MPEG-7 application in [107], it has not defined any descriptor for this feature yet. However, the edge histogram descriptor, which is defined in the MPEG-7 texture part, has been used for the SBIR [108].

2.4.2 Compressed Domain Techniques

The large volume of visual data necessitates the use of compression methods. Multimedia databases usually store the visual content in compressed form and most images are stored using existing compression techniques. In order to reduce the cost of decompression of the image data and applying pixel domain techniques, it may be more efficient (if applicable) to index visual information in the compressed form. This approach often has a lower cost for computing and storing the indexes [20].

Compressed domain indexing is broadly classified into two main categories: spatial domain, and transform domain techniques (see Figure 2.2). The major techniques in the first category are vector quantization (VQ) and fractals. The second category is generally based on image transformation techniques including DFT (discrete Fourier transform), DCT (discrete cosine transform), subbands/wavelets, and KLT (Karhunen-Loeve transform). In the following we briefly review these techniques.

2.4.2.1 Vector Quantization (VQ)

A vector quantizer can be defined as a mapping $V$ of an $n$-dimensional Euclidian space $R^n$ into a finite subset $X$ of $R^n$ that is:

$$V : R^n \rightarrow X$$

where $X = \{x_i\}; i = 1, 2, 3, \ldots, K$, and $x_i$ is the $i$th vector in $Y$. $Y$ is a set of reproduction vectors, called VQ codebook or VQ table, and $K$ is the number of vectors in $X$ [10]. For the purpose of image compression (or data transmission), each data vector $x$ belonging to $R^n$ is mapped onto a codeword in the codebook, and the address or the index of that codeword is stored (or transmitted). The reverse procedure is accomplished for decoding. The VQ method has the attractive feature that a compressed image is represented as a sequence of numbers, each of which identifies
a code vector in the codebook. Therefore, the VQ compressed image consists of code vectors represented by numbers (indexes).

Idris and Panchanathan [109] propose a VQ-based image indexing technique. In this technique, the compressed images and the labels are stored in the database and the histograms of the labels are used as feature vectors, which are employed for indexing. Another method of image retrieval based on the VQ technique is proposed in [110]. If a code vector is used by an image, it will be recorded in a usage map. The similarity between images is calculated using the corresponding usage maps. The retrieval precision is low since two totally different images can have similar usage maps. This is due to the fact that the usage map does not show how many times a particular code vector is used by an image.

An alternative approach is discussed in [10]. For a given image, VQ compression is applied first. Each block of pixels is then represented by a code vector index number. Next, the number of occurrences of each index is calculated to obtain an index histogram $H_D(h_1, h_2, h_3, \ldots, h_n)$ for database image $D$. $h_i$ is the number of times the code vector $i$ is used by image $D$, and $n$ is the total number of code vectors in the codebook. For image retrieval, the distance between the query image $Q$ and the database image $D$ is obtained using the corresponding index histograms as follows:

$$d(Q, D) = \sum_{i=1}^{n} |H_{D_i} - H_{Q_i}|$$  \hspace{1cm} (2.6)

This distance value is used to rank similar images. The retrieval process in this approach is very similar to the basic color histogram approach. The main difference is that in the basic color histogram the number of pixels with the same color is used while in the VQ-based technique, the histogram represents the number of different code vectors.

An index-compressed VQ (IC-VQ) technique is developed by Shanbehzadeh and Ogunbona [111,112] and employed for image retrieval [113]. The indexes generated by this method, use information about the relationships between two adjacent image blocks which are referred to as “inter-block correlations”. It has been shown that the retrieval performance of the IC-VQ method is better than generic VQ and color histogram approaches in some limited applications.
2.4.2.2 Fractal

Fractal coding is classified within the broader area of model-based image compression, where an image is represented by a mathematical model. In a fractal-based approach, an image, normally an object, is formulated by a number of parameters or mathematical equations [114]. Since very little data is required to represent these parameters and equations, a very high compression rate can be achieved. This compression method can be used for content-based image retrieval, where the parameters and values defining the equations serve as image/object indexes. The image similarity is calculated from the differences between these parameters.

In terms of fractal geometry, a fractal is a geometric form whose irregular details reappear at different scales and angles, which can be described by affine or fractal transformations. Originally, fractals have been used to generate images. Currently, fractal formulas can be used to describe almost all real-world pictures [10]. Fractal image compression is the inverse of fractal image generation, i.e. instead of generating an image from a given formula, fractal image compression searches for sets of fractals in a digitized image which describe and represent the entire image. [115] and [116] are two review papers surveying different approaches in fractal-based image generation, compression, and retrieval.

A texture-based image retrieval method is proposed in [117]. It determines the image similarity based on the match of fractal codes. Each image is decomposed into block-based segments, which are assembled as a hierarchy using inclusion relationships. Each segment is then fractally encoded. During image retrieval, the fractal codes of a query image are used as a key and are matched with the fractal codes of the database images. The retrieval performance is improved by applying searching and matching algorithms to the aforementioned hierarchy of images existing in the database.

The performance of wavelets and fractals in image retrieval is compared in [118]. The mean absolute values and variances of different subbands are utilized as the image features in the wavelet domain. On the other hand, in the fractal domain, a joint fractal coding of two images is employed for matching. The authors have concluded, based on simulation results, that the wavelet-based technique is more effective for
images containing strong texture features whereas the fractal-based technique performs relatively better for a variety of images. However, the conclusion is valid only for the data set used and only in this particular framework. That is, the retrieval performance may change if other approaches with different data sets are tested.

Fractal interpolation is used in [119] to describe edge images and Pi et al. [120] introduce a 2D histogram of fractal parameters for image retrieval. Based on the fact that a fractal transform is determined by luminance offset and contrast scaling, a statistical index is proposed in this approach using histograms of luminance offsets.

The fractal-based approach is usually used where images in the database and the query images have texture attributes. The major problems with fractal-based approach are (a) it is difficult to find the required affine transforms for general images, and (b) one image can be formulated by different sets of affine transforms.

2.4.2.3 Discrete Fourier Transform (DFT)

The discrete Fourier transform is a well known mathematical tool and has a great influence on image and signal processing. The DFT uses complex exponential basis functions. As it has a good energy compaction property, it provides a good coding performance. The DFT possesses several remarkable properties that are useful in indexing, retrieval, and pattern matching. Most importantly, the magnitude of the DFT coefficients are translation invariant. They can be manipulated to be rotation and scale invariant too [121].

A two-threshold retrieval algorithm is proposed in [58]. The thresholds enable the user to declare the closeness of a match. The first threshold is employed for an intensity match and the second one is for a texture match. Neural network-based classification of satellite images, using texture measures in the Fourier domain, have been evaluated in [122]. The magnitude of the Fourier spectrum is the basic feature in all measures. The measures fall in three different categories: (a) statistical measures including average magnitude of Fourier coefficients, variance of magnitude, maximum magnitude, and energy of magnitude, and (b) radial distribution of Fourier coefficient, and (c) angular distribution of the coefficients. It has been shown that the
radial and angular measures yield good classification when a few dominant frequencies are present in the images. On the other hand, the statistical measures provide a satisfactory classification in the absence of dominant frequencies [20].

Image registration is a general technique which allows two image data sets to be brought into a geometric alignment. The Fourier transform is applied for image registration and outperforms gradient-based approach [123]. Different variations of the transform are used for object-based image retrieval [72, 82, 83, 124, 125]. Folkers and Samet have applied the DFT to describe logo images using generic shapes including rectangle, circle, and ellipse [126]. A comparative study of DFT-based descriptors for shape representation and retrieval is reported in [127]. The authors have studied different Fourier-based shape descriptors including centroid distance, position function, area function, and cumulative angular function in terms of robustness, speed, and retrieval performance. They conclude that the centroid distance approach is better in retrieval performance than other approaches. We have implemented this approach and examined it for sketch-based shape retrieval in Chapter 6 of this dissertation.

Although the DFT method has resulted in reasonable performances in many applications, there are some applications where the time-domain approaches perform better. For example, Wang et al. [128] have applied an efficient approach to leaf image retrieval which outperforms the curvature scale space (CSS) and the DFT. The proposed approach utilizes simple shape features including a centroid-contour distance curve, eccentricity and angle codes histogram.

2.4.2.4 Discrete Cosine Transform (DCT)

The DCT uses real sinusoidal basis functions. It has energy compaction efficiency near to optimal Karhunen-Loeve transform for most natural images. Accordingly, all international image and video compression standards, such as JPEG, MPEG and H.261/H.263 exploit the DCT for compression and coding [20]. Currently, most images are compressed using the JPEG method and it is desirable to index and retrieve images based on JPEG compressed data. Briefly, JPEG does compression in three main stages: (a) DCT computation, (b) quantization, and (c) entropy coding, which is performed utilizing variable-length coding [129].
Based on entropy coding at the last stage of JPEG, it is difficult to extract semantics directly from the compressed data. Therefore, the common approach is to do reverse entropy coding and carry out indexing and retrieval tasks based on the quantized DCT coefficients [10].

Ngo et al. [130] have proposed an approach for extracting features directly from the JPEG compressed data. Shape, texture, and color features are described in the DCT domain by exploiting ten DCT coefficients. Although the speed of retrieval is high, the performance is moderate. In [131] some characteristic images (pictorial indexes) are extracted from their original compressed domains, which are DCT and wavelet. The scale invariance property is achieved by normalizing the pictorial indexes to a predefined size. The computational cost to extract the pictorial index from the corresponding original image is considerably expensive.

A technique based on mutual relationships between the DCT coefficients of unconnected regions in both database and query images is proposed in [132]. A set of $2K$ windows is selected and randomly paired to produce $K$ pairs of windows. For each window the average of each DCT coefficient is computed. As a result, a 64-dimensional feature vector is generated. Next, the feature vectors, corresponding to a pair of windows, are compared and each pair of components is assigned a bit of 0 or 1 based on their similarity. Therefore, each pair of windows will be assigned 64 bits. Finally, the similarity of the query and the database images is evaluated by the overall similarity of all the bits in all window pairs. A similar approach, proposed by Ordonez et al. [133], is applied for indexing medical images in JPEG compressed format. Here, the combination of spatial and spectral information is employed for feature extraction. Spectral information is extracted from the DCT coefficient histogram and spatial information is derived from the gravity centers and inertia moments associated to the frequency histograms.

### 2.4.2.5 Wavelet Transform

The result of wavelet compression is a set of wavelet coefficients. Recently, the discrete wavelet transform (DWT) has become popular in image coding, indexing, and retrieval. An image, recursively, is passed into a set of lowpass and highpass
filters. The output of the filters are down sampled in order to maintain the same data rate. The DWT has many applications in signal processing and is also a powerful tool to describe planar curves, images and videos.

Basically, to index an image using the DWT, it is proposed that only a selected number of coefficients is used. The selected coefficients are quantized into -1, 1, and 0, if the coefficient is a large negative number, large positive number, or otherwise, respectively [10]. The extracted sequence of -1, 1 and 0 is used as the index of the image. Similarly, the query image is represented by a sequence of -1, 1, and 0 during the image retrieval process. The weighted sum of differences of the corresponding numbers in two sequences, related to the query image and the target image, is calculated and used for ranking the database images. The weights are selected to differentiate the significance of different coefficient positions.

Jacobs et al. [95] have proposed a searching algorithm based on multiresolution wavelet decompositions of query and target images. The coefficients of the decompositions are shrunk down into “image signatures”. First, a standard 2D Haar wavelet decomposition of each database image is obtained. Overall average color and the indexes and signs of the 60 largest magnitude wavelet coefficients are stored. For each query image the same procedure of Haar-based wavelet decomposition is applied and the average color and the largest 60 coefficients are obtained and used for comparison.

An algorithm for image retrieval using both color and texture features has been proposed by Yumin and Lixia [134]. Circular region energy in the low frequency bands of the wavelet transform is employed as the color feature of the image and the energy in high frequency bands of multi-scale wavelet transforms is exploited as the texture feature. Rotation and scale invariant features using log-polar wavelets are proposed in [135]. An energy signature is computed for each subband of the extracted wavelet coefficients which are employed for similarity measure. The experimental results show that this algorithm has superiority in retrieval accuracy of textured images over the traditional wavelet approach.

A hybrid technique which utilizes the compression power of wavelets and the affine
transform invariant property of the Fourier approach has been developed by Sabhawal and Subramanya [136]. The user specifies an index size, which is used to determine the wavelet decomposition level. Then, the image is wavelet transformed up to the specified level and the approximation part is extracted. Finally, a double DFT power spectrum is computed from the approximated part of the wavelet transformed image to enforce the invariant properties.

Oh et al. [137] have proposed to use self-organizing maps in neural networks for image clustering. The algorithm employs multiresolution wavelets to extract feature vectors for color and texture attributes. As the size of the feature vector of the input data (used for learning) and the number of units for the input layer of the self-organizing map is $8 \times 8 \times 3 = 192$, the proposed approach has a low speed. Although the method shows effective retrieval performance, the algorithm was tested on a set of only 120 images, hence the results are not conclusive.

The best way to represent a signal using wavelets is to scan the entire image for a *mother wave* that best represents that particular image. However, the mother wave would have to be sent/saved along with the image data, thereby increasing the size of the compressed file. A new still-image compression standard, referred to as the JPEG-2000, has been developed by the ISO JPEG committee. Many researchers have focused on different aspects of the JPEG2000 compression method [138–142]. Part 1 of the standard, which defines a baseline codec, was issued as an International Standard in 2001 and is starting to appear in a diverse set of products. The JPEG2000 is based on wavelet compression and provides the potential for numerous advantages over the existing JPEG standard. Advantages of the JPEG2000 mechanism are beyond the improved compression efficiency at low bit rates for large images. The new functionalities include multi-resolution representation, scalability and embedded bit stream architecture, lossy to lossless progression, ability of region-of-interest (ROI) coding, error resilience, remaining identical in multiple compression cycles, and a rich file format. Sanchez et al. [141] have proposed a method for prioritized coding of region-of-interest. Here, multiple ROI are encoded in the JPEG2000 image-coding framework using rearrangement of packets in the code-stream to place the ROI before the background coefficients. The method utilizes a Gaussian priority distribution.
to assign different priorities to background and ROI packets.

The JPEG2000 method, picks a universal mother wave ahead of time, so that the receiver of a compressed file can assume which mother wave is being used, eliminating the need to send it along with the file. The JPEG2000 compression specifies a 9/7 wavelet for ordinary *lossy* compression, and a 5/3 wavelet for *lossless* compression, comparable to today’s run-length-limited encoding. Compression ratios of 200:1 can be achieved with the 9/7 wavelet and 2:1 compression with the lossless 5/3 wavelet [143, 144]. The algorithm can also be implemented on VLSI chips [140].

2.4.2.6 Karhunen-Loeve Transform (KLT)

KLT or principal component analysis (PCA), is an approach for image transform and compression based on the statistical properties. The basis functions in this technique are the eigenvectors of the autocorrelation matrix of the image. The technique provides maximum energy compaction. It is statistically the optimum transform [20]. As the basis functions employed in this method are image adaptive, a satisfactory indexing performance is achievable comparing the KLT coefficients. The coefficients are obtained by projecting the images in KLT space.

Applications of the KLT to aerial color image analysis are proposed in [145] and [146]. Here, KLT is used to perform region-based segmentation of images based on color and texture features. For every color scene a particular distribution of energy, which can be represented in a chromatic axis system, is defined. The KLT technique has been used in the Photobook system for face recognition [76]. In this approach, *eigenfaces*, a set of optimal basis images, are created based on a randomly chosen subset of face images. The query image is then projected onto the *eigenfaces*. Here, faces are recognized based on the Euclidean distance between the corresponding KLT coefficients of the query and the database images. As the KLT basis images are ranked utilizing the eigen values, the salient characteristics of face images can be well represented using s first few (15-20) KLT coefficients [20, 76].

Despite the fact that the KLT technique possesses the potential of providing good compression performance, it is generally not used in traditional image coding be-
cause of its higher complexity. Furthermore, its computational demands and its batch calculation nature have limited its applications. Therefore, different variations of the generic KLT method are proposed and employed in specific applications including remote sensing, aerial image analysis, cloud classification and face recognition. Levy and Lindenbaum have proposed a sequential algorithm to increase computational speed of the traditional KLT method [147].

2.5 Different Queries

An essential aspect of any VIR and CBIR system is “how to pose a query”. Main approaches are query-by-keyword, and query-by-example. Query-by-keyword is used in traditional textual-based (attribute- and annotation-based) approach. In contrast, the query-by-example is the most salient feature of CBIR. Here, the user provides an initial image and asks the system to find similar images. Query-by-example can be divided into four different categories: (a) a full-color image is freely chosen by the user and applied as the query, or (b) a full-color image is selected from a predefined set of images, or (c) a freely sketched image is built up on the screen or created by scanning an image, or (d) the sketched image is selected from a limited number of sketches provided by the system. Availability of combinations of these querying ways is desirable for many users.

Since users do not always have an example image at hand, most commercial systems, influenced by the ideas originally developed for IBM’s QBIC system [37,50,94], provide facilities to choose/generate the initial image query. The original QBIC interface allows users to generate color queries either by varying the relative amounts of red, green, and blue, or by selecting a predefined color from a palette. Texture queries could also be specified by choosing from an existing palette, and shape queries by sketching the desired object on the screen. These methods proved adequate as an image querying interface, but they are cumbersome. Therefore, later systems have integrated a set of more intuitive methods. These include sketch-based interface for shape retrieval and provision of primitives such as rectangles and circles to help generating the query image [148].
Jacobs et al. have implemented an interactive querying platform [95]. Here, the user may choose any stored image file as an initial query or paint a sketch with color and texture attributes using graphical tools provided. Rather than choosing or painting a query, the user can also depict any of the retrieved images to serve as a subsequent query. Scanned images could be posed as a query as well. Delivering scanned images as the initial query is the easiest way for many novice users as they do not need to draw and paint their desired examples.

The shape queries using image databases (SQUID) system [149], which is developed at the University of Surrey, uses a library of fish contours for query selection. A demonstration version of the SQUID system can be found at its web site [150]. Matusiak et al. [103], uses the same database as SQUID, but users can draw the requested marine shape by hand on a query submission window. The curvature scale space method is employed to extract features for the query and the database shapes. In [105] the input query is a simple shape which is drawn by hand. On the other hand, in the QVE method [94] and in the approach proposed by Chan and Kung [151], a more complicated sketch query is submitted. Here, the input query is scanned and feature extracted for matching. This is based on the fact that images which contain more than a few simple objects are difficult to generate by software tools. Usually they are sketched by hand on a paper or canvas and then scanned to be a digital image.

Realizing that many users have limited artistic ability, VisualSEEk [38] and ImageRover [152] allow query shape to be built up on the screen from available primitives such as rectangles, circles, and ellipses. Commercial systems usually combine text-based and content-based querying methods to achieve a higher retrieval performance.

2.6 Similarity Measures

Similarity measurement is a key point in CBIR algorithms. Theses algorithms search image databases to find images similar to a given query, so, they should be able to evaluate the amount of similarity between images, quantitatively. Therefore, feature
vectors, extracted from the database image and from the query, are often passed through a distance function (metric) to find out the degree of closeness. For any distance function $d$, the following metric axioms must be satisfied [153]:

- $d(f_1, f_2) \geq 0$, $d(f_1, f_2) = 0$ if and only if $f_1 = f_2$
- symmetry property, i.e. $d(f_1, f_2) = d(f_2, f_1)$
- triangle inequality property, i.e. $d(f_1, f_3) \leq d(f_1, f_2) + d(f_2, f_3)$

where $f_1, f_2, f_3$ are three different $N$-dimensional feature vectors.

The aim of any distance function (or similarity measure) is to calculate how close the feature vectors are to each other. There exist several common techniques for measuring the distance (dissimilarity) between two $N$-dimensional feature vectors $f$ and $g$. Each metric has some important characteristics related to an application. The following are the most important metrics used in the literature.

- **Manhattan metric**
  The Manhattan (Cityblock), or $\ell_1$ metric is simply defined by:
  \[
  D_{Manh}(f, g) = \ell_1(f, g) = \sum_{i=1}^{N} |f_i - g_i|
  \]  
  (2.7)

- **Euclidean metric**
  The Euclidean, or $\ell_2$ metric is obtained as:
  \[
  D_{Eucl}(f, g) = \ell_2(f, g) = \sqrt{\sum_{i=1}^{N} (f_i - g_i)^2}
  \]  
  (2.8)

- **Chebychev metric**
  The Chebycheve (Maximum), or $\ell_\infty$ metric definition is:
  \[
  D_{Cheb}(f, g) = \ell_\infty(f, g) = \max |f_i - g_i|
  \]  
  (2.9)

- **Minkowski metric**
  The Minkowski, or $\ell$ metric is the general form of the $\ell_1, \ell_2,$ and the $\ell_\infty$ metrics,
which is defined as:

\[ D_{Min,k}(f, g) = \ell(f, g) = \sqrt[\alpha]{\sum_{i=1}^{N} |f_i - g_i|^\alpha} \]  

(2.10)

where \( \alpha = 1 \) gives the Manhattan (\( \ell_1 \)) metric, \( \alpha = 2 \) yield to the Euclidean (\( \ell_2 \)), and \( \alpha = \infty \) produces the Chebychev (\( \ell_\infty \)) metric.

- **Mahalanobis metric**

  This metric is usually used in pattern recognition and classification problems for finding minimum distance between classes. The metric is defined, using covariance matrix \( C \), as follows:

\[ D_{Maha}(f, g) = \sqrt{(f - g)^t \cdot C^{-1} \cdot (f - g)} \]  

(2.11)

\( C \) is an identity matrix if \( f \) and \( g \) features are independent and consequently, the Mahalanobis metric would be equal to the Euclidean (\( \ell_2 \)) metric. Here, the symbol \(^t\) denotes transposition.

Figure 2.5 shows different contours for constant Chebychev, Manhattan, Euclidean, and Mahalanobis metrics in 2D space [154]. The most common metrics among these family are the Manhattan and the Euclidean metrics [4, 95].

The weighted Manhattan and weighted Euclidean metrics are widely used for ranking in image retrieval [22, 106, 155]. The MPEG-7 standard uses the \( \ell_1 \) metric for com-
paring shape features [124,156], the weighted \( \ell_1 \) metric for feature vectors extracted from color and texture features [63], and the \( \ell_2 \) metric for face recognition [108].

Although the best metric for different features are usually selected empirically, it would be based on the existing emphasis in the metric. For instance, the fact that the differences are squared before summation in the Euclidean metric enforces greater emphasis on those values for which the dissimilarity is large. On the other hand, the Canberra metric, which is defined in Eq. 2.12, is very sensitive to small changes near zero. For each term in the summation, the numerator signifies the difference while the denominator normalizes the difference. Therefore, each term will never exceed one. Thus, the output of the metric ranges from 0 to the number of variables used.

\[
D_{Can}(f, g) = \sum_{i=1}^{N} \frac{|f_i - g_i|}{|f_i| + |g_i|} \tag{2.12}
\]

Consequently, if small values in the vectors are more important than large values, the Canberra metric may produce better performance than the Euclidean distance.

The correlation measure has been used for comparing vicinity tables in [106]. The measure is the covariance, divided by the variances. It is defined by:

\[
D_{Corr}(f, g) = \frac{\sum_{i=1}^{N} (f_i - \bar{f})(g_i - \bar{g})}{\sqrt{\sum_{i=1}^{N} (f_i - \bar{f})^2 \sum_{i=1}^{N} (g_i - \bar{g})^2}} \tag{2.13}
\]

where \( \bar{f} \) and \( \bar{g} \) are the average values of \( f \) and \( g \) vectors respectively. This similarity measure takes values between -1 (the least correlated) and 1 (the most correlated) and gives the cosine of the angle between the two feature vectors measured from the mean. The amount of correlation between corresponding query image and the target image in the database has been employed in [106] to reduce the search space.

Similarly, autocorrelation measure is defined as:

\[
D_{Auto-Corr}(f, k) = \frac{\sum_{i=1}^{N} (f_i - \bar{f})(f_{i+k} - \bar{f})}{\sum_{i=1}^{N} (f_i - \bar{f})^2} \tag{2.14}
\]

That is the correlation between a feature vector and a lagged version of itself. A high correlation is likely to indicate a periodicity in the corresponding feature. If the autocorrelation is calculated for all lags \( (k = 0, 1, 2, \ldots, N - 1) \), the resulting series is called the autocorrelation series or the correlogram. Color correlogram is used for
image indexing in [31] and edge orientation correlogram is employed in [157] for edge image comparison and matching.

Given two sets of points $A = \{a_1, a_2, \ldots, a_p\}$ and $B = \{b_1, b_2, \ldots, b_q\}$, the Hausdorff distance is defined by:

$$H(A, B) = \max(h(A, B), h(B, A))$$

(2.15)

where $h(A, B) = \max_i \min_j D(a_i, b_j)$ and $D(\cdot, \cdot)$ is any underlying distance function such as Manhattan or Euclidean metrics. As the Hausdorff distance measures the similarity of two sets of points, it can be applied to determine the extend to which one image resembles another. An efficient method for computing the Hausdorff distance is proposed in [158]. Huttenlocher et al. [159] compared the Hausdorff distance with the binary correlation method on edge maps and conclude that the former performs better. They also provide algorithms for computing the Hausdorff distance between all possible relative positions of a binary image and a translated model of the image [160].

Several similarity measures for image matching are reviewed and classified in [161]. Several shape similarity measures are compared against human perceptual judgments in [162], and the authors conclude that some tested measures are far from human perception. Moreover, other similarity measures have been developed for specific applications that abandon the strict metric axiom. These metrics/measures are tested usually in limited environments. Although they produce better retrieval performance than the conventional metrics, their benefits are limited to that specific areas and they could not be used broadly. For example, the similarity between two color histograms, $H^X$ and $H^Y$ can be measured using the normalized intersection as follows:

$$\frac{\sum_{i=1}^{N} (H^X_i, H^Y_i)}{\sum_{i=1}^{N} H^Y_i}$$

(2.16)

Although it has been shown that this measure is reasonably insensitive to image resolution, histogram size, viewpoint, depth and occlusion [34, 51, 163], it does not consider the perceptual similarity between the different bins. A metric which takes into account the similarity between the bins is defined by:

$$\sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} [H_i^X - H_i^Y][H_j^X - H_j^Y]$$

(2.17)
where the weights $a_{ij}$ denote the cross correlation between the colors corresponding to bins $i$ and $j$. Although the metric in Eq. 2.17 has a higher computational complexity than the histogram intersection (Eq. 2.16), it is closer to human judgment for color similarity [1]. Jain and Vailaya [34] use color histogram intersection metric in three different channels for measuring the similarity between images. They also apply a similar metric for edge histogram. Their work has a great influence on later researchers in this field.

**Weighted mean variance** technique is proposed in [60] for measuring the similarity in texture-based image retrieval using Gabor filter bank. The distance between two images $I$ and $J$ is defined by:

$$D_{WMV}(I, J) = \sum_m \sum_n d_{mn}(I, J)$$ (2.18)

where

$$d_{mn}(I, J) = \left| \frac{\mu^I_{mn} - \mu^J_{mn}}{\sigma(\mu_{mn})} \right| + \left| \frac{\sigma^I_{mn} - \sigma^J_{mn}}{\sigma(\mu_{mn})} \right|$$ (2.19)

In this approach a Gabor filter bank with 4 scales and 6 orientations is used to produce $6 \times 4 = 24$ filtered images. The mean and standard deviation of each filtered image are calculated and taken as a feature vector $[\mu_{00}, \mu_{01}, \ldots, \mu_{35}, \sigma_{00}, \sigma_{01}, \ldots, \sigma_{35}]$ for each image $I$ and $J$ in Eq. 2.19. $\sigma(\mu_{mn})$ and $\sigma(\sigma_{mn})$ stand for the standard deviation of the respective features over the entire database. They are employed to normalize the individual feature components, which increases the robustness and improve retrieval performance. Kokare et al. [164] have evaluated nine different distance metrics using a texture database of 1856 images. The weighted mean variance distance outperforms the Manhattan, Euclidean, Chebychev, and Mahalanobis metrics in that test, while the Canberra metric performs better than the weighted mean variance measure.

We have proposed a distance measure for ranking database images based on features extracted from edge pixels neighborhood information [106]. The proposed technique results in better retrieval performance than the Hausdorff, Euclidean, Manhattan, and weighted Manhattan measures. Here, the goal is to find a similarity factor between a sketched query image $q$ and images in the database ($I_d$), based on a neighborhood vector $f$. More precisely, suppose $f_q$ is the neighborhood vector for query image $q$. 
and \( f_d \) for a database image \( I_d \). To find the similarity between \( q \) and \( I_d \) we define a measure \( \mu(q, I_d) \) as:

\[
\mu(q, I_d) = \sum_{i=1}^{N} k_i
\]

and

\[
k_i = \begin{cases} 
  f_d^i - \theta & \text{if } (f_d^i \geq \theta) \land (f_q^i < \theta), \\
  f_q^i - \theta & \text{if } (f_d^i < \theta) \land (f_q^i \geq \theta), \\
  0 & \text{else}
\end{cases}
\]

where \( \theta \) is a constant value and \( f_d^i \) are the elements of the vector \( f \). The measure depends on \( \theta \), and \( \theta \) is computed for different databases differently. As the database becomes larger finding the \( \theta \) constant which converges for all entries becomes more challenging.

### 2.7 Retrieval Performance

In this section the issue of evaluating retrieval performance is addressed. Retrieval performance is normally measured using two parameters: (a) retrieval speed, which is usually referred to as **efficiency**, and (b) retrieval accuracy, which is referred to as **effectiveness**. The meaning of speed is clear, and the higher it is, the better the performance of the CBIR system. All image retrieval systems try to reduce search space, employ operative indexing and similarity measure schemes to increase speed and improve the efficiency. Efficiency is usually reported by a time unit. It is the duration needed to produce the final results.

The indexing scheme and the similarity measure together with rotation, translation, and scaling invariant features determine the rate of successful search (effectiveness) \[10, 23\]. The way of reporting the effectiveness varies greatly in different systems. Several performance evaluation techniques in CBIR are reviewed by Muller *et al.* \[165\]. Considering search for exact match or search for similar matches, we can divide performance evaluation methods into two categories. These are discussed in the following subsections.
2.7.1 Exact Match

There are applications where a query image \( q \) is submitted by the user to find a particular match image \( \hat{q} \) within the database. This is referred to as \textit{exact match}. For example, in an art gallery search, the user may pose a hand-drawn rough sketch of the \textit{Mona Lisa Smile} image and intends to see if there exist any image of that painting in the gallery. The original work of Hirata and Kato (QVE) [9], which is adapted in IBM’s QBIC system [94], falls in this category. Here, the performance of the system is measured using the following \( R_n \) measure. This criterion shows the ratio to retrieve the original painting in the best \( n \)-candidates.

\[
R_n = \frac{n_q}{Q_{\text{total}}} \times 100
\]

where \( n_q \) is the number of queries which resulted in the retrieval of the target image in the first \( n \) ranks, and \( Q_{\text{total}} \) is the total number of queries. For example, suppose there exist 18 queries to evaluate a system, if 12 of them find the original images in the top three retrievals, and 15 of them find the original images in the top ten retrievals, then \( R_3 = 66.7\% \ (12/18 \times 100) \) and/or \( R_{10} = 83.3\% \ (15/18 \times 100) \) are reported to exhibit the retrieval performance of the system.

Several \( R_n \) (e.g. \( n = 1, 2, 3, 5, 10, \ldots \)) are required for evaluating a retrieval system. This causes ambiguity when comparing a number of retrieval systems. For this reason, and as comparing different systems with one singular value is much simpler and more understandable, we propose a score-based technique, which assigns a percentage value \( \eta \) to an image retrieval system (exact match) based on its ability to find the original images [106].

To describe this, let \( M \) be the total number of images in the database and \( \hat{q} \) be an image among them which is supposed to be found when providing the query image \( q \). Sorting images, using any similarity measure in descending order, will generally put \( \hat{q} \) at row \( k \) where \( 0 \leq k \leq M-1 \). For example, \( k = 0 \) means that the requested image has been found in the first retrieval (rank 1), smaller \( k \)’s mean better performance. The score assigning scheme is then defined by:

\[
S_q = \frac{M - k}{M}
\]
For each query $q$ there is a $S_q$, and for a set of $q$’s there is a set of $S_q$’s. Therefore, the overall effectiveness of the underlying test system should be considered upon a set of $S_q$’s as:

$$\eta = \frac{\sum S_q}{Q_{total}} \times 100$$ (2.22)

A large $\eta$ indicates a good ability to find the most similar answers (exact matches) to the given queries. The $\eta$ parameter can serve as an evaluation tool for the retrieval systems where the exact match is of interest.

### 2.7.2 Similar Matches

In contrast to the exact match, most CBIR systems look for a range of similar images to a given query. Here, for evaluation, we need to know the ground truth of any submitted query as a priori information. This means that for each query image, all relevant (similar) images must be known for the test of effectiveness of the method.

A simple performance evaluation measure called Bull’s Eye Performance (BEP) is used in [82]. The measure is defined by:

$$\text{BEP} = \frac{\text{numbe of relevants among top } 2N \text{ retrievals}}{\text{total number of relevants}}$$ (2.23)

where $N$ is the number of relevant images.

Inherited from IR (information retrieval) researchers, Recall and Precision are two common performance evaluation measures used in CBIR. They are usually presented as a Precision-Recall graph [72, 83, 165].

The Recall factor measures the ability of the system to retrieve all images that are relevant. It demonstrates the robustness of the retrieval process and is defined by [4]:

$$\text{Recall} = \frac{\text{relevant correctly retrieved}}{\text{total relevants}}$$ (2.24)

On the other hand, the Precision factor measures the ability of the system to retrieve only images that are relevant. It presents the accuracy of the retrieval process and is defined by:

$$\text{Precision} = \frac{\text{relevant correctly retrieved}}{\text{total retrieved}}$$ (2.25)
Since the Precision-Recall graph alone may not contain all the desired information, alternative measures based on the Precision and Recall factors are defined. For example, Mean Average Precision, Recall at 0.5 Precision, and \( R(n) \), which is the recall after \( n \) images have been retrieved, have been used in the literature [165].

Although the Recall and Precision are well known retrieval performance measures, they are basically “hit-and-miss” counters. In other words, the retrieval performance is based on the number of retrieved images which have similarity measures that are greater than a given threshold. For more specific comparisons, however we also need rank information among the retrieved images. Modified versions of Recall and Precision factors, which consider the rank information, are defined in [166]. They are employed in [167] for evaluating image retrieval methods. Here, each query image \( q \) has three relevant images \( q_1, q_2, \) and \( q_3 \). The two measure are defined as: (a) Recall Rate:

\[
\mu(q) = |\{q_i \mid \text{rank}(q_i) \leq 12, \ i = 1, 2, 3\}|
\]  

and (b) Precision:

\[
\rho(q) = \frac{1}{466} \sum_{i=1}^{3} \phi(\text{rank}(q_i))
\]  

where function \( \phi(\text{rank}(q_i)) = 0 \) when \( \text{rank}(q_i) > 12, \phi(12) = 1, \phi(11) = 2, \) and \( \phi(i) = \phi(i + 1) + \phi(i + 2) \) for \( i = 10, \ldots, 1 \). The precision measure in this modified definition is based on the classical Fibonacci sequence. The denominator 466, which is \( \phi(1) + \phi(2) + \phi(3) \), is used for normalization. When there are several queries, the average of \( \mu \) and \( \rho \) (\( \bar{\mu} \) and \( \bar{\rho} \)) are obtained and reported as evaluation factors.

More generally, the Average Normalized Modified Retrieval Rank \(^1\) (ANMRR), which was developed during the MPEG-7 standardization activity, is a measure that exploits not only the Recall and Precision information but also the rank information among the retrieved images. It is defined in the MPEG-7 standard [168] as follows:

\[
AVR(q) = \sum_{k=1}^{NG(q)} \frac{\text{Rank}(k)}{NG(q)}
\]  

\[
MRR(q) = AVR(q) - 0.5 - \frac{NG(q)}{2}
\]

\(^1\) Occasionally referred to as: Average Normalized Modified Retrieval Rate, e.g. in [108].
\[
NMRR(q) = \frac{MRR(q)}{L + 0.5 - 0.5 \times NG(q)}
\] (2.30)

\[
ANMRR = \frac{1}{Q_{total}} \sum_{q=1}^{Q_{total}} NMRR(q)
\] (2.31)

where \(NG(q)\) is the number of ground truth images for a query \(q\). \(L = \min(4 \times NG(q), 2 \times GTM)\), where \(GTM\) is \(\max\{NG(q)\}\) for all \(q\)'s of a data set. \(Rank(k)\) is the rank of the found ground truth images, counting the rank of the first retrieved image as one. A \(Rank\) of \(L + 1\) is assigned to each of the ground truth images which are not in the first \(L\) retrievals. \(Q_{total}\) is the total number of queries in the test.

For example, suppose a given query \(q_i\) has 10 similar images in an image database (\(NG = 10\)). If an algorithm finds 6 of them in the top 20 retrievals (\(L = 20\)) in the ranks of 1, 5, 8, 13, 14, and 18, then the \(AVR(q_i) = 14.3\), \(MRR(q_i) = 8.8\), and \(NMRR(q_i) = 0.5677\). Note that NMRR and its average (ANMRR) will always be in the range of \([0, 1]\). Based on the definition of ANMRR, the smaller the ANMRR, the better the retrieval performance.

The ANMRR measure will be used in subsequent chapters whenever we need to evaluate the retrieval performance of different methods dealing with similar matches.

### 2.8 Semantics in Image Retrieval

Incorporating semantics in visual information retrieval is one of the most active research areas in the growing multimedia technology. The main goal is to align the results of search engines, as much as possible, to human expectations. Low level features, while effective in particularly focused applications, are semantically primitive when compared to human similarity judgment [169]. Combining image features and inventing new models much closer to human perception have been studied to achieve the goal. Bridging the gap between human perception and low level features in visual information retrieval has been addressed in different directions, for example, by applying neural networks, fuzzy methods, relevance feedback and by combining content-based and text-based approaches. In this section, some representative approaches are presented.
A nonlinear relationship between image features, such as color, texture and shape, based on radial basis function network is suggested in [170]. The main focus in this approach is on human-computer interaction in image retrieval. An interactive image retrieval system based on back-propagation neural networks is designed in [171]. Vertan and Boujemaa [172] concentrate on possible embedding of uncertainty in image retrieval using image colors and histogram-based descriptors. The uncertainty caused by quantization of the color components and the uncertainty caused by the human perception of colors is addressed. They define a set of fuzzy color histograms to solve the problem.

In [173] a fuzzy model for considering multiple attributes (color, shape, texture and spatial relation) in an image is presented. Fuzzy sets is used to model the imprecision and vagueness of objects in an image. Each object in an image is represented by a node with multiple attributes. The approach takes the image retrieval problem as a sub-graph matching problem. Wang et al. [174] applies the fuzzy integral for retrieving leaf images. In this application, the fuzzy combination of different shape-based feature sets is used in the retrieval task. The centroid-contour distance curve, eccentricity and the angle code histogram are the three features used in this study.

The fuzzy information retrieval model for textual documents and the similarity-based query model are extended to implement a new fuzzy model in the image retrieval context [175]. The proposed model consists of image representation, query representation, and query processing divisions. Han and Ma [176] introduce the fuzzy color histogram concept. The approach is considering the color similarity of each pixel’s color through a fuzzy-set membership function. This is in contrast with conventional color histogram which considers neither the color similarity across different bins nor the color dissimilarity in the same bin.

An image retrieval method based on human perceptual clustering of color images is proposed in [166]. This color clustering technique produces, for each image, a small set of representative colors. These colors capture the chroma properties of the image. For this, a small set of sizable contiguous regions which captures the spatial/geometrical properties of the image is employed.
To improve the semantic power of visual information retrieval, a multi-layered framework is developed in [169]. The framework integrates current works on content-based information retrieval at various levels of abstractions. The layered architecture includes data layer, feature layer, object layer and semantic layer. Retrieval is suggested to be carried out at any layer. The authors propose the use of relevance feedback at different layers to improve retrieval performance. At each level, the appropriate amount of domain knowledge should be used to include semantics. Each level provides an abstraction to the level above, so that further changes can be made with little side effects.

Colombo et al. [177] apply expert’s knowledge in the form of a set of linguistic rules to incorporate semantics to visual information retrieval. The adage which says “an image is worth a thousand words” is criticized and the paper claims that the converse also holds true sometimes. Different levels of signification using a layered representation of visual knowledge is encouraged in this approach. Defining general rules that capture visual meaning is not a simple task, those rules should be derived from specific domain characterizations and possibly be refined and tailored to specific classes of users and applications.

Almost always, the overall performance of any retrieval system can be improved using relevance feedback, that is employing human in the loop to give negative or positive feedback about the retrieved results. Zhou and Huang [26] recently reviewed the relevance feedback topic in image retrieval. The overall structure of a typical algorithm can be summarized in three main steps:

1. Initial retrieval results, generated by query-by-keyword, sketch or example, are provided to the user.

2. A judgment on the currently displayed images as to whether, and to what degree, they are relevant or irrelevant, is given by the user.

3. The algorithm is refined (e.g. machine learns) and tries again (goto 2 )

In [178], a relevance feedback method is proposed to improve the retrieval performance of the MPEG-7 edge histogram descriptor. It combines two previous rele-
vance feedback techniques namely, query point movement and term weighting [163]. The first method essentially tries to improve the estimate of the ideal query point by considering the user’s choice of the relevant and irrelevant images among the retrieved images. The term weighting method, on the other hand, tries to improve the relative importance of the feature values in the similarity measure.

Intermediate features that are the combination of low-level semantic features and high-level image features are used in [179]. Six intermediate features including sky, water, snow, rock, vegetation, and sand are defined in the paper. They can be arranged to produce high-level concepts. The system is trained to learn these intermediate features from a small-annotated database. In this study, the Precision and Recall performance criterions are used for the evaluation of the image retrieval techniques, and it is concluded that the new approach is superior to the standard color histogram indexing method.

A learning-based similarity measure scheme is proposed in [180]. It learns a boundary that separates the images in the database into two parts. Images that lie on the promising side of the boundary are then ranked by the Euclidean distance. The scheme takes into account the perceptual similarity between images and improves the retrieval performance. Support vector machines and AdaBoost techniques are employed in the learning process.

We examined the semantic power of five different low and middle level approaches by a subjective test [64]. The approaches which passed through the test are: (a) invariant moments, (b) MPEG-7 edge histogram descriptor (EHD), (c) histogram of edge directions (HED), (d) the correlation method (query by visual example), and (e) the proposed edge pixel neighborhood histogram (EPNH) method. The approaches are employed to produce 10 ranked images of similar logos to 30 different image queries. We used the S8 part of the MPEG-7 still images content set for the test. It contains 2810 different black and white logo and trademark images. Ranked results, generated by different approaches, are submitted to 17 subjects to evaluate and give a mark between 1 to 10 to each method based on their similarity to the query image. As the result, the correlation method, which is the slowest method, obtained the best closeness to human judgment in this study. Next in the rank were, the EPNH, the
EHD, the HED, and the invariant moments methods respectively based on the human judgment. These latter methods have reasonable speed (efficiency) compare to the correlation method.

### 2.9 Content-Based Image Retrieval Systems

Since the early 1990’s, many image retrieval systems including commercial, demonstration, and research ones, have been built. A number of review papers on image database systems or multimedia information systems have been published [4, 10, 27, 29, 148]. They investigate technical aspects of existing systems including salient features, indexing technique, querying method, similarity measure, relevance feedback, and result presentation.

Text-based search engines on the web (Internet) sometimes allow us to indicate the media type is an image. Google Image Search (http://www.google.com.au/images/), HotBot (http://hotbot.lycos.com/), and NBCi (http://www.nbci.com/) are examples of these. A number of search engines are more specifically designed for images retrieval using the image content for the retrieval task. A directory of most active web-based image search engines, along with links to their web sites, are provided in the Big Search Engine Index (http://www.search-engine-index.co.uk/ImagesSearch/). AltaVista Photofinder (http://www.altavista.com/image/default), Yahoo!’s Picture Gallery (http://gallery.yahoo.com/), Corbis (http://pro.corbis.com/), and Lycos Image Search (http://multimedia.lycos.com/) are some representative web-based image search engines. The prominent aspects of these systems are explained and compared in [27].

Most image retrieval systems support one or more of the following facilities:

- search by text
- search by example
- search by sketch
- random browsing, and
customized navigation

Although there exists a rich set of search facilities today, systematic studies involving actual usage in practical/specific applications still need to be completed. Based on the huge diversity of users, more demands will be on more specific image search engines in the future.

In this section, we first explore a general structure of CBIR systems and then introduce some representative systems and highlight their distinct characteristics.

2.9.1 General Structure

Image retrieval systems, including CBIR systems and text-based systems, can be described conceptually, by the general structure depicted in Figure 2.6. The user interface (UI), which is typically consists of a query formulation part and a result presentation (visualization) part, is the front page of most systems dealing with input and output.

Image requesting is summarized in the following three major ways [27]:

- **direct query**: is to specify the query in terms of keywords (text-based), or in terms of image features that are extracted from the image, such as a color histogram or a texture descriptor.

- **query by example**: is to provide an image or sketch (in the form of scanned, drawn, or computer file) from which features of the same type must be extracted as for the database images. This is to find out a similarity measure between corresponding features.

- **browsing**: is to explore through the database or a portion of the database one by one. This alternative is usually employed in systems which support relevance feedback, so that the system can refine the search again and again.

*Feature extraction* from the database images is accomplished off-line at the population time, while for the query this is an on-line process at the request time. The
Figure 2.6 General structure of image retrieval systems

*matching process* does similarity measuring and the necessary comparisons. The indexes of those images which are selected to retrieve are passed into the *image pointers* process. It obtains image pointers (image id’s), and the *fetching process* physically retrieves the images from the database. The resulting images are queued through the *visualization part* of the UI to the user.

As is discussed in Section 2.4, several classes of features, or their combinations, can be used in CBIR. Commercial systems usually use a combination of content-based features together with text-based features to improve performance. In the next
subsection, some representative systems are investigated.

2.9.2 Representative Systems

There are several CBIR systems in the market as commercial products or as research projects. Many are created and improved continuously. Here, we present a few well known image retrieval systems exploring their salient characteristics.

2.9.2.1 VIR Image Engine

VIR Image Engine from Virage Inc. is a known commercial system. Histogram and graph are used as two basic abstract data types. The search engine is available as a series of independent modules. System developers can build these modules into their own programs. This facilitates the extension of a system by building new types of query interfaces. Adding customized modules such as processing specialized collection of images (e.g. trademarks) is possible too. Alternatively, the system is available as an add-on to existing DBMS’s such as the Oracle or Informix [181].

The query canvas, which allows query-by-sketch, is provided in the graphical user interface. It consists of a bitmap editor where the user can sketch a picture with existing drawing tools and color it using the colors from a palette. Also, the user can bring onto the canvas a desired image from an existing album and modify it using the same drawing tools. Relevance feedback is also employed. An application of Virage technology is AltaVista Photofinder (see Section 2.9.2.8). Virage technology has also been extended to the management of video data [182]. Details of the Virage commercial products can be found on the Web at http://www.virage.com/.

2.9.2.2 QBIC

IBM’s QBIC [50,94] is the first commercial CBIR system. It is by far the most cited system and its framework and techniques have profound effect on most later image retrieval systems. It is available commercially either in stand-alone form, or as part of other IBM products such as the DB2 Digital library.

It offers image retrieval by any combination of color, texture, shape, and keyword.
Color features employed are average color in different color space (for the whole image or for an object), and color histogram in the Munsell coordinates. Its texture features are a modified version of the Tamura’s model. It is composed of coarseness, contrast, and directionality attributes. The shape features consist of area, circularity, eccentricity, major axis orientation and a set of algebraic moment invariants. For sketch queries, the QBIC system uses a modified version of the QVE proposed by Hirata and Kato [9]. The details of the QVE technique are discussed in Chapter 3 of this thesis.

The QBIC system uses KLT to perform dimension reduction and employs $R^*$-tree indexes to construct multidimensional indexing, which slightly improves the search efficiency. The resulting images are presented in decreasing similarity order. There is an option to show the matching score at the output list. Any retrieved image can be applied as a seed for a subsequent query by example. http://www.qbic.almaden.ibm.com/ currently describes the latest developments and new applications of the QBIC system.

### 2.9.2.3 VisualSEEk and WebSEEk

VisualSEEk [38], a visual feature search engine, and WebSEEk [183], a web oriented text/image search engine, have both been developed at the Colombia University, New York. The two main features of these are: (a) using spatial relationships of image regions, and (b) visual feature extraction from compressed domain. Color, texture, shape and keywords are also exploited for image retrieval. 2D strings and quad-trees [184] data structures using the method proposed in [91] are exploited in this system to represent object layout. To speed up the retrieval process, binary tree-based algorithms are developed for indexing. The system supports queries based on both visual content and keywords.

Further prototype systems from this group include the VideoQ [185, 186], a video search engine which allows users to submit motion queries, and the MetaSEEk [39], which is a meta-search engine for images on the web. A demonstration version of the VisualSEEk and WebSEEk is provided in http://www.ctr.columbia.edu/VisualSEEk/, and http://www.ctr.columbia.edu/webseek/ respectively. An exhibiting demonstra-
A typical VisualSEEk query interface is depicted in Figure 2.7-(a). The fifteen top results of the WebSEEk system for query keyword “lion” is shown in Figure 2.7-(b). Occasionally, there is no logical relationship between some found images and the submitted keyword. This is, most probably, due to an irrelevant caption of the image. Relevance feedback is supported to refine the search and improve user satisfaction.

2.9.2.4 MARS

Multimedia Analysis and Retrieval System (MARS) was initiated at the University of Illinois at Urbana-Champaign and further developed at the University of California at Irvine, California [51, 163]. The MARS differs from other systems in both the research scope and the techniques used as it is an interdisciplinary system. Multiple research communities including image processing, computer vision, DBMS, and IR are involved in this project. It integrates research directions on indexing, exact match,
ranked retrievals, and human-machine interface.

The main focus of MARS is on how to organize various features into a meaningful retrieval design in contrast with finding a single best feature for a large collection of images. It supports a combination of low-level features (color, texture, and shape) and text-based descriptions. The image is divided into $5 \times 5$ sub-images first and then color and texture characteristics are obtained for each subimage.

Simple queries can be specified either by example, or pointing an image database, or directly by choosing desired texture and color attributes. Boolean operators can be applied to generate complex queries. Text-based queries are accomplished by filling several meta-data fields provided at a query interface page.

The database consists of images of ancient African artifacts from the Fowler Museum. Latest information regarding MARS can be obtained from http://www-db.ics.uci.edu/pages/research/mars.shtml/. The demonstration version is available at http://www-db.ics.uci.edu/pages/demos/index.shtml/. Figure 2.8 shows a list of images similar to the shape of which located at the upper-left position. Any of the retrieved images could then be selected as another query.

Figure 2.8 An example of MARS retrieval result, the upper-left shape is the input query
2.9.2.5 NETRA

Netra is a prototype image retrieval system developed at the University of California, Santa Barbara [61, 62]. An initial prototype of the Netra system is used in Alexandria Digital Library (ADL). At first, it segments images into regions of homogeneous color. Of those regions then, color, texture, shape, and spatial location features are extracted. Main research aspects of the Netra system are: (a) its Gabor filter based texture analysis, (b) neural network based image thesaurus construction, and (c) edge flow based region segmentation.

There are 25 different categories for 2500 images in the database (100 images in each category). Any of the images in the database can be selected as the query example. The user can click on one of the regions and select one of the four image characteristics: color, texture, shape, and spatial location. Alternatively, the user can directly specify the color and spatial location. The region of interest can be defined using two bounding boxes provided in a spatial location querying tool. The preferred region is determined by the inner box while the outer box is used to constrain the object to be inside the region. Therefore, if the selected object contains any part exceeding the outer box, they will not be considered as the query.

The Euclidian distance is used as the metric for color and shape attributes whereas the Manhattan distance is applied for measuring the similarity of texture related vectors. Color, texture and shape are indexed separately. The first attribute specified by the user is used to retrieve about 100 image candidates. This attribute and the possible other attributes are then being exploited to rank the retrieved images. The latest works on the Netra system such as Netra2 and Netra V, an object-based video representation project, have been introduced on the web at location http://vision.ece.ucsb.edu/netra/.

2.9.2.6 Photobook

The Photobook system [76], developed at the MIT Media Lab, is a set of interactive tools for searching and browsing images. It consists of three subsystems from which shape, texture, and face features are extracted. To submit a query, the user selects
one or several images from the existing set of still images, or enters an annotation filter. When the results are displayed, the user can select another query image(s) and reiterate the querying process. Prior to any database search, indexing is performed by selecting a few prototypes that span an image category. The distance of any image in the database to the average of the prototypes is calculated and stored. When a query is submitted, the distance of the query image to the average is computed and the database is sorted accordingly and presented to the user, page by page.

The FourEyes system, a later version of the Photobook system, considers human intervention into the image annotation phase and the retrieval loop. Here, pre-computed features as well as learning from the user’s interaction are used for performing the search. The focus in this system is on how to best utilize multiple models and which model combination can closely represent the user similarity criteria. The face recognition approach of the Photobook has been employed by the Viisage Technology company in a package called FaceID. This package is used in several United States police departments [27]. Further information on the Photobook system, its applications, and an online demonstration can be found at http://vismod.media.mit.edu/vismod/demos/photobook/. Figure 2.9 depicts two examples of a Photobook search based on shape and texture.
2.9.2.7 Blobworld

The BlobWorld system [187], developed at the University of California, Berkeley, is one of the well known retrieval systems that uses image region to improve retrieval performance. It provides a transformation from raw pixel data to a small set of localized coherent regions in a color and texture space. The user can view the internal representation of the submitted query and the resulting images; it enables the user to know why some non-similar images are returned and can therefore modify his/her query accordingly.

At first, the user selects a category, which limits the search space. Then, in an initial query, the user highlights a region (called blob), and intimates the importance of that blob by “Not”, “Somewhat”, or “Very” indicators. Finally, the user specifies the importance of the blob’s color, texture, location and shape/size by the same designators (Not, Somewhat, or Very). More than one blob can be used for querying. In the first version of the Blobworld system, the ellipses are used to symbolize blobs, but in more recent versions, the real boundaries of the region are used to specify a blob. Keyword-based search, as an optional method, is also provided. Here, the system looks for the entered text in keywords, captions, and the CD titles to find out the most appropriate images.

A histogram of 218 bins of the color coordinates in the Lab-space is used to describe the color feature. The retrieved images are sorted based on overall similarity and presented together with the segmented version showing the region. Highlighting the regions which are responsible for the retrieval facilitates the use of relevance feedback. Despite the fact that the segmentation might not always conform to what the user is looking for, it provides a simple way to specify initial queries.

Figure 2.10 shows an example Blobworld query result page. Latest information on the system can be found at http://elib.cs.berkeley.edu/photos/blobworld/. An online demonstration is provided at http://elib.cs.berkeley.edu/photos/blobworld/start.html on the web.
2.9.2.8 AltaVista Photofinder

AltaVista Photofinder is one of the most famous web-based search engines. It allows a confined content-based image retrieval, both from special collections, and from the web. It was originally developed at DEC Research Lab, and is now run by Altavista Company. Although no details are given about the exact features used, similarity is based on visual characteristics such as color, texture, and shape.

The user first types one or more keywords to search for images have these words in their file names, ALT text (a piece of text associated with images by the system), nearby text, and/or the page metatags. The retrieved images are shown to the user as thumbnails, without any explicit order. All image results have a More Info page that gives more details about that specific image. The results can be limited to Photos or Graphics. The Photos and Graphics settings choose images based on their appearance. It is also possible to choose between color and grayscale images. The system is built on the VIR Image Engine, (described in Section 2.9.2.1). Currently, the AltaVista image search engine is located at http://www.altavista.com/image/default.
2.10 Chapter Summary and Conclusion

Two major approaches in image retrieval: text-based, and content-based, have been reviewed in this chapter. Focus is placed on recent trends in the content-based approach. Several works dealing with content features, similarity measures, performance evaluation, and semantic issues have been explained, summarized, and classified. Representative CBIR systems have been addressed in the last section.

The SBIR has been recognized as an interesting topic for further research, specifically when the query image has no color and texture attributes (i.e. rough sketch images). This is due to the significance of this topic in new image retrieval systems and since only a few works have addressed the problem. The aim of this thesis is to fill this gap by providing efficient and effective techniques for the SBIR, facilitating rough sketched queries for search in regular image databases. This topic will be discussed with more details in the subsequent chapters.
Chapter 3

Sketch-Based Image Retrieval (SBIR)- Applicable Methods

3.1 Introduction

While the general problem of content-based image retrieval (CBIR) has received a great deal of attention, not many works have addressed sketch-based image retrieval (SBIR). Some approaches entitling "sketch-based" have addressed a problem that is different from the one considered here, and some are dealing with images having color and texture attributes as the sketch query. In the following we briefly describe six different approaches which can be used directly (or can be adapted) for the SBIR, where the sketched query, recruited for the retrieval task, has no color and texture attributes. These approaches are implemented, tested, and compared for efficiency (retrieval speed) and effectiveness (retrieval accuracy) in the subsequent chapters whenever applicable. Some techniques are suitable only for the shape retrieval, e.g. the invariant moments, while some approaches, such as the query by visual example (QVE), are useful for the general image retrieval.

This chapter is organized as follows: Section 3.2 describes the QVE method and Section 3.3 explains the histogram of edge direction technique. The invariant moments are discussed in Section 3.4 followed by the MPEG-7 edge histogram descriptor and angular radial transform methods which are explained in Sections 3.5 and 3.6, respectively. Finally, the polar Fourier descriptors are described in Section 3.7. The
last section presents a chapter summary and conclusion.

3.2 Query by Visual Example (QVE)

The work of Hirata and Kato, Query by Visual Example (QVE) [9], is one of the earliest approaches that address SBIR. IBM corporation adapts a modified version of the approach in its QBIC system [94]. It defines a pictorial index for each image, including query and database images. The QVE method performs image retrieval by computing the correlation between the corresponding indexes extracted from the original image and the sketched query [9, 188].

Here, a contour image, which roughly approximates the general composition of the original image, is referred as pictorial index. An adaptive differential filter, based on the Weber-Fechner law of human vision mechanism (briefly explained in the following), on RGB space is used to extract edge points for abstraction.

**Weber-Fechner Theory** Human beings can distinguish a change in the magnitude of a stimulus based on the current magnitude of that stimulus. For example, it is easy to distinguish a 1-kg weight from a 2-kg weight, but it is difficult to distinguish a 11-kg weight from a 12-kg weight. In the 1800s, Weber developed a quantitative description of the relationship between stimulus intensity and discrimination now known as Webers law, namely

$$\Delta S = K \cdot S$$

(3.1)

where $\Delta S$ is the perceived intensity difference relative to a background stimulus $S$ and $K$ is a constant. Fechner applied Webers law to sensory experience in the 1860s. He found that the intensity of a sensation is proportional to the logarithm of the strength of the stimulus

$$I = K \cdot \log \frac{S}{T}$$

(3.2)

where $I$ is the subjective experienced intensity, $T$ is the threshold, or minimally perceptible stimulus level, and $S(>T)$ is the strength of the supra-threshold stimulus. This relationship is referred to as the Weber-Fechner law [138].

The algorithm for full-color image abstraction is outlined in the following section.
3.2.1 Abstraction Algorithm

An abstract image is produced through the following steps [188]:

- **Normalization**: apply the affine transformation and the median filter to the input matrix (RGB intensity values) to obtain its regular-sized (64 × 64 pixels) image.

- **Gradient in the RGB space**: calculate the normalized gradients of the RGB intensity values, \( \partial_{ij} \), for each pixel \( p_{ij} \) on the regular-size image using:

\[
\partial_{ij} = \frac{|\Delta p_{ij}|}{|I_{ij}|}
\]  

(3.3)

where, \( |\Delta p_{ij}| \) is the local maximum difference and \( |I_{ij}| \) is the local power of the intensity values.

- **Global edge candidates**: select the pixels of large gradient values as global edge candidates using the following criterion:

\[
\partial_{ij} \geq \mu + \sigma
\]  

(3.4)

where, \( \mu \) and \( \sigma \) are the average and the standard deviation of the gradients of the regular-sized image.

- **Local edge candidates**: calculate the local average and the local standard deviation of the gradient values, i.e. \( \mu_{ij} \) and \( \sigma_{ij} \), only for each global edge candidate. Select the local edge candidates which satisfy the following condition:

\[
\partial_{ij} \geq \mu_{ij} + \sigma_{ij}
\]  

(3.5)

- **Abstract image**: apply a thinning procedure and a shrinking procedure to the local edge image to get the abstract image.

The extracted abstract image is used as the pictorial index for the original images. The binarized and thinned rough sketch which is normalized to regular size is called linear sketch and is used for the retrieval algorithm. The algorithm is given in the following section.
3.2.2 Retrieval Algorithm

Local and global correlations are used for the SBIR. The following is the outline of the algorithm used for matching [9]:

- **Local block:** divide an abstract image \( P_t = \{ p_{ij} \} \) and a linear sketch \( Q = \{ q_{ij} \} \) into \( 8 \times 8 \) local blocks. The size of local blocks is \( 8 \times 8 \) since the image regular size is \( 64 \times 64 \) (see Figure 3.1).

- **Local correlation:** calculate the correlation \( C_{\delta \varepsilon}^{ab} \) between the local blocks \( P_t^{ab} \) and \( Q^{ab} \) by the following bit-wise summation with shifting \( Q^{ab} \) by \( \delta \) and \( \varepsilon \) over the corresponding block of \( P_t \):

  \[
  C_{\delta \varepsilon}^{ab} = \sum_s \sum_r (\alpha p_{rs} \cdot q_{r+\delta s+\varepsilon} + \beta p_{rs} \cdot q_{r+\delta s+\varepsilon} + \gamma p_{rs} \oplus q_{r+\delta s+\varepsilon}) \tag{3.6}
  \]

  The \( a \) and \( b \) parameters are used to indicate a unique block in the images. The coefficients \( \alpha, \beta, \) and \( \gamma \) are control parameters used to estimate matching and mismatching patterns and take up values 10, 1 and -3, respectively. The maximum value of \( C_{\delta \varepsilon} \) for all \( \delta \) and \( \varepsilon \) for each block is called the local correlation factor \( C_L^{ab} \):

  \[
  C_L^{ab} = \max(C_{\delta \varepsilon}^{ab}) \text{ for } -4 \leq \delta \leq 4 \text{ and } -4 \leq \varepsilon \leq 4 \tag{3.7}
  \]

- **Global correlation:** calculate the global correlation \( C_t \) between the abstract image \( P_t \) and the linear sketch \( Q \) as a sum of the local correlation values as follows:

  \[
  C_t = \sum_{\alpha=0}^{7} \sum_{\beta=0}^{7} C_L^{ab} \tag{3.8}
  \]

- **Retrieval:** apply the above three steps to every abstract image \( P_t \) on the pictorial index. Sort \( \{ C_t \} \) in the descending order. The first one is the best candidate for the QVE approach.

It is worthwhile to mention that IBM has modified the above algorithm in its QBIC system [94]. Here, the pictorial indexes are computed through the following steps:

- Each color image is converted to a single band luminance.
The binary edge image is computed using the Canny edge operator [102].

Image size is reduced to 64 × 64.

To do the size reduction, the image is partitioned into blocks of size \( w/64 \times h/64 \), where \( w \) is the width of the image and \( h \) is the height in pixels. Next, if any pixel in a partition of the full size edge image is an edge pixel, the corresponding pixel in the reduced edge map is set to an edge pixel. Finally, the reduced image is thinned and used as the abstract image input to the retrieval algorithm.

Although the method has a good ability to find similar images in several applications, it does not allow efficient indexing [9]. Furthermore, because of the expensive computational cost it is time consuming to use the method in large-scale image databases. Most importantly, the method can only tolerate minor local rotations. It is not rotation invariant and does not allow for large global rotations.

### 3.3 Histogram of Edge Directions (HED)

The histogram of edge directions (HED), representing image information, is one of the well known methods in the image retrieval literature [28, 189, 190]. Here, the
emphasis is on a fast yet robust, scheme for image retrieval. Abdel-Mottaleb [101] utilizes this method by applying the Canny edge operator [102] to find strong edges in an image and then quantizes them into 4 directions (horizontal, vertical, and the two diagonals) to build histograms of edge directions for different image regions. The histograms are then used as hash values in a hash table indexing scheme. Jain and Vailaya [34] introduced the use of edge directions as an image attribute for shape description. In the absence of color information or in images with homogenous colors this histogram is a significant tool in searching for similar images. The histogram in conjunction with invariant moments has been employed in a case study of trademark registration [74]. Here, we summarize the technique proposed by Jain and Vailaya [34].

The approach is simply based on the distribution of edge angles. The edge information contained in the database images is extracted off-line using the Canny edge operator. The corresponding edge directions are, subsequently, quantized into \( B \) bins of \( 360\,^\circ /B \) each. The resulting histogram is translation invariant. Thus, the positions of different objects (shapes) in the image have no effect on the distribution of edge directions. On the other hand, the histogram is inherently scale variant. That is, two image identical in every respects except their size, will yield different numbers of edge points and hence different histograms. To overcome the problem and to have scale invariance, the histogram is normalized with respect to the number of edge points in the image. Note that the drawback of normalized histograms is its inability to perform partial matches. More precisely, If an image \( J \) is a part of image \( I \), then the histogram of \( J \) is contained within the histogram of \( I \), while normalizing the histograms does not satisfy this property. This shortcoming is important only if partial matching is of interest.

The histogram of edge directions is also rotation variant. A rotation of the image shifts each of the edge directions by the amount of rotation. Rotation also affects the membership in the bins. To reduce the effect of rotation, the histogram is smoothed as follows:

\[
I_s[i] = \frac{\sum_{j=i-k}^{i+k} I[j]}{2k + 1}
\]  

(3.9)
where \( I_S \) is the smoothed histogram, \( I \) is the original normalized histogram, and the parameter \( k \) determines the degree of smoothing.

Using these image features, the matching results depend on the number of bins in the histogram. By choosing a lower number of bins, the matching speed is increased but this reduces the accuracy in case of an arbitrary rotation of the image. On the other hand, using a higher number of bins reduces the matching speed and also requires that the edge directions be found to a very high degree of accuracy. \( B = 72 \) which equals to the steps of \( 5^\circ \) is usually selected.

### 3.4 Invariant Moments

Invariant moments are widely used in pattern recognition and image analysis [74, 156, 191]. Two main moment approaches are the geometric moments and Zernike moments.

In general, the geometric moments describe an image as a numeric function with respect to a reference axis or frame and defined as:

\[
M_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) \, dx \, dy
\]

where \( p, q = 0, 1, 2, \ldots, \infty \), and \( f(x, y) \) is the density distribution function of the image. It is known that if \( f(x, y) \) is piecewise continuous and has nonzero values only in a finite region of the \( xy \) plane, then moments of all orders exist. Importantly, the moment set \( \{M_{pq}\} \) is uniquely determined by \( f(x, y) \), conversely, \( f(x, y) \) is uniquely determined by \( \{M_{pq}\} \).

The shape of an image could be represented in terms of seven functions defined on invariant moments \( \phi_1 - \phi_7 \). They have been widely used in a number of applications [73, 74, 178]. The first six functions \( \phi_1 - \phi_6 \) are invariant under rotation and the last one \( \phi_7 \) is both skew and rotation invariant. They are based on the central \( i, j \)-th moments \( \mu_{ij} \) of a 2D image \( f(x, y) \), which are defined as follows [4]:

\[
\mu_{ij} = \sum_x \sum_y (x - \bar{x})^i (y - \bar{y})^j f(x, y)
\]
Sketch-Based Image Retrieval (SBIR) - Applicable Methods

where \( x = M_{10}/M_{00} \) and \( y = M_{01}/M_{00} \) are indicating the center of mass (COM) of the shape. Then, defining \( \gamma = (i+j)/2 + 1 \) and \( \eta_{ij} = \mu_{ij}/\mu_{00}^2 \), the invariant functions are obtained by:

\[
\begin{align*}
\phi_1 &= \eta_{20} + \eta_{02} \\
\phi_2 &= (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2 \\
\phi_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
\phi_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
\phi_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \cdot [3(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]
+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \cdot [3(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\
\phi_6 &= (\eta_{20} - \eta_{02})^2 - (\eta_{21} + \eta_{03})^2 + 4\eta_1(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
\phi_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) \cdot [(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]
- (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03}) \cdot [3(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2]
\end{align*}
\]

(3.12)

The \( \phi \) functions make for each image a 7-entry vector, which serves as a feature vector and is used for measuring the similarity between images. Although the retrieval speed is very high due to compact feature vector, there are some disadvantages. First, the geometric moments have the form of projection onto monomial \( x^p y^q \) that is the basis set which is complete but not orthogonal. Second, geometric moments are vulnerable to noise, and for all objects that have \( n \)-fold symmetries they are all zero [124].

Zernike moments, on the other hand, using orthogonal basis functions, are less sensitive to noise than geometric moments and are more powerful in discriminating objects with \( n \)-fold symmetries. They are exploited for building a region-based shape descriptor in the MPEG-7 standard [156]. Zernike orthogonal polynomials are employed to derive Zernike moment invariants of an image \( f(x, y) \) as follows:

\[
A_{nl} = \frac{\eta + 1}{\pi} \int_0^{2\pi} \int_0^\infty [V_n(r, \theta)]^* f(r \cos \theta, r \sin \theta) r \, dr \, d\theta
\]

(3.13)

where \( n = 0, 1, 2, \ldots, \infty \), and \( l \) takes on positive and negative integer values subject to the condition that \( n - |l| \) is even and \( |l| \leq n \). The Zernike polynomial is defined
as:

\[ V_{nl}(x, y) = R_{nl}(r) e^{i\theta} \]  \hspace{1cm} (3.14)

and the radial polynomial is:

\[ R_{nl}(r) = \sum_{s=0}^{(n-|l|)/2} \frac{(-1)^s}{s!(n+|l|)/2-s!} \frac{(n-s)!}{(n-|l|)/2-s!} r^{n-2s} \]

\[ = \sum_{n-k \text{ even}}^{n} \sum_{|l|=n-k}^{n} B_{n,l} r^k \]  \hspace{1cm} (3.15)

If only Zernike moments of order less than or equal to \( N \) are given, then the image function \( f(x, y) \) can be approximated by:

\[ f(x, y) \approx \sum_{n=0}^{N} \sum_{n-|l| \text{ even}, |l| \leq n} A_{nl} V_{nl}(x, y) \]  \hspace{1cm} (3.16)

The magnitudes of \( \{A_{nl}\} \) are used for image matching and the similarity between images is measured using the \( \ell_1 \) (Manhattan) distance [124, 156].

In the case of hand-drawn sketched images, \( f(x, y) \) is a binary image representing the outline of the query. One obvious advantage of the use of binary functions is the low computational complexity. As \( f(x, y) \) is a black and white sketch, computation is not required for every pixel. In fact, computation is dependent on the number of pixels in the sketched query. This approach has been employed for the SBIR in [124] and shows better performance than the traditional 2D Fourier transform method.

### 3.5 MPEG-7 Edge Histogram Descriptor (EHD)

The MPEG-7 standard defines the edge histogram descriptor (EHD) based on edge point distribution for texture characterization [63]. The distribution of edges is not only a good texture signature, it is also useful for image-to-image matching in the absence of any homogeneous texture. The EHD is basically intended for gray-image to gray-image matching but changing intensity and threshold parameters make it applicable for black and white queries in the sketch-based retrieval. The technique has been used for the SBIR in [108].
A given image is first divided into 16 sub-images \((4 \times 4)\), and local edge histograms are computed for each sub-image. To compute the edge histogram, each of the 16 sub-images is further subdivided into image-blocks as depicted in Figure 3.2. The size of each image block is proportional to the size of the original image and is assumed to be a multiple of two. The number of image blocks, independent of the original image size, is a constant \((\text{desired} \ num \ of \ blocks)\) and the block size is figured as follows:

\[
x = \sqrt{\frac{\text{image width} \times \text{image height}}{\text{desired} \ num \ of \ blocks}} \tag{3.17}
\]

\[
\text{block size} = \left\lfloor \frac{x}{2} \right\rfloor \times 2 \tag{3.18}
\]

where the \text{image width} and \text{image height} represent horizontal and vertical size of the image, respectively. The default value for the \text{desired} \ num \ of \ blocks is 1100.

Each image-block is then partitioned into four \((2 \times 2)\) blocks of pixels (see Figure 3.3). The pixel intensities for these four sub-blocks are computed by averaging the luminance values of the existing pixels. In the case of black and white sketch images, the luminance takes only the value of one or zero. All existing edges are grouped into five predefined classes: vertical, horizontal, \(45^\circ\) diagonal, \(135^\circ\) diagonal, and isotropic (i.e. non-directional).

For each image block, we determine whether there is at least one edge and which edge type is predominant. When an edge exists and the predominant edge type is de-
determined, the histogram value of the corresponding edge bin increases by one. The potential edge directions are determined for each image block using five $2 \times 2$ filter masks corresponding to the $2 \times 2$ subdivisions (sub-blocks) of the image-block. The edge types and the corresponding filter masks are shown in Figure 3.4. To explain this, let us represent the average gray level for four sub-blocks at $(i, j)$th image-block as $a_0(i, j), a_1(i, j), a_2(i, j),$ and $a_3(i, j)$, respectively and let $f_v(k), f_h(k), f_{45}(k), f_{135}(k),$ and $f_{iso}(k)$ denote the filter coefficients for vertical, horizontal, $45^\circ$ diagonal, $135^\circ$ diagonal, and isotropic edges, respectively, where $k = 0, 1, 2, 3$ determines the location of the sub-block. Using these representations, the respective edge magnitudes $m_v(i, j), m_h(i, j), m_{45}(i, j), m_{135}(i, j),$ and $m_{iso}(i, j)$ for the $(i, j)$th image-block are calculated as follows:

$$
m_v(i, j) = \sum_{k=0}^{3} a_k(i, j) \times f_v(k)$$
$$m_h(i, j) = \sum_{k=0}^{3} a_k(i, j) \times f_h(k)$$
$$m_{45}(i, j) = \sum_{k=0}^{3} a_k(i, j) \times f_{45}(k)$$
$$m_{135}(i, j) = \sum_{k=0}^{3} a_k(i, j) \times f_{135}(k)$$
$$m_{iso}(i, j) = \sum_{k=0}^{3} a_k(i, j) \times f_{iso}(k)$$

Now, if the maximum value among five edge strengths obtained from Eq. 3.19 is greater than a threshold ($T_{edge}$) as in:

$$\max \{m_v(i, j), m_h(i, j), m_{45}(i, j), m_{135}(i, j), m_{iso}(i, j)\} > T_{edge}$$

then the underlying block is designated to belong to the corresponding edge class. Otherwise, the image-block contains no significant edge and is considered as monotone block. The default value of ($T_{edge}$) for gray scale images is 11 and for binary sketches is set to 0.
Figure 3.4 Five different edges and corresponding filter coefficients for edge detection: (a) vertical, (b) horizontal, (c) 45° diagonal, (d) 135° diagonal, and (e) isotropic (non-directional) [63, 155]

In the MPEG-7 standard, the isotropic filter has been chosen heuristically. It is hard to find filter coefficients that are applicable for all types of non-directional edges. For this reason, an algorithm has been proposed in [155] which is not using the isotropic filter. Here, if the image-block does not belong to any of the monotone or one of the four directional edge blocks, it is classified as a isotropic block. The flowchart of this algorithm is shown in Figure 3.5.

The histogram for each sub-image represents the frequency of occurrence of the five classes of edges in the corresponding sub-image. As there are 16 sub-images and each has a 5-bin histogram, a total of $16 \times 5 = 80$ bins in the histogram is achieved. Won et al. [155] proposed efficient use of the EHD by extending the histogram to 150 bins. The extended histogram is obtained by grouping the image blocks into 13 clusters (4 vertical, 4 horizontal and 5 square clusters). Each cluster contains 4 sub-images as depicted in Figure 3.6-a. In addition to this semi-global histogram with $13 \times 5 = 65$ bins, another 5-bin global histogram is computed by combining all 16 local bins. This results in an overall 150 (80+65+5) bin histogram that is used for measuring the similarity between images (see Figure 3.6-b). As an advantage, the global and the semi-global histograms could be reconstructed directly from the local histogram at the matching time and need no extra space to store in memory.

For normalization, the number of edge occurrences for each bin is divided by the total
Figure 3.5 An edge classification flowchart without using the isotropic edge-filter for computing the MPEG-7's EHD descriptors [155]

number of image blocks in the corresponding sub-image. Finally, for the similarity matching, the distance $d(A, B)$ of two image histograms $A$ and $B$ (see Figure 3.6-b) is obtained by the following weighted Manhattan distance:

$$
d(A, B) = \sum_{i=0}^{70} |A_{Local}(i) - B_{Local}(i)| \\
+ 5 \times \sum_{i=0}^{4} |A_{Global}(i) - B_{Global}(i)| \\
+ \sum_{i=0}^{64} |A_{SemiGlobal}(i) - B_{SemiGlobal}(i)|
$$

The approach achieves good retrieval performances in some applications [63, 108, 155], however, it does not exhibit rotation invariance property since it not only ignores global rotation in an image by employing rectangular divisions but also considers only five predefined edge directions for the local blocks.
Angular Radial Transform (ART)

Angular radial transform (ART) descriptor is an efficient tool to retrieve object information [22]. An ART-based descriptor is adopted by the MPEG-7 standard [63] as a shape descriptor. The descriptor can describe complex objects (consisting of multiple disconnected regions) as well as simple objects. As the descriptor is based on the regional property, it shows robustness to segmentation noise, e.g. the salt and pepper noise.

By definition, the ART is a unitary transform defined on a unit disk that consists of the complete orthonormal sinusoidal basis functions in polar coordinates. From each image, a set of ART coefficients $\psi_{mn}$ of order $m$ and $n$ is extracted using the following formula:

$$\psi_{mn} = \int_0^{2\pi} \int_0^1 V_{mn}^* (\rho, \theta) f (\rho, \theta) \rho \, d\rho \, d\theta$$  \hspace{1cm} (3.22)

where $f (\rho, \theta)$ is an image intensity function in polar coordinates, and $V_{mn} (\rho, \theta)$ is the ART basis function of order $m$ and $n$ that is separable along the angular and radial
directions, that is:

\[ V_{mn}(\rho, \theta) = A_n(\theta)R_m(\rho) \]  

(3.23)

In order to achieve rotation invariance, an exponential function is used for the angular basis function:

\[ A_n(\theta) = \frac{1}{2\pi} e^{jm\theta} \]  

(3.24)

and the radial basis function is defined by a cosine function as:

\[ R_m(\rho) = \begin{cases} 
1 & m = 0 \\
2\cos(m\pi\rho) & m \neq 0 
\end{cases} \]  

(3.25)

It has been shown that the magnitudes of the ART are rotation invariant [192]. Recently, Hoynck and Ohm [193] have shown that shapes with partial occlusions can still be recognized using a modified version of this technique. The discrete ART coefficients of a gray scale image can be found easily using a look-up table [63]. The algorithm is summarized in the following section.

### 3.6.1 ART Extraction Algorithm

A four step algorithm for generating a 36-entry feature vector (Art[36]), using the ART technique is:

1. **Basis function generation:** to reduce complexity, \( V_{mn}(x, y) \) is computed directly in the Cartesian coordinates rather than converting it after computing \( V_{mn}(\rho, \theta) \) in the polar coordinates. For efficient implementation of the algorithm, the idea of using lookup table is employed. For this, we construct a set of complex basis functions of ART in two 4-dimensional arrays, \( BasisR[12][3][LutSize][LutSize] \) and \( BasisI [12][3] [LutSize] [LutSize] \) for real and imaginary parts, respectively. The LutSize is the third and forth sizes of the lookup table. The default value for the LutSize is 101 [63].

2. **Size normalization:** the center of mass is aligned to coincide with that of the lookup table. If the size of the image and that of the lookup table are different, the image is mapped onto the corresponding lookup table using linear interpolation.
3. **ART transformation**: the real and imaginary parts of the ART coefficients \((ArtR[12][3] \text{ and } ArtI[12][3])\) are computed, in a raster scan order, by summing up the multiplication of a pixel value in an image with the suitable entry in the corresponding lookup table.

4. **Area normalization**: the magnitude of the ART coefficients \((ArtM[n][m])\) are computed using root mean squaring of the corresponding real and imaginary parts. Finally, each element of the \(ArtM\) is divided by \(ArtM[0][0]\) to generate normalized coefficients.

The 2D vector \(ArtM\) is finally converted to a 1D vector \(Art\) using row order substitution. The MPEG-7 standard suggests the magnitude of the ART coefficients, excluding \(ArtM[0][0]\), to be quantized by 4 bits (i.e. a 16-level quantizer) using a predefined quantization table. The table is constructed by an exponential distribution model, that is \(\lambda e^{-\lambda x}\). The parameter \(\lambda = 18\) was determined experimentally.

The positive effect of the quantization stage is size reduction, however, the matching power of the method is decreased. As the quantization stage has not been used in this thesis, the details are not presented here. Interested readers can refer to [63] and [108] for more information.

The ART technique is inherently suitable for gray scale or black and white images. Color images are first converted to gray intensity images by eliminating the hue and saturation while retaining the luminance and then the ART algorithm is applied. The distance between two images \(A\) and \(B\) is measured by the Manhattan \((\ell_1)\) distance between the corresponding ART normalized coefficients, i.e.

\[
d(A, B) = \sum_{i=1}^{35} |Art_A(i) - Art_B(i)|
\]

### 3.7 Polar Fourier Descriptors (PFD)

The 2D Fourier transform in polar coordinates is employed for shape description in [125]. Its supremacy over the 1D Fourier descriptors, curvature scale space descriptors and Zernike moments has been shown in [83]. In this approach, an image
in the polar coordinates is treated as a normal rectangular region and the 2D Fourier transform is applied on this rectangular area. This polar Fourier transform (PFT) has a form that is similar to the traditional 2D discrete Fourier transform in the Cartesian coordinates. Consequently, for a given gray scale or black and white image \( f(x, y) \), the polar Fourier descriptors (PFD) are obtained as follows:

\[
pft(r, \phi) = \sum_{r} \sum_{\theta} f(r, \theta) e^{[2\pi i (r \cos \theta + \frac{\pi}{2} \phi)]}
\]

where \( r = [(x - x_c)^2 + (y - y_c)^2]^{0.5} \), \( 0 \leq r < R \) and \( \theta_i = i(2\pi / T), 0 \leq i < T \); \( 0 \leq \rho < R, 0 \leq \phi < T \). \((x_c, y_c)\) is the center of mass of the image, and \( R \) and \( T \) are the radial and angular resolutions. \( \rho \) and \( \phi \) are the radial and angular frequencies, respectively.

The above extracted Fourier coefficients are only translation invariant [125]. The following normalization process makes them scale and rotation invariant:

\[
PFD = \left\{ \left[ \frac{pft(0, 0)}{area} \right], \left[ \frac{pft(0, 1)}{pft(0, 0)} \right], \ldots, \left[ \frac{pft(0, n)}{pft(0, 0)} \right], \ldots, \left[ \frac{pft(m, 0)}{pft(0, 0)} \right], \ldots, \left[ \frac{pft(m, n)}{pft(0, 0)} \right] \right\}
\]

where parameter \( area \) is the area of the bounding circle in which the polar image resides. \( n \) is the maximum number of angular frequencies and \( m \) is the maximum number of radial frequencies selected. The following section summarizes an algorithm to implement the PFD method.

### 3.7.1 Polar Fourier Transform Extraction Algorithm

An algorithm is given in [83] for implementing the PFT technique which results in a translation, scale, and rotation invariant \( PFD \) feature vector. The algorithm is composed of two main stages:

- **Polar Fourier transform**: using two radial and angular resolutions, namely \( Rad \) and \( Ang \), real and imaginary parts of the polar Fourier transform of a given image \( f(x, y) \) are computed. The real part \( FR \) is the result of multiplying corresponding pixels in the image with the cosine function and the imaginary part \( FI \) resulting from multiplying the pixels with the sine function (based on Eq. 3.25).
Rotation and scale normalization: using root mean squaring for corresponding real and imaginary parts ($FR$ and $FI$), the $PFD$ coefficients are computed. Next, the $PFD$ coefficients are normalized applying Eq. 3.26.

3.8 Chapter Summary and Conclusion

In this chapter six different approaches including: (1) query by visual example, (2) histogram of edge directions, (3) invariant moments, (4) MPEG-7 edge histogram descriptor, (5) angular radial transform, and (6) polar Fourier transform, have been discussed in more detail. Relevant algorithms, diagrams, and pseudo codes which are used to implement the techniques were provided. These approaches can be used for sketch-based image retrieval (SBIR) when the query image is a rough black and white image containing one or several objects.

In the next chapter, we will introduce a new approach for the SBIR and compare the results with the above approaches.
Chapter 4

SBIR Using Image Abstraction and Angular Partitioning

4.1 Introduction

In this chapter we present a novel method for image similarity measure where a hand-drawn rough black and white sketch is compared with an existing database of full-color images (art works and photographs). The proposed approach provides SBIR in terms of the evaluation of non-precise, easy to input sketched information. The method can then facilitate user interfaces with options of either retrieving similar images in the database or ranking the quality of the sketch against a given standard, i.e. the original image model. Alternatively, the inherent pattern matching capability of the system can be utilized to allow for the detection of distortion in any given real time image sequences in vision driven applications. The proposed method can cope with images containing several complex objects in an inhomogeneous background. Two abstract images are obtained using strong edges of the model image and the morphologically thinned outline of the sketched image. The angular-spatial distribution of pixels in abstract images is then employed to extract new compact and effective features using the Fourier transform. The extracted features are rotation and scale invariant and robust against translation.

Experimental results from 7 different approaches confirm the efficacy of the proposed method in both the retrieval performance and the time required for feature extraction.
and search.

The outline of this chapter is as follows: in the next section, we present a brief background. The details of the proposed approach are discussed in Section 4.3. Section 4.4 exhibits evaluation criteria and detailed experimental results. It also provides a comparison of the proposed method with six other approaches presented in Chapter 3 based on retrieval performance, feature extraction time, and search speed. Finally, a chapter summary is given in Section 4.5.

4.2 Background

Representative content-based image retrieval systems include the QBIC [94], MARS [51], MetaSEEk [39], VisualSEEk [38], and the Blobworld [187] (see Chapter 2 for more details). The MPEG-7 standard defines descriptors derived from three main image content features: color, texture, and shape [21, 22]. The VisualSEEk algorithm proposed by Di Sciascio et al. [97] consider the spatial object layout as a significant content feature complementing color and texture attributes.

User interaction is one of the most important aspects of any multimedia system. A simple intuitive user interface makes the system more attractive and applicable. In sketch-based image retrieval (SBIR), where the query example is a rough and simple black and white hand-drawn sketch, color and texture lose their original ability to serve as content keys. Visual shape descriptors are useful in the SBIR when the model and the query image contain only one object in a plain background [105]. In multiple-object scenes, the object layout is a powerful tool, however object extraction and segmentation costs and rotation variance are the major drawbacks.

The edge pixel neighborhood information (EPNI) method utilizes a neighborhood structure of the edge pixels to make an extended feature vector [106]. The vector is used efficiently for measuring the similarity between sketched queries and arbitrary model images. The semantic power of the method is examined in [64]. Although the method is scale and translation invariant it does not exhibit rotation invariance property. Matusiak et al. [103] and Horace et al. [105] have reported using rough
and simple hand-drawn shape as input query for the SBIR. The approach in [103] is based on the curvature scale space, which is computationally expensive and has been shown to be less efficient than the Fourier descriptors and the Zernike moments techniques [83]. In [105] several dominant points are extracted for each contour using information derived from the convex hull and the contour curvature. Query-by-sketch is adequately investigated in [97] using the spatial relationships between shapes in an image. The approach introduces a way to represent shapes, their spatial arrangement, and color and texture attributes in the user sketch. Furthermore, in this approach, segmentation of the database image is required and the sketched queries include color and texture attributes.

Intuitive user interfaces, such as sketch-based ones, liberate the user from concerns about precision, orientation, scale, texture and color. In this chapter the focus, therefore, is on the problem of finding image features, invariant to rotation and scale changes, which can be used efficiently in sketch-based retrieval where images have several objects in an inhomogeneous background. Since object extraction and segmentation are not needed in this approach, therefore the database image (model) and the query image may consist of several complex objects. The input query has no color and texture attributes. We also eliminate any constraint regarding the shape of the objects and the existence of any background.

Six alternative approaches are implemented for comparison in this chapter. These are: (1) query by visual example, (2) histogram of edge directions, (3) invariant moments, (4) MPEG-7 edge histogram descriptor, (5) angular radial transform, and (6) polar Fourier transform. The details of these approaches are presented in Chapter 3.

### 4.3 Angular Partitioning of Abstract Image (APAI)

The main objective of the method presented here is to transform the image data into a new structure that supports measuring the similarity between a full-color image and a black and white hand-drawn sketch.

The edge map of an image carries the solid structure of the image independent of the
color attribute. Edges are also proven to be a fundamental primitive of an image for the preservation of both semantics and perceived attributes [194]. Furthermore, in the SBIR, edges form the most useful feature for matching purposes [103, 105, 106]. According to the assumption that sketched queries are more similar to the edge maps, which contain only the perceptive and vigorous edges, we initially obtain two abstract images through two different procedures (see Figure 4.1).

A. The first procedure is for image abstraction of the full-color images within the database (Figure 4.1-a). This procedure extracts the strong edges of the input image using a statistical approach (described next). The resulting image contains the most perceived outline of the image, which is comparable to the sketched query. This is fed into feature extraction stage.

B. The second procedure is for image abstraction of the sketched queries (Figure 4.1-b). According to the assumption that the input query is a black and white hand-drawn image with no color and texture attributes, a morphological thinning procedure [121] is employed for abstraction. This is based on the fact that in many cases there are thick lines in the sketched query created by the user which need to be thinned. This results in a desired outline of the query image containing only the most important and perceived pixels. The resulting thinned sketch is then used in the feature extraction step.
4.3.1 Statistical Image Abstraction

The full-color model image is initially converted to a gray intensity image $I$ by eliminating the hue and saturation while retaining the luminance. There are many edge extraction algorithms in the literature and the Canny algorithm [102] is the most useful one due to its computation and extraction power [195, 196]. In the proposed approach, the edges are extracted using the Canny operator with $\sigma = 1$ and Gaussian mask of size = 9 using the following procedure for depicting the most perceived edges.

The values of high and low thresholds for the magnitude of the potential edge points are automatically computed in such a way that only the strong edges are retained. This improves the general resemblance of the resulting edge map and the hand-drawn query. In order to depict strong edges, let $G$ be the Gaussian 1D filter and let $g$ be the derivative of the Gaussian used in the Canny edge operator. Then

$$H(k) = \sum_i G(i) g(k + 1 - i)$$  \hspace{1cm} (4.1)

is the 1D convolution of the Gaussian and its derivative.

$$X(u, v) = \left[ \sum_{j=1}^{V} I'(u, j) H(v - j) \right]'$$  \hspace{1cm} (4.2)

and

$$Y(u, v) = \sum_{i=1}^{U} I(i, v) H(u - i)$$  \hspace{1cm} (4.3)

where $u = 1, 2, 3, \ldots U$ and $v = 1, 2, 3, \ldots V$ are the vertical and horizontal edge maps, respectively. $U$ is the number of rows and $V$ is the number of columns in the image $I$. The’ notation indicates matrix transpose. The magnitude of the edge points is then obtained as:

$$\Gamma(u, v) = \sqrt{X(u, v)^2 + Y(u, v)^2}$$  \hspace{1cm} (4.4)

The $\Gamma(u, v)$ values are in $[0,1]$. For efficient selection of the high and low thresholds, first, a 64-bin histogram $h$ of the $\Gamma(u, v)$ values is generated. Next, we create the corresponding cumulative histogram $C$ and find the minimum index $\iota$ in $C$ that is greater than $\alpha \cdot U \cdot V$, where $\alpha$ denotes the percentage of non-edge points in the image ($\alpha = 0.7$ is an adequate choice for many images). Figure 4.2 depicts the $h$ and
Figure 4.2 Edge points distribution for (a) three different images, (b) corresponding $h$ histograms, and (c) corresponding cumulative $C$ histograms

$C$ histograms for three different images. As it can be seen, the brighter the image, the wider the $h$ histogram.

To retain strong edges of the image, $\beta \cdot \epsilon$ is selected as the high threshold value and $0.4\beta \cdot \epsilon$ is used for the low threshold value in the Canny edge operator. $\beta$ is a parameter that controls the degree of the strength of the edge points. Higher $\beta$’s lead to a lower number of edge points but more perceptive ones (see Figure 4.3). Consequently, the gray image $I$ is converted to an edge image $P$ using the Canny edge operator exploiting the above automatically extracted thresholds.

As mentioned above, for the query images, the procedure of black and white mor-
Figure 4.3 The effect of parameter $\beta$ on edge maps.

Phological thinning [121] is applied to extract a thinned version of the sketched image. This image, denoted as $Q$, shows an outline of the query and contains the main structure of the user request. It contains the spatial distribution of pixels similar to the strong edge map of the model image $P$. Isolated pixels are removed in both $P$ and $Q$ images to reduced noise effects. Figure 4.4 shows a sketched image and the corresponding morphologically thinned and isolated pixels removed version.

To achieve scale and translation invariance in the space of query and model images, the following normalization procedure is subsequently applied. First, the bounding boxes of the images $P$ and $Q$ are obtained. The area in the bounding box of the comparable images are then normalized to $J \times J$ pixels using a nearest neighbor interpolation. Working in the bounding box of images lets the region of interest to be located anywhere in the scanned paper yielding translation invariance. The proposed normalization of the images $P$ and $Q$ then ensures scale invariance since different size images are converted to predefined-size images. The resulting image is called abstract image $\Omega$. 
4.3.2 Angular Partitioning (AP)

We define angular partitions (slices) in the surrounding circle of the abstract image $\Omega$ (see Figure 4.5). The angle between adjacent slices is $\varphi = 2\pi/K$ where $K$ is the number of angular partitions in the abstract image. $K$ can be adjusted to achieve hierarchical coarse to fine representations. Any rotation of a given image, with respect to its center, moves a pixel at slice $S_i$ to a new position at slice $S_j$ where $j = (i + \lambda) \mod K$, for $i, \lambda = 0, 1, 2, \ldots K - 1$.

The number of edge points in each slice of $\Omega$ is chosen to represent the slice feature. The scale and translation invariant image feature is then $\{f(i)\}$ where

$$f(i) = \sum_{\theta = \frac{i\varphi}{K}}^{\frac{(i+1)\varphi}{K}} \sum_{\rho = 0}^{R} \Omega(\rho, \theta)$$  \hspace{1cm} (4.5)

for $i = 0, 1, 2 \ldots K - 1$. $R$ is the radius of the surrounding circle of the abstract image. Although different lighting configurations yields to different numbers of edge points in the slices, the abstraction process (Section 4.3.1) considerably reduces such effect.

The feature extracted above will be circularly shifted when the image $\Omega$ is rotated.
SBIR Using Image Abstraction and Angular Partitioning

Figure 4.5 Angular partitioning splits the abstract image into $K$ successive slices.

For $\tau = l2\pi/K$ radians ($l = 0, 1, 2 \ldots$). To show this, let $\Omega_{\tau}$ denote the abstract image $\Omega$ after rotation by $\tau$ radians in counterclockwise direction:

$$\Omega_{\tau}(\rho, \theta) = \Omega(\rho, \theta - \tau) \quad (4.6)$$

Then,

$$f_{\tau}(i) = \sum_{\theta = i2\pi/K}^{(i+1)2\pi/K} \sum_{\rho=0}^{R} \Omega_{\tau}(\rho, \theta) \quad (4.7)$$

are the image feature elements of the $\Omega_{\tau}$ for the same $i$’s. We can express $f_{\tau}$ as:

$$f_{\tau}(i) = \sum_{\theta = i2\pi/K}^{(i+1)2\pi/K} \sum_{\rho=0}^{R} \Omega(\rho, \theta - \tau)$$

$$= \sum_{\theta = i2\pi/K}^{(i+1)2\pi/K} \sum_{\rho=0}^{R} \Omega(\rho, \theta) \quad (4.8)$$

$$= f(i - l)$$

where $i - l$ is a modulo $M$ subtraction. $f_{\tau}(i) = f(i - l)$ indicates that there is a circular shift in the image feature $\{f_{\tau}(i)\}$ corresponding to the image feature $\{f(i)\}$, representing $\Omega_{\tau}$ and $\Omega$ respectively.
Using the 1D discrete Fourier transform of \( f(i) \) and \( f_r(i) \) we obtain:

\[
F(u) = \frac{1}{K} \sum_{i=0}^{K-1} f(i)e^{-j2\pi ui/K}
\]

\[
F_r(u) = \frac{1}{K} \sum_{i=0}^{K-1} f_r(i)e^{-j2\pi ui/K}
\]

\[
= \frac{1}{K} \sum_{i=l}^{K-1-l} f(i - l)e^{-j2\pi ul/K}
\]

\[
= \frac{1}{K} \sum_{i=-l}^{K-1-l} f(i)e^{-j2\pi u(i+l)/K}
\]

\[
= e^{-j2\pi ul/K} F(u)
\]

Based on the property \( |F(u)| = |F_r(u)| \), the scale, translation, and rotation invariant image feature is chosen as \( \Psi = \{|F(u)|\} \) for \( u = 0, 1, 2 \ldots K - 1 \). The extracted features are robust against translation because of the aforementioned normalization process. Choosing a medium-size slice (e.g. \( \varphi = 30^\circ \)) makes the extracted features more vigorous against local variations which are common in hand-drawn sketches. This is based on the fact that the number of pixels in such slices varies slowly with local translations. The features are rotation invariant due to the Fourier transform applied.

Figure 4.6 summarizes the procedure discussed so far and Figure 4.7 shows an image example, its 90\(^{\circ}\) rotated version, corresponding abstract images before size normalization, abstract images superimposed with the APAI slices, and extracted \( \{f(i)\} \) and \( \Psi = \{|F(u)|\} \) features. Experimental results (Section 4.4) confirm the robustness and efficiency of the proposed method.

![Figure 4.6](Image.png)

**Figure 4.6** Bounding box of the abstract image \( P \) or \( Q \) is size normalized and supplied to the angular partitioning process. The final result is an associated feature vector \( \Psi \) used for similarity measurement.
Figure 4.7 (a) An image example, (f) the 90° rotated version, (b,g) corresponding abstract images before size normalization, (c,h) abstract images superimposed with angular partitions, (d,i) number of pixels in different slices, and (e,j) invariant features extracted using the Fourier transform, respectively.

4.4 Experimental Results and Efficacy Evaluation

In this section, first, the similarity measurement and criteria employed for retrieval performance assessment are discussed. Then, the effect of different parameters of the APAI method is investigated and finally comparative results, showing the degree of rotation and scale invariant properties are presented.

4.4.1 Similarity Measurement and Retrieval Performance Criteria

Similarity between images is measured by the $\ell_1$ (Manhattan) distance between the two corresponding feature vectors (different similarity measures have been discussed in Section 2.6). Suppose $\Psi_P$ and $\Psi_Q$ represent two different images $P$ and $Q$ respectively. The similarity between image $P$ and $Q$ is the inverse of their Manhattan
distance calculated as:

\[ d(P, Q) = \sum_u |\Psi_P(u) - \Psi_Q(u)| \]  (4.10)

“Recall” and “Precision” are well known retrieval performance measures. They are basically “hit-and-miss” counters. In other words, the retrieval performance is based on the number of retrieved images which have similarity measures that are greater than a given threshold. For more specific comparisons, however, we also need the rank information among the retrieved images. The Average Normalized Modified Retrieval Rank (ANMRR), which was developed during the MPEG-7 standardization activity, is a measure that exploits not only the Recall and Precision information but also the rank information among the retrieved images. It is described in details in Section 2.7.2 of this thesis. We reiterate that the ANMRR is always in the range of [0, 1]. Based on the definition of the ANMRR, the smaller the ANMRR, the better the retrieval performance.

To answer the question of “What is the behavior of the retrieval algorithm if the ground truth of a query image is empty?”, the receiver operating characteristic (ROC) curve [18] is exploited. ROC is an interesting analysis tool in two-class problems and is employed here to evaluate the ability of the system when no similar images to a given query exist. To this end, queries with no similar images in the database, in addition to queries with some similar images in the database, are given to the system and the response of the system which could be either (a) “there exist some similar images” or (b) “there exists no similar image” is logged. To assess this interesting and novel capability of the retrieval process we conducted several tests with different thresholds and results were obtained. For different threshold values, the true positive ratio (sensitivity) and the false positive ratio (1-specificity) are computed [197] and the corresponding ROC curve is depicted (see Section 4.4.2 for the details).

One of the difficulties involved in achieving a ROC curve for the SBIR is the diversity in the range of the similarity measures. To select an effective threshold for all queries, their distances to the database images should be in a tantamount interval. In order to overcome the problem, distance values are normalized to be within the same range of [0,1]. More precisely, let \(d(P, Q)\) denotes the Manhattan distance between \(Q\) and
$P$ images. Readers are requested to refer to Section 2.6 of this thesis for a review of similarity measures. The normalization is done as follows [74]:

$$d_n(P, Q) = \frac{d(P, Q) - \min \text{dist}}{\max \text{dist} - \min \text{dist}}$$

(4.11)

where $d_n(P, Q)$ is the normalized distance. $\min \text{dist}$ and $\max \text{dist}$ are the minimum and the maximum distance values of the query image $Q$ to the database images according to the corresponding feature vectors. Consequently, the threshold for computing sensitivity and specificity varies in the interval of zero to one for all queries.

### 4.4.1.1 Extensions to ROC Curve

Receiver operation characteristics (ROC) analysis is an established method of measuring diagnostic performance in medical imaging studies. However, it has been used in other disciplines as well. Traditionally, artificial neural networks (ANN) have been applied as a classifier to find one best detection rate. Recently researchers have begun to report ROC curve results for ANN classifiers. The current standard method of generating ROC curves for an ANN is to vary the output node threshold for classification. Woods et al. [198] have proposed a different technique for generating ROC curve for a two-class ANN classifier. Here, ROC points are generated from a single trained ANN by systemically varying some underlying parameters in the network. This method generates a better ROC curve in terms of both the area under the curve (AUC) and the distribution of operating points across the true positive (TP) and false positive (FP) ranges.

Chakraborty et al. [199] has developed a new method for measuring small differences in diagnostic task performance between two imaging modalities. In this method, called differential receiver operation characteristic (DROC), the observer is shown a pair of images of the same patient, one from each modality $A$ and $B$. The patient can be normal or abnormal but this information is not known to the observer. The observer select the image which is preferred for the specific diagnostic task and he/she assigns a rating score to each observed image. Analysis of this experiment yields the area under the DROC curve ($A_{d}$). If parameter $A_{d}$ is greater than 0.5, then modality $B$ is superior to modality $A$ and conversely, if it is less than 0.5, then $A$ is superior to $B$. Based on experimental result, the DROC method was found to track the con-
ventional ROC method and to yield far greater sensitivity. Consequently, the DROC method has been shown to have potential ability for detecting suitable differences in image quality.

Moreover, clinical tasks involving lesion localization do not fit the ROC paradigm. Alternative approaches namely free-response ROC (FROC) and localization ROC (LROC), have their own limitations [200]. They neglect intra-image correlations, with consequent questionable statistical validity. Furthermore, the image scoring criterion is arbitrary and affects the results, and they do not cater for the sub-tasks of detection and localization. A new FROC model that deals with these issues has been proposed in [200]. This model is used to generate simulated FROC data that illustrate the aforementioned limitations. Measures of detection and localization performance are also proposed on [200], which will allow FROC performance to be interpreted quantitatively, and observer performance experiments to be conducted with greater statistical power.

### 4.4.2 The Effect of Parameter Variations

In order to evaluate retrieval effectiveness of the proposed method and to test the effects of parameter variations on the retrieval performance, a database of model and query images was created and several experiments were conducted. The database is a collection of different model and query images called ART-PHOTO BANK. It includes 4000 full-color heterogeneous images of various sizes in the model part (500 in groups of 8) and 400 sketches in its query part (100 in groups of 4). Images in the model part are a true-balanced combination of 250 art works, gained from the World Art Kiosk at California State University, and 250 real natural photographs from set S3 of the MPEG-7 database. Each group contains 8 similar images created by rotation in steps of $45^\circ$. This results in a variety of scaled and rotated samples. Images in the query part are hand-drawn black and white sketches similar to 100 arbitrary candidates from the model part and their rotated versions ($90^\circ, 180^\circ$ and $270^\circ$). This is to simulate different vertical and horizontal directions when posing a sketched query to the retrieval system. Sketches were drawn by different users and then scanned with 200 dpi resolution using a Hewlett-Packard scanner model scanjet.
The performance of the proposed method for various internal parameters is evaluated using the ANMRR measure. Different levels of abstraction ($\beta$ parameter) and different numbers of angular partitions ($\varphi$ parameter) for several size normalization parameter $J$ were tested. This is to determine which set is the most suitable for SBIR. In our experiments we have chosen $NG(q) = 8$ for all $q$’s, $L = 16$, and $Q_{total}=400$.

Figures 4.8-a, b and c depict the resulting ANMRR for $\beta = 1$, 2, and 3 respectively. In each figure, four different values for the size normalization parameter $J$ with eight different numbers of angular partitions (corresponding to $\varphi = 2^\circ$, $5^\circ$, $10^\circ$, $15^\circ$, $20^\circ$, $30^\circ$, $45^\circ$, and $60^\circ$) were examined. Figure 4.8-c, which relates to $\beta = 3$, exhibits the best results. Generally, $\beta = 3$ results in smaller ANMRR values than $\beta = 1$ and $\beta = 2$ depicted in Figures 4.8-a and b, respectively. Higher $\beta$’s were also examined, but the results were worse than those in Figure 4.8-b. The optimum performance for $\beta = 3$ is due to the following reasons: (a) smaller $\beta$’s generate significant noise in the edge images resulting in considerable variation among the abstract images (see Figure 4.3), and (b) higher $\beta$’s generate edge images whereas essential information is lost, rendering the comparison with query images ineffective.

Normalizing the image size to $257\times257$ yields the best results. This is based on the fact that images in the database having an average size closer to $257\times257$ than the other sizes ($65\times65$, $129\times129$, and $513\times513$). Consequently, normalizing with this size keeps more information in the abstract images and reduces the adverse effect of normalization.

With smaller $\varphi$’s (higher numbers of slices), the proposed method exhibits better performance with the exception of $\varphi = 2^\circ$ (see Figure 4.8-3). The reason is that the system can capture more details with a medium to small number of slices, but with very small slices, the method loses the robustness against small translations which always exist between abstract images and the thinned sketched queries and consequently the overall performance degrades. Therefore, $\beta = 3$, $J = 257$, and $\varphi = 5^\circ$ are chosen as the system’s internal characteristics.
Figure 4.8 The ANMRR (Average Normalized Modified Retrieval Rank) measure of the APAI method with (a) $\beta = 1$, (b) $\beta = 2$, and (c) $\beta = 3$ for a varying number of slices and four different normalized sizes.

Figure 4.9 shows the ROC curve of the system with the aforementioned parameters. The false positive ratio (FPR) i.e. 1-specificity, is obtained, posing 50 queries which have some similar images in the database, using

$$FPR = \frac{C}{C + D}$$

(4.12)

where $C$ is the number of cases where the system says “there exist no similar image”, and $D$ is the number of cases where the system says “there exist some similar images”. Similarly, the true positive ratio (TPR) i.e. sensitivity, is obtained posing
another 50 queries which have no similar images in the database, using

\[ TPR = \frac{A}{A + B} \]  

(4.13)

where \( A \) is the number of cases where the system says “there exist no similar image”, and \( B \) is the number of cases where the system says “there exist some similar images”. Threshold values are set to 1 (lower-left corner) downward to zero (upper-right corner) with the step of 0.1 on the normalized distance \( d_n(P, Q) \). As it can be seen, the sensitivity of the proposed method is higher than its specificity. This comes from the fact that rejecting queries with no similar images is easier than finding similar images to a given sketched query.

4.4.3 Comparative Results

To compare the retrieval performance of the proposed method with some other approaches within the literature, we implemented the following algorithms: (a) query by visual example (QVE) as used in the QBIC system [9, 94], (b) histogram of edge directions (HED) introduced by Jain and Vailaya [34], (c) Zernike moment invariants (ZMI) [124, 156], (d) MPEG-7 edge histogram descriptor (EHD) [63, 155], (e) angular radial transform (ART) [22, 63], and (f) polar Fourier descriptors (PFD) proposed by Zhang and Lu [83, 125]. The aim is to show the degree of rotation and scale invariance for these approaches. There are some other approaches for the SBIR such as [97], introduced by Di Sciascio et al., but since they need image segmentation at
the preprocessing stage and the queries contain color and texture attributes, they are not applicable to this study.

It should also be noted that the initial input to the ZMI, ART, and PFD methods is the thinned version of the sketched query and the strong edge map of the model image. This is to eliminate the adverse effect of color and texture diversity of the images on these methods. Moreover, in order to create a uniform assessment situation for all methods, we ignore the quantization stage in the EHD and the ART methods. This will remove the retrieval performance disadvantage of quantization for the EHD and the ART methods.

All methods were tested using the ART-PHOTO BANK database. We applied \( k = 1 \) in the HED method, resulting in a 70-entry feature vector. In the ART method, a 35-entry feature vector is achieved using \( m = 3 \) and \( n = 12 \) as recommended in [63]. For the ZMI method, we used 36 moments as suggested in [156], resulting in a 36-entry feature vector. For the EHD method, \textit{desired num of blocks} was set to 1100 and \( Th_{edge} \) was set to 11 (the default values) for the model images, and \( Th_{edge} \) was set to zero for the queries since they are binary images. A 150-bin histogram is obtained employing local, semi-global and global histograms. Furthermore, we followed the algorithm given in [83] to obtain a 60-bin feature vector in the PFD approach. The proposed APAI method (Section 4.3) resulted in a 72-bin feature vector using \( \beta = 3 \), \( J = 257 \), and \( \varphi = 5^\circ \) as the internal values based on the discussion in the previous subsection.

The \( \ell_1 \) (Manhattan) distance was used for measuring the similarity between all image features, while for the HED method a weighting factor of 5 for the global bins, as recommended in [155], was applied. The \( \ell_2 \) (Euclidian) distance was exploited for measuring the similarity between the PFD features [83], and a global correlation factor was employed for measuring the similarity between images in the QVE [9] method.

Figure 4.13(a) shows the results expressed by the ANMRR measure. Again, we used \( NG(q) = 8 \), \( L = 16 \), and \( Q_{total} = 400 \) for all methods. The results confirm that the proposed method yields the best retrieval performance (the lowest ANMRR i.e.
0.3070). The ART, PFD, and ZMI methods also show reasonable retrieval performance i.e. 0.3589, 0.3908 and 0.3984, respectively. These methods perform better under the rotation test because their basis functions are designed specifically to be rotation invariant. The basis function used in the ART method can efficiently capture the similarity among images as it splits the image (in the transform domain) into radial and angular directions.

The retrieval performance of the HED method is in a moderate level (0.4801), because the bins in the edge direction histogram are shifted during image rotation, therefore, rotation invariance is hardly achieved in this method. Although a histogram smoothing is applied to overcome the problem [34], a better option might be a shift invariant operator such as absolute value of the Fourier transform to improve the rotation invariance property for this method.

The retrieval performances of the MPEG-7’s EHD method (0.5816) and the QVE (0.6713) method confirm their lack of rotation invariance property. The EHD method considers only five predefined edge directions in local blocks which is inadequate to achieve rotation invariance. The QVE method compares neighboring pixels in the corresponding image regions and ignores global translation and rotation phenomena. Figures 4.10, 4.11, and 4.12 show three sets of retrieved images using the APAI method. Note that here the ANMRR picks up the 16 \((2 \times GTM)\) top similar images. Therefore, as each input query has 8 \((GTM)\) similar images in the database, in the best case there are only 8 similar images in the retrieval list. For Figure 4.10, the NMRR (normalized modified retrieval rank) is 0.2400, and for Figures 4.11 and 4.12, the NMRR is 0.3600 and 0.4200, respectively.

The scale invariance property of the methods, in particular, were also tested. All methods recruit size normalization, bounding box limitation, and/or center of mass alignment to achieve the scale and translation invariant properties. They exploit different normalized sizes, for example, the QVE method normalizes images to \(64 \times 64\), the ART method to \(101 \times 101\), and the APAI method to \(257 \times 257\). In this experiment, we applied the original 100 queries on a smaller database which contains 2000 full-color images. This database includes the aforementioned 500 original images supplemented by scaled versions created by three scale factors of 0.5, 1.5 and 2.
Once again, we obtained the ANMRR of the seven different approaches for these 100 queries with $NG(q) = 4$ for all $q$'s, $L = 8$, and $Q_{total} = 100$. Figure 4.13(b) shows the results. As can be seen, the ANMRR measure of the proposed method, and the QVE, MPEG-7’s EHD and HED methods are acceptable (less than 0.5). The QVE method shows the best retrieval performance in this test which confirms that the method well tolerates size variation and local translation. However, as already explored, the method is not rotation invariant. It also needs more computation power than the other methods since it calculates a global correlation factor between images, which significantly slows down the matching process. The resulting ANMRR of the other methods, i.e. ART, ZMI and PFD, are high (more then 0.5). The reason seems to be that they concentrate on the rotation invariance property but are acutely sensitive to image size.

It is worthwhile to mention that all methods, except the QVE method, generate a
feature vector for each image. Therefore, they can support indexing easily. The length of the feature vector (feature space dimension) for the selected methods are not exactly the same, i.e. 72, 36, 60, 36, 70, and 150 are the vector length for the APAI, ART, PDF, ZMI, EHD, and MPEG-7 EHD methods, respectively. On the other hand, the QVE approach, which uses a correlation scheme for measuring the similarity between images, cannot be used to generate indexes for the database.

Finally, the feature extraction time (FET) and the search time (SET), which are important factors for both the database population phase (off-line) and the query processing phase (on-line), were computed for the above methods. The average values of the FET and SET parameters, using the ART-PHOTO BANK database, are obtained using a Pentium-III, 1000 MHz machine and are shown in Figure 4.14. The FET parameters for the QVE, HED, APAI, and EHD methods are higher than for the ART, PDF, and ZMI methods. The reason being that the edge extraction, which is a time consuming procedure, is only employed in the first group and not in the second
The SET parameter, which deals directly with the comparison of feature vectors, shows proportionality to the corresponding vector’s length with the exception of the PFD and QVE methods. The PFD method is the only one that uses the $\ell_2$ distance for measuring the similarity between features, which needs more computation time than the $\ell_1$ distance. The QVE method neglects feature vector advantages and calculates the global correlation between abstract images during the on-line phase. Therefore, its SET parameter is the highest (187 Sec.). Note the discontinuity in the time axis, which shows that the search time for the QVE method is more than 10 times longer than for the other methods.

### 4.5 Chapter Summary and Conclusion

The approach presented in this chapter (the APAI method) enables measuring the similarity between a full-color model image and a simple black and white sketched
query. The images are arbitrary and may contain several complex objects in an inhomogeneous background. The approach deals directly with the entire image and needs no computationally intensive image segmentation and object extraction. Abstract images are defined based on statistical threshold finding to retain strong edges of the model image and morphological thinning of the query image. Angular partitioning of the abstract image, using the Fourier transform, is exploited to extract features that are scale and rotation invariant and robust against translation. Experimental results, using the APAI approach and the ART-PHOTO BANK as the test bed, show significant improvements in the ANMRR measure and rotation tolerance over six other well known approaches within the literature. Individual tests on scale invariance were also conducted and showed that the proposed method has better retrieval performance than five other approaches. While the QVE method exhibits the best tolerance for scale variations, its computational time is the highest (i.e. more than ten times longer). The proposed method depicts good retrieval performance in both rotation and scale tests while it relies on reasonable feature extraction and search times.
The proposed intuitive user interface allows the user to interact with the system through a rough hand-drawn sketch without concern for precision, scale, orientation or color. The pattern matching capability of the system facilitates search and retrieval as well as other possible applications such as distortion measurement, drawing skill training and new generation man-machine interfaces.

Some improvements on this approach are presented in the next chapter.
Chapter 5

Advanced Features Using Angular Radial Partitioning

5.1 Introduction

This chapter presents a novel approach for image representation based on geometric distribution of edge pixels. It improves retrieval power of the angular partitioning method (Chapter 4) with a more precise partitioning scheme. In particular, edge image comparison is investigated. For efficient description of an arbitrary edge image, the image is divided into $M \times N$ angular radial partitions and local features are extracted for these partitions. The entire image is then described as a set of spatially distributed invariant feature descriptors using magnitude of the Fourier transform. The approach is scale and rotation invariant and tolerates small translations and erosions. The extracted features are characterized by their compactness, fast extraction and matching times. They exhibit improvements in retrieval performance using the ANMRR measure. Experimental results, using an image database initiated from a movie, confirm the supremacy of the proposed method.

The rest of the chapter is organized as follows: a concise background is presented in the next section. Section 5.3 describes the proposed approach in details. Next, Section 5.4 presents experimental results on the proposed method and compares several approaches. Finally, Section 5.5 summarizes the chapter.
5.2 Background

Rotation and translation invariant properties are crucial in most recognition tasks and should be considered in the features chosen for image retrieval. The invariant methods can be categorized in the following two main approaches:

- **Image alignment**, i.e. a transformation is applied to the image so that the object in the image is placed in a predefined standard position. Furthermore, the approach relies on the extraction of geometric primitives like extrema of the boundary curvature, bitangents, or inflection points. Segmentation of the object is necessary and the approach is not trivial especially when there exists more than one object in the image [201].

- **Invariant features**, i.e. using invariant image characteristics which remain unchanged if the object rotates or moves. Although this approach has attracted considerable interest [202], it is still based on geometric primitives. This means extraction of primitives such as extrema of the boundary curvature and/or center of mass (COM) is needed. Therefore it suffers from the same shortcomings of the image alignment approach (e.g. object segmentation is needed).

It is desirable to avoid segmentation preprocessing and to start directly with the image pixels. One possibility is to employ invariant moments such as the regular moments or the Zernike moments [191]. However, for eliminating image translations it is necessary to identify at least one matching point between images. The most common choice is the center of mass (COM) for calculating central moments. Thus moments can be considered as a hybrid of alignment and invariants approaches.

The edge points hold considerable information about the image structure especially in the absence of color and/or texture information or in images where they are not the discriminating factors. Furthermore, there are applications such as sketch-based image retrieval where only the edge map of the database image is comparable to the sketched query [103, 105, 106]. A face feature representation, called Line Edge Map (LEM), is proposed in [203] to integrate the structural information with spatial
information of a face image by grouping pixels of face edge maps into line segments. The edge pixel neighborhood information (EPNI) method [106] exploits neighboring arrangement of the edge pixels to make scale and translation invariant features. The EPNI method suffers from lack of rotation invariance property.

Jia and Wang [204] have proposed a structural feature description based on geometric partitioning of edge images. They employ a sequential sampling model to partition edge pixels into circular blocks. The approach is computationally intensive. It also requires a predefined value (the number of edge pixels for each block) that need to be found empirically. The extracted feature vectors for different images have different sizes and the matching procedure is nontrivial.

5.3 Angular Radial Partitioning (ARP)

The main objective of angular radial partitioning (ARP) is to transform the image data into a new structure that supports measurement of the similarity between images in an effective, segmentation-free, and efficient manner with emphasis on capturing scale and rotation invariance properties.

Edge points carry very useful information in image processing and computer vision tasks. Moreover, sketched queries are considerably similar to the edge maps. Therefore, in the ARP algorithm the following initial steps are applied in the same manner as described in Chapter 4:

- The images in the database are converted to gray intensity by eliminating the hue and saturation while retaining the luminance.
- An edge extraction operator, e.g. Canny edge operator is applied on this gray scale image to obtain an edge image.
- Morphological thinning and noise reduction are applied for the sketched queries.
- In order to achieve scale invariance property, the resulting images are then normalized to $W \times W$ pixels.
Figure 5.1 Angular Radial Partitioning of an image to $N$ angular and $M$ radial sectors where $k = 0, 1, 2 \ldots M - 1$ and $i = 0, 1, 2 \ldots N - 1$

The resulting normalized image is called $I$ and used for feature extraction. In the following, we consider pixels $I(\rho, \theta)$ to be either equal to "1" for the foreground pixels, or "0" for the background pixels.

The algorithm uses the surrounding circle of $I$ for partitioning it to $M \times N$ sectors, where $M$ is the number of radial partitions and $N$ is the number of angular partitions. The angle between adjacent angular partitions is $\theta = 2\pi/N$ and the radius of successive concentric circles is $\rho = R/M$ where $R$ is the radius of the surrounding circle of the image (see Figure 5.1).

The number of edge points in each sector of $I$ is chosen to represent the sector feature. The scale invariant image feature is then \( \{f(k, i)\} \), where

$$f(k, i) = \sum_{\rho = \frac{k+1}{M}}^{\frac{k+1}{M}} \sum_{\theta = \frac{i+1}{N}}^{\frac{i+1}{N}} I(\rho, \theta)$$

(5.1)

for $k = 0, 1, 2 \ldots M - 1$ and $i = 0, 1, 2 \ldots N - 1$.

The feature extracted above will be circularly shifted when the image $I$ is rotated $\tau = l2\pi/N$ radian ($l = 0, 1, 2 \ldots$). To show this, let $I_{\tau}$ denote the image $I$ after rotation by $\tau$ radians in counterclockwise direction:

$$I_{\tau}(\rho, \theta) = I(\rho, \theta - \tau)$$

(5.2)

Then, based on similar discussion in Chapter 4, it is obvious that:

$$f_{\tau}(k, i) = f(k, i - l)$$

(5.3)
where $f_r(k, i)$ are the image feature elements for $I_r$ for the same $k$ and $i$ while $i - l$ is a modulo $M$ subtraction. It means that there is a circular shift, for individual $k$'s, in the image feature $\{f_r(k, i)\}$ representing $I_r$, in comparison with the image feature $\{f(k, i)\}$ representing $I$.

Using 1D discrete Fourier transform of $f(k, i)$ and $f_r(k, i)$ for each $k$ we obtain

$$ F(k, u) = \frac{1}{N} \sum_{i=0}^{N-1} f(k, i) e^{-j2\pi ui/N} $$

$$ F_r(k, u) = \frac{1}{N} \sum_{i=0}^{N-1} f_r(k, i) e^{-j2\pi ui/N} $$

$$ = \frac{1}{N} \sum_{i=0}^{N-1} f(k, i - l) e^{-j2\pi ui/N} $$

$$ = \frac{1}{N} \sum_{i=-l}^{-l-1} f(k, i) e^{-j2\pi x(i+l)/N} $$

$$ = e^{-j2\pi ul/N} F(k, u) $$

Similarly, because of the property $|F(k, u)| = |F_r(k, u)|$, the scale and rotation invariant image features are chosen as $\{|F(k, u)|\}$ for $k = 0, 1, 2 \ldots M - 1$ and $u = 0, 1, 2 \ldots N - 1$. Similarity between images is measured using the $\ell_1$ (Manhattan) distance between corresponding extracted features.

Choosing a medium-size sector (e.g. $M = 3$ and $N = 8$) makes the invariant image features extracted above to be robust to other small variations as well (i.e. translation, erosion and occlusion). This is due to the fact that the number of edge pixels in such sectors varies slowly with such variations.

Figure 5.2 shows an image example, its $90^\circ$ rotated version, corresponding edge maps superimposed with ARP sectors, and extracted $f(k, i)$ and $F(k, u)$ features.

Experimental results (Section 5.4) confirm retrieval performance improvement of the ARP method in comparison to the angular partitioning (AP) scheme.
Figure 5.2 (a) An image example, (b) its $90^\circ$ rotated version, (c,d) corresponding edge images superimposed with angular-radial partitions, (e,f) number of pixels in the individual partitions, and (g,h) invariant features extracted using the Fourier transform.
5.4 Experimental Results, Comparison and Discussion

In order to evaluate the retrieval effectiveness and efficiency of the proposed method two sets of experiments were conducted as explained below. First, a comparison between the ARP and AP (Chapter 4) methods for the SBIR is provided. We have used the same database of art works and photographs and sketched queries (ART-PHOTO-BANK) as described in Section 4.4.2. Second, as a new application of the ARP technique, edge image matching is investigated with more details. Here, another database initiated from a movie has been used for the experiments. All necessary justifications for this application are provided. The results are compared with the AP and four other approaches.

The ANMRR criterion (described in 2.7.2) is used as the retrieval performance measure. Feature extraction and search times are also compared. These are obtained using a Pentium-III, 1000 MHz machine in all experiments.

5.4.1 The ARP Methods for Sketch-Based Image Retrieval

Here, we apply the ARP method for the SBIR and compare the results with that of the AP method. The same preprocessing stages for both methods including statistical image abstraction, morphological thinning, and size normalization have been applied. In this test, we use $\beta = 3$, $J = 257$, and $\phi = 5^\circ$ as the internal parameters of the AP method (referred to as the APAI method in Chapter 4), resulting in a 72-entry feature vector. These conditions led to the best performance in the experiments already presented in Chapter 4. To make a similar footing for the ARP method, $M = 6$, $N = 12$, and $W = 257$ are chosen as the internal parameters of the ARP technique. As a result, a similar resolution, i.e. a 72-bin feature vector, is obtained for this method as well. The ART-PHOTO-BANK, containing 4000 images in the database and 400 hand-drawn sketches (explained in 4.4.2) is chosen as the test bed. The Manhattan ($\ell_1$) distance is employed for measuring the similarity between corresponding features. Figure 5.3 shows the Manhattan ($\ell_1$) distance calculated from all database images to an arbitrary input query. The corresponding least sixteen distance
images are retrieved and presented as the retrieval results. The smallest values are distributed within the horizontal axis of Figure 5.3 since similar images are randomly distributed in the database.

As an improved example, Figure 5.4 depicts the retrieval results of the ARP method applied on the same query which already has been used for the AP method (Figure 4.12 in Chapter 4). Employing the AP method, the NMRR=0.4200 had been obtained for this specific query. Now, the enhanced ARP technique generates NMRR=0.2000 from the same database with an equal resolution features (72 number of bins). The NMRR is calculated to show the performance of the system for this particular query. The NMRR is obtained for all queries and their average (ANMRR) is reported as the overall performance measure.

The ANMRR was computed to be 0.3070 and 0.2858 for the AP and the ARP methods, respectively. This confirms a better retrieval performance for the ARP method when applied to the SBIR. The reason is that the ARP technique captures more spatial detail than the AP technique as it overlays sectors, instead of slices, for data collection.
**Figure 5.4** Using APR method; the top 16 retrieved images, in row order, when posing a sketched image (upper-left) with 8 similar images in the ground truth, NMRR=0.2

### 5.4.1.1 A Time-Based Comparison of the ARP and AP Methods

It is important to note that although the retrieval power has been improved using the ARP method, nevertheless it requires more computational power for feature extraction. The off-line process, which is in the database population phase takes more longer time as the size of the database increases. However, this is tolerable since at the population phase the speed is a non-critical issue. On the other hand, query feature extraction which occurs at the on-line phase is only for one submitted query and can be neglected. More precisely, the average feature extraction times using images in the ART-PHOTO-BANK, the FET parameter was computed to be 2.86 Sec. and 3.12 Sec. for the AP and the ARP methods respectively. This means using the
AP method, a time of 3 hours, 10 minutes, and 40.08 seconds is required to feature extract all images in the database (4000 images). For the ARP method this time is 3 hours, 28 minutes and 0.12 seconds which is about 18 minutes longer. The time difference for one image is only 0.26 Sec. which is non-sensible.

5.4.2 The ARP Method for Edge Image Matching

As a new application of the proposed method, edge image matching is investigated in particular. As stated in the background section, image segmentation is a computational intensive stage in many image and vision-based applications. Avoiding segmentation is highly desirable and could significantly speeds up many recognition and retrieval tasks. The ARP method is a segmentation-free approach which can be applied for edge image matching. To this aim, in the following we present the results of experiments conducted to justify the ARP method for edge image matching.

5.4.2.1 The Effect of Parameter Variations

To show the effect of parameter variations of the ARP method when applied for edge image matching, different numbers of angular and radial partitions \((N \text{ and } M)\), using several size normalization parameters \(W\) were tested. We applied the ARP method on a database of 4320 images. The database was made by choosing 60 different pictures of "Animals have young" movie from the MPEG-7 content set V14, and rotating each picture successively. The primary frame size was \(352 \times 288\) pixels and each frame was rotated 71 times in \(5^\circ\) steps. To discard empty parts created after rotation, we cropped the central \(200 \times 200\) square and put it in our database. The cropped images are not only rotated versions of the original image but also are slightly translated, with some extra and truncated parts near the borders (see Figure 5.5 for some examples). The Canny operator, as an edge detector, with \(\sigma = 1\) and Gaussian mask of size 9, was used to obtain the edge map of all images. Size normalization, using nearest neighbor interpolation, was applied on the edge images. To evaluate the accuracy of the proposed approach, we applied the original images as queries (60 images) while considering the rotated-cropped ones as database entries (4320 images).
Figure 5.5 Image examples: the first column shows the original images and the next columns are the variants (rotated, small translated and eroded)
Table 5.1: The ANMRR of the ARP method with 3 radial and varying angular partitions using the movie derived database (4320 images)

<table>
<thead>
<tr>
<th>Normalized size</th>
<th>3 × 4 partitions</th>
<th>3 × 6 partitions</th>
<th>3 × 8 partitions</th>
<th>3 × 9 partitions</th>
<th>3 × 12 partitions</th>
<th>3 × 24 partitions</th>
<th>3 × 36 partitions</th>
<th>3 × 72 partitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>101×101</td>
<td>0.2921</td>
<td>0.2269</td>
<td>0.1538</td>
<td>0.1853</td>
<td>0.1922</td>
<td>0.1660</td>
<td>0.1661</td>
<td>0.1679</td>
</tr>
<tr>
<td>129×129</td>
<td>0.2625</td>
<td>0.1847</td>
<td>0.1090</td>
<td>0.1402</td>
<td>0.1441</td>
<td>0.0934</td>
<td>0.1010</td>
<td>0.0865</td>
</tr>
<tr>
<td>201×201</td>
<td>0.2072</td>
<td>0.1250</td>
<td>0.0552</td>
<td>0.0806</td>
<td>0.0832</td>
<td>0.0464</td>
<td>0.0451</td>
<td>0.0257</td>
</tr>
<tr>
<td>257×257</td>
<td>0.2216</td>
<td>0.1334</td>
<td>0.0694</td>
<td>0.0903</td>
<td>0.0931</td>
<td>0.0538</td>
<td>0.0518</td>
<td>0.0293</td>
</tr>
</tbody>
</table>

Table 5.1 shows retrieval performance using 3 radial partitions with varying, coarse to fine, angular partitions. The observation indicates that, for all normalized sizes, the performance improved by increasing the number of angular partitions in most cases. There are however, some exceptions in the table especially around 40° angular partitions. It appears that there are two contradictory factors: a) benefits of capturing more details using small partitions and b) robustness against extra and truncated regions using large partitions. The exceptions in the results seems to be due to the advantage of medium size partitions which can tolerate small variations (rather than rotation) caused by translation and normalization.

Table 5.2 exhibits the ANMRR using 12 angular partitions with different radials. It is observed that increasing the number of radial partitions improves the retrieval performance for all normalized sizes, with no exceptions.

Increasing the number of angular or radial partitions yields a bigger number of entries in the final feature vector which slows the on-line matching process. It also needs more storage and memory space. Therefore, a compromised decision on the number of partitions must be reached based on the application in use.

The effect of normalization is also depicted in Tables 5.1 and 5.2. The retrieval performance is the best in the normalized size of 201×201 as it is chosen close to the original image size. However, other sizes (101×101, 129×129 and 257×257) exhibit slow degradation of retrieval performance compared to 201×201 normalized size. It proves the robustness of the ARP method against varying image size.
TABLE 5.2: The ANMRR of the ARP method with 12 angular and varying radial partitions using the movie derived database (4320 images)

<table>
<thead>
<tr>
<th>Normalized size</th>
<th>5 × 12 partitions</th>
<th>7 × 12 partitions</th>
<th>10 × 12 partitions</th>
<th>15 × 12 partitions</th>
<th>18 × 12 partitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>101×101</td>
<td>0.1263</td>
<td>0.1107</td>
<td>0.1002</td>
<td>0.0952</td>
<td>0.0910</td>
</tr>
<tr>
<td>129×129</td>
<td>0.0974</td>
<td>0.0786</td>
<td>0.0716</td>
<td>0.0646</td>
<td>0.0566</td>
</tr>
<tr>
<td>201×201</td>
<td>0.0443</td>
<td>0.0317</td>
<td>0.0244</td>
<td>0.0217</td>
<td>0.0200</td>
</tr>
<tr>
<td>257×257</td>
<td>0.0512</td>
<td>0.0361</td>
<td>0.0269</td>
<td>0.0253</td>
<td>0.0223</td>
</tr>
</tbody>
</table>

TABLE 5.3: The ANMRR of the ARP method using selected data sets (2160, 1440, 720 and 480 images)

<table>
<thead>
<tr>
<th>Partitions</th>
<th>10° rotated</th>
<th>15° rotated</th>
<th>30° rotated</th>
<th>45° rotated</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 × 12</td>
<td>0.0764</td>
<td>0.0789</td>
<td>0.0617</td>
<td>0.0972</td>
</tr>
<tr>
<td>3 × 8</td>
<td>0.0508</td>
<td>0.0557</td>
<td>0.0579</td>
<td>0.0472</td>
</tr>
</tbody>
</table>

In addition, to show the rotation invariance property of the proposed ARP method in respect to the rotation step applied to the images, we obtained retrieval performance for four other data sets. These include data sets of rotated images with the steps of 10°, 15°, 30° and 45°. The extracted data sets contain 2160, 1440, 720 and 480 images respectively. Again, we applied 60 different original images as queries while regarding rotated images as database entries. The results, as expressed in Table 5.3, indicate that by using 3 × 12 partitions, the best performance is obtained on the image data set of the 30° step. On the other hand, using 3 × 8 partitions, the retrieval performance related to the data set of 45° step is the best. The fact is based on matching of the rotated image with the angular partitions. However, the retrieval performance of other rotations is still reasonable.

The results presented exhibit the average performance. The retrieval process tolerates variations much better for some queries than the others. The $\ell_1$ (Manhattan) distance between two query examples ($q_{10}$, $q_{52}$) and the main database images (4320 images) are shown in Figure 5.6. Each of 72 adjacent values are related to different rotated versions of each 60 query images. $q_{10}$ (row 5 in Figure 5.5) is an example
where the proposed method does not tolerate variations (rotation, translation, erosion and occlusion). Since the $\ell_1$ distance of some non-similar images are less than the distance to the similar variations. (Figure 5.6-a). However, $q_{52}$ (row 3 in Figure 5.5) is an example where the ARP method tolerates variations as the $\ell_1$ distances to the query’s variations are less than the others (Figure 5.6-b)

![Figure 5.6](image-url) The Manhattan ($\ell_1$) distance using the ARP method to the database images for (a) the $q_{10}$, and (b) the $q_{52}$ query examples

### 5.4.2.2 Comparative Results

To compare the AP and ARP methods in the edge image matching application, four alternative approaches are selected. The Zernike moment invariants, MPEG-7’s edge histogram descriptor (EHD), histogram of edge directions (HED) and angular radial transformation (ART) methods (details have been presented in Chapter 3) were
adapted and applied on the same test data. For the Zernike moment invariants we used 36 moments as suggested in [156] resulting in a 36-entry feature vector. For the EHD method, the \textit{desired\_num\_of\_blocks} was set to 1100 (the default value) and the \((T_{\text{edge}})\) parameter set to zero (because the edge images are all binary). A 150-bin histogram was obtained employing local, semi-global and global histograms. Furthermore, we used \(k = 1\) in HED method, resulting in a 70-entry feature vector and a 35-entry feature vector was achieved using \(m = 3\) and \(n = 12\) in ART method as recommended in [63]. For the AP method (Chapter 4), we used normalized size \(J = 257\) and \(\phi = 10^\circ\), resulting in a 36-entry feature vector. Similarly, we set \(M = 3\) and \(N = 12\) in the proposed ARP method (Section 5.3) to make a similar length (36-entry) feature vector.

In order to create a uniform assessment situation for all methods, we ignored the quantization stage in the EHD and ART methods. This removes retrieval performance disadvantage of quantization for these methods. The \(\ell_1\) distance was used for measuring the similarity between all image features, while for the HED method a weighting factor of 5 for global bins, as recommended in [155], was applied.

Figure 5.7 shows the results expressed by the ANMRR. The results confirm that the proposed method yields the best retrieval performance (the lowest ANMRR i.e. 0.0832). The AP, ART and Zernike moment invariants methods also show reasonable retrieval performance i.e. 0.1335, 0.1803 and 0.3127 respectively but the EHD (0.7106) and HED (0.8552) techniques are not as robust to rotation as the others. It is also remarkable that the length of the feature vectors for the ARP, AP, ART and Zernike moments methods are almost the same (36,36,35,36) while the length of the feature vector is 70 for the HED and 150 for the EHD methods.

We have already compared feature extraction and search times (FET and SET) for the above methods, except the ARP method, in Section 4.4.3. Presently, Figure 5.8 exhibits the comparison of these criteria for the ARP and the AP methods. The FET and the SET are averaged times computed on the movie derived database (4320 images). As already mentioned for the application of SBIR, the ARP method needs more time for feature extraction since the partitioning scheme is more complex. Search times are nearly the same. This is due to the fact that both techniques use an equal length
Figure 5.7 Retrieval results of different methods with the ANMRR feature vectors (36-entry).

Figure 5.8 The feature extraction time (FET) and search time (SET) for the angular partitioning (AP) and the angular radial partitioning (ARP) methods

5.5 Chapter Summary and Conclusion

The chapter introduces a novel approach for image representation based on geometrical distribution of edge pixels. Image abstraction is applied in the preprocessing
stage and the resulting normalized images are the input to the feature extraction process. Object segmentation is not needed, thereby the input image may consist of several complex objects. For efficient description of an arbitrary edge image, we propose dividing the edge image into $M$ radial and $N$ angular geometric partitions (sectors). The local features are computed with accumulating normalized edge pixels in the image sectors. The image is then represented as a set of spatially distributed feature descriptors. Applying the Fourier transform and using the magnitude of the transformed vectors ensures the rotation invariance. The method is also scale invariant and tolerates small translations and erosions. Image matching is achieved by measuring the distance between corresponding feature vectors. The compact feature vector (36 entries) not only accelerates the on-line matching process but also minimizes the storage requirements. These characteristics make the approach attractive for large-scale image database searching.

The application of the ARP method for the SBIR was examined and compared with that of the AP method. The retrieval performance of the ARP is better than the AP method while the feature extraction speed for the AP method is higher. Experimental results in respect to the edge image matching application, using an image database derived from a movie, show its supremacy in retrieval performance as well.

In the next chapter we will introduce a line segment extraction approach and explain its applications in the sketch-based shape retrieval.
Chapter 6

Sketch-Based Shape Retrieval Using Line Segment Distribution

6.1 Introduction

This chapter presents a novel effective method for line segment extraction using chain code differentiation. The resulting line segments are employed for shape feature extraction. Length distribution of the extracted segments along with distribution of the angle between adjacent segments are exploited to extract compact hybrid features. The extracted features are exploited for sketch-based shape retrieval. Comparative results obtained from three proposed methods and four other well known methods within the literature have been discussed. Using MPEG-7 contour shape database (CE-1) as the test bed, the new proposed method shows significant improvement in retrieval performance for sketch-based shape retrieval. The Average Normalized Modified Retrieval Rank (ANMRR) is used as the performance indicator. Although the retrieval performance has been improved using the proposed method, its computational intensity is more than for some other methods. This results in a longer feature extraction time (FET).

The outline of the chapter is as follows: a brief background is presented in the next section. The new proposed method is detailed in Section 6.3 followed by comparative results and discussion in Section 6.4. Finally, Section 6.5 concludes the chapter.
6.2 Background

Humans can easily recognize objects from their shapes. Many applications including computer vision, object recognition, and image retrieval and indexing are likely to use shape features. Shape feature extraction has attracted a large body of research over the last two decades [74, 205, 206]. In the CBIR shape is exploited as one of the primary image features for retrieval purposes [4, 22]. Shape representation techniques fall into three main approaches: feature vector approach (most popular technique), transformation approach, and relational approach. The choice of a particular representation is usually driven by application needs. In the following we briefly explain these three approaches.

A shape is represented as a numerical vector in the feature vector approach. The difference between two shapes is evaluated based on the distance between perspective feature vectors. In the transformation approach [207, 208], shapes are distinguished by measuring the effort needed to transform one shape to another. Similarity is measured as a transformation distance. Since techniques in this approach perform run-time evaluation of shape differences and do not support indexing, their retrieval performances are inefficient. In the relational approach [4, 38], complex shapes (or the scene) are broken down into a set of salient component parts, which are individually described through suitable feature vectors. The overall description in the relational approach includes both the description of the individual parts and the relation between them. This approach is not commonly used in shape-based retrieval but, instead, is widely employed for the recognition of complex shapes or scenes.

The feature vector approach is generally divided into two categories: contour-based and region-based [72]. The former exploits shape boundary information while the latter uses the whole area of the shape. Contour-based methods have gained more popularity based on the simplicity of shape contour feature acquisition and the sufficiency of the contour to represent shape in many applications. The Fourier descriptors (FD) and autoregressive methods, two representative techniques in this category, are compared in [209]. The curvature scale space (CSS) approach is adopted by the MPEG-7 standard for extraction of a contour-based descriptor [210]. It is computa-
tionally expensive and highly dependent on the contour continuity. The 2D Fourier transform and the polar Fourier descriptors (PFD) have been used frequently for shape-based retrieval (detail is presented in Chapters 2 and 3).

In region-based techniques, all the pixels within a shape region are taken into account to obtain the shape representation. Moment descriptors are commonly employed in region-based methods to describe shape. Zernike moments and angular radial transform (ART) methods are exploited in the MPEG-7 to extract region-based shape descriptors [156, 210].

Although much work has been done in the area of image retrieval using shape queries, very few have considered hand-drawn shapes as the input query. Matusiak et al. [103] and Horace et al. [105] have previously reported using rough and simple hand-drawn shape as input query for sketch-based shape retrieval. The approach in [103] is based on the curvature scale space method that is computationally expensive and has been shown to be less efficient than the Fourier descriptors and Zernike moments techniques [83]. In [105] several dominant points are extracted for each contour using information derived from the convex hull and the contour curvature.

We have introduced the AP and ARP methods in Chapters 4 and 5 of this thesis, respectively. The nature of an image containing one isolated shape is different from an image including multiple objects. The former possesses an image plane which is more scarce than the image plane of the latter which has more pixels therein. Hence, the philosophy behind angular or radial partitioning, used in the AP and ARP methods, is not adequate for shape description even though it is reasonably sufficient for multi-component images.

In the next section, we present a new method for contour polygonization using chain code representation of the boundary shapes. The method is employed to extract efficient features for sketch-based shape retrieval.
6.3 Contour Polygonization Using Chain Code Differentiation (CPCD)

In this section the details of the chain-based line segment extraction method are discussed. The input of the method is a digitized curve $C$ derived by any contour extraction technique on a planar shape. In addition, any thinned sketched contour can be used as the input $C$. Since a chain code is a more succinct way of representing a contour, this representation is adopted here. First, the starting point of an 8-connectivity chain code is determined using raster scanning the curve plane $I$ [121]. The macro chain $A_i = \{a_1, a_2 \ldots a_{n_i}\}$, $i = 1, 2, \ldots, m$, where $m$ is the number of chains in $I$ and $n_i$ is the chain length, is obtained and put in a chain set $\{A_i\}$.

Note that for a simple shape from the database, there usually exists only one closed contour $C$ but for a corresponding sketched query, there are sometimes more than one curve per shape. This is due to occasional disconnectivities resulting from free-hand-drawing, scan resolution, and associated noise. The proposed method can cope with multiple-contour shape as well as one-contour shape since it considers several chains to be in the set $\{A_i\}$. For each $A_i$ in $\{A_i\}$ we apply the following steps (see Figure 6.1):

1. Eliminating chain noise: noisy points which make the chain over oscillating are eliminated by median filtering. Applying a third order one-dimensional median filter on the vector $A_i$ reduces the effect of such points adequately. Figures 6.1-b and c show the effect of reducing the number of chain points by median filtering.

2. Shifting operation: the standard chain code representation has the wraparound drawback [211]. For example, a line along $-22.5^\circ$ direction (in the forth quarter of the trigonometric circle) is coded as $\{707070\ldots\}$ using standard chain code representation. To eliminate or reduce such wraparound, a new modified code $B_i = \{b_1, b_2 \ldots b_{n_i}\}$ for each $A_i = \{a_1, a_2 \ldots a_{n_i}\}$ can be extracted by a shifting operation defined recursively as:
\[
\begin{align*}
    b_1 &= a_1 \\
    b_k &= g_k, g_k \in Z \mid (g_k - a_k) \mod 8 = 0 \text{ and } |g_k - b_{k-1}| \text{ is minimized for } k = 2, 3, \ldots, n_i
\end{align*}
\] (6.1)

The line along $-22.5^\circ$ direction is now coded as $\{787878\ldots\}$. Comparison of Figures 6.1-c and 6.1-d shows the wraparound effect.

3. Smoothing operation: the shifted chain code $B_i$ is then smoothed by a five-point Gaussian filter $\{0.1, 0.2, 0.4, 0.2, 0.1\}$ [211]. The resulting shifted and smoothed waveform is called $\Gamma(\theta)$, where $\theta$ is the traversing variable (Figure 6.1-e).

4. First derivative and break points extraction: $d\Gamma/d\theta$ determines the rate of change of $\Gamma(\theta)$ with respect to $\theta$. The extreme points of this derivative is considered as break points $(\zeta_i)$, if they are greater than a threshold $\tau$. The $\tau$ value determines the degree of coarse-to-fine approximation of the input curve with a polygon. Figures 6.1-f and g depict the resulting derivative and the corresponding extracted line segments to rebuild the shape.

The selection of threshold $\tau$ has a great influence on the rebuilding process. Figure 6.2 shows the effect of the variable $\tau$ and the resulting number of segments (NoS) for an example shape. As can be seen, smaller values of $\tau$ make the resulting polygon to resemble the contour curve more closely (with more line segments). For shape retrieval, we are not interested in very fine polygons because the extracted features need to represent only the overall structure of the shape and the details are not important. Therefore, $\tau = 0.3$ is chosen in our experiments. It is interesting to note that higher $\tau$’s can be used for extracting a major axis of the shape as shown in Figure 6.2-i.

The line segment $l_k$ which connects $\zeta_k$ to $\zeta_{k+1}$ is considered as the lineal approximation of the micro chain lying between the two points. This segment is employed to construct the approximating polygon. Although a finer approximation for the micro chain from $\zeta_k$ to $\zeta_{k+1}$ can be employed to obtain more line segments and consequently more precisely fitted polygons, experiments on many test data showed that
Figure 6.1 (a) An example thinned hand-drawn sketch, (b) the corresponding chain code (c) median filtered code, (d) shifted code, (e) smoothed representation, (f) derivative, and (g) extracted line segments
there is no significant improvement in the retrieval performance by applying such extra computations. This arises from the fact that overall structure of the shape can be well captured by a moderate number of segments. Therefore the set

\[ L_i = \{ l_k \} \]  \hspace{1cm} (6.2)

where \( l_k \) is the straight line segment connecting \( \zeta_k \) to \( \zeta_{k+1} \) will be used as the line segment set of chain \( A_i \). Finally, the union of all \( L_i \) sets, say \( L \), for \( i = 1, 2, \ldots, m \), is obtained:

\[ L = \{ L_1 \} \cup \{ L_2 \} \cup \ldots \cup \{ L_m \} \]  \hspace{1cm} (6.3)

\( L \) is the line segment set of the underlying shape which is used as polygonal approximation of the boundary curve \( C \).

6.3.1 A Hybrid Shape Similarity Measure

The length of extracted line segments (\( l_k \in L \)) are rotation and translation invariant. Normalizing the length by the maximum length makes it scale invariant as well. In
addition, the angle between successive segments in $L$ is scale, translation and rotation invariant.

For the purpose of shape recognition and retrieval, we employ the aforementioned principles to extract two discriminating and affine transforms invariant vectors $\Psi$ and $\Phi$ using the following procedure:

1. Initialize $\Psi$ (30 entry) and $\Phi$ (18 entry).

2. For each segment in $L$:
   - compute the length,
   - normalize the length with the maximum length, and
   - uniformly quantize the normalized length to 30 equal parts and add one to the corresponding entry of $\Psi$.

3. For each segment pair in $L$ which are adjacent:
   - compute the angle between the segments (corner angle), and
   - uniformly quantize the corner angle to 18 equal parts (to make steps of $10^\circ$) and add one to the corresponding entry of $\Phi$.

Next, we combine these two feature vectors to make hybrid shape features used for similarity measure. For this, the distance between two different shapes is computed using the combination of Euclidian distances obtained individually from corresponding $\Psi$ and $\Phi$ vectors. As mentioned before, one of the difficulties involved in integrating different distance measures is the difference in the range of associated distance values. In order to overcome the problem, the two distance values are normalized first to be within the same range of $[0,1]$ and then are integrated with a weighting scheme. More precisely, let $Q$ be a query image and $P$ be a database image. Let $D^n_\Psi(Q,P)$ denotes the normalized Euclidian distance between $Q$ and $P$ on the basis of geometric segment’s length and $D^n_\Phi(Q,P)$ denotes their normalized Euclidian distance on the basis of corner angle. The normalization is accomplished as follows:

$$D^n_\Psi(Q,P) = \frac{[D_\Psi(Q,P) - \text{mindist}_\Psi]}{[\text{maxdist}_\Psi - \text{mindist}_\Psi]}$$

$$D^n_\Phi(Q,P) = \frac{[D_\Phi(Q,P) - \text{mindist}_\Phi]}{[\text{maxdist}_\Phi - \text{mindist}_\Phi]}$$

(6.4)
where \( D_\psi(Q, P) \) and \( D_\Phi(Q, P) \) are the Euclidian distances between \( Q \) and \( P \) based on geometric length and based on corner angle respectively. \( \text{mindist} \) and \( \text{maxdist} \) are the minimum and the maximum distance values of the query image \( Q \) to the database images according to the corresponding distance used (i.e. \( D_\psi \) or \( D_\Phi \)).

Finally, an integrated and hybrid distance \( D \) between \( Q \) and \( P \) is defined as:

\[
D(P, Q) = \frac{w_1 \times D_\psi^n(P, Q) + w_2 \times D_\Phi^n(P, Q)}{w_1 + w_2}
\]

(6.5)

where \( w_1 \) and \( w_2 \) are the weights assigned to to the length-based distance and the angle-based distance, respectively. In current implementation of this thesis we have used \( w_1 = w_2 = 1 \).

### 6.4 Experimental Results

To evaluate the retrieval performance of the proposed method, in comparison with other well known methods, seven different approaches were implemented. The Fourier descriptors (FD) [72], PFD method [82], Zernike moment invariants (ZMI) [156], the ART [22, 210], AP (Chapter 4), ARP (Chapter 5) and the proposed CPCD (Section 6.3) methods were employed to extract features for sketch-based shape retrieval. The comparative results are presented in this section.

The MPEG-7 contour shape database CE-1, set A1 and A2 [72], was used as the common test bed for all methods. The database consists of pre-segmented shapes, defined by single closed contours acquired from real world objects. It takes into consideration the common shape distortions and the inaccurate nature of shape boundaries in segmented shapes. Set A1 is for the test of scale invariance and contains 420 shapes of 70 classes (6 in each class). An image of each class was scaled by the following factors: 200\%, 30\%, 25\%, 20\%, and 10\%. In a similar manner, set A2 contains 420 images created by rotation of the original 70 shapes by the following angles: 9°, 36°, 45°, 90°, and 150°. Consequently, appending sets A1 and A2 forms a database of 840 images in 70 groups, each with 12 similar images. Figure 6.3 depicts one image from each class in the database and Figure 6.4 shows examples of variation within a class.
Figure 6.3 Shapes from 70 different classes of the MPEG-7, CE-1 database
Since the database images are all binary, we obtained the boundary contour of each image as the set of foreground pixels which have at least one neighboring background pixel. We also collected 105 different hand-drawn sketches similar to randomly selected shapes (Figure 6.5 shows some examples). They were morphologically thinned to represent shape's boundary contour. To be able to evaluate scale and rotation invariance properties, the database and the sketched images were chosen to have varying sizes and directions.

Once again, the ANMRR criterion is used for measuring the retrieval accuracy (described in 2.7.2). Figure 6.6 exhibits the resulting ANMRR for different methods. As can be seen, the proposed CPCD method shows the best retrieval performance (i.e. ANMRR=0.3408). This is due to the ability to capture global and local structural similarities between database images and the sketched queries. In other words, the extracted features are capturing the overall structural properties of the shape using the length distribution of predominant sides. In addition, the local properties of the shape are also exploited utilizing angle size distribution of the corner points.
Figure 6.5 Examples of sketched shapes
6.4.1 Discussion

It is worthwhile to note that as the ZMI and ART methods are region-based approaches, their retrieval performances are the lowest in the current application, i.e, 0.4819 and 0.4687, respectively. The ARP method shows more effective performance (0.3739) than the FD (0.4632), AP (0.4312), and PFD (0.4125) methods, respectively. The varying degree of performance in these methods arises from different algorithms they employ for feature extraction.

The FD and PFD methods are based on the Fourier transform in the Cartesian and polar coordinates, respectively. The basis function used in these methods can effectively capture the similarity between different contour shapes as they exploit magnitude and phase information of the Fourier coefficients. However, they cannot tolerate disconnections. Moreover, this kind of transform is more effective for contour-based shape retrieval, that is when the outline of the shape is used for feature extraction.

The AP and ARP methods, described in Chapters 4 and 5 respectively, also exhibit good performances where there is enough spatial information in the relevant slices or sectors. In the current application (sketch-based shape retrieval), that uses only the
contour data, there is no sufficient information in terms of the number of pixels in such areas. This shortcoming can be partially overcome by refining the partitioning scheme used in the AP and ARP approaches which adversely increases the extraction time.

Figure 6.7 shows average feature extraction times for the aforementioned algorithms using a Pentium-III, 1000 MHz machine on a collection of both the database and the sketched shapes. The AP and ARP methods possess the shortest feature extraction times as they are performed in the pixel domain and simply collect data from a few associated regions. The AP extraction time is shorter as its partitioning scheme is less complicated than the ARP’s scheme. The FD, CPCD, and PFD methods need a moderate time for feature extraction and the ZMI and ART methods are the slowest. Longer extraction times for the transform-based methods (i.e. FD, PFD, ZMI, and ART) arise from higher computational cost of handling complex basis functions. For the CPCD method, the convolution and differentiation operations involved in the smoothing and the break point extraction stages are the most time consuming steps in the algorithm.

![Figure 6.7 Average feature extraction time for different methods using both database and sketched shapes on a Pentium-III, 1000 MHz machine](image-url)
6.5 Chapter Summary and Conclusion

A novel and effective feature extraction approach for sketch-based shape retrieval is proposed in this chapter. The approach is based on the distribution of line segments extracted by chain code differentiation. The extracted features are affine transform invariant. The boundary contour is approximated by a polygon using a line segment set. Predominant characteristics of the polygon, sides and corners, are employed in a feature extraction algorithm. Extracted features have a hybrid nature combining two different feature vectors. The approximating polygon can be adjusted within a wide range of very fine to very coarse based on the application’s requirements.

Experimental results using the MPEG-7 shape database CE-1, set A1 and A2 including 840 shapes, and 105 different sketched queries, confirm the robustness and retrieval performance improvement of the proposed method. The ANMRR and feature extraction time (FET) are used as performance criteria. It is shown that the proposed method has the best retrieval performance in the current application. However, its computational intensity is moderately high.

In the next chapter we will present a case study investigating signature-based document retrieval.
Chapter 7

Case Study: Signature-Based Document Retrieval

7.1 Introduction

In this chapter, we present a case study of signature-based document retrieval using (a) document image decomposition and (b) signature verification procedures.

The previously described methods, including angular partitioning (AP), angular radial partitioning (ARP), and contour polygonization using chain code differentiation (CPCD), are applied for feature extraction of human signatures existing in a document database. The database contains document images with Persian/Arabic text combined with English text, headlines, ruling lines, trade mark and cursive signature. Signature region extraction is accomplished first, exploiting connected component analysis and labelling in conjunction with special characteristics of the signature shape. Next, feature extraction is applied using four different techniques. Comparative results show performance differences among the methods as well as the tradeoff between accuracy and speed.

The chapter is organized as follows: an introductory background is given in the next section. A new cursive signature region extraction technique is explained in Section 7.3 and comparisons of four signature feature extraction approaches for the retrieval accuracy and computational speed are provided in Section 7.4. Finally, chapter sum-
mary and conclusion are presented in Section 7.5.

7.2 Background

Multimedia processing is one of the major areas in information technology (IT). Automatic document analysis is a fundamental issue in many applications including optical character recognition (OCR), form and bank check reading and document image storage and retrieval. Document image understanding has been an interesting research area for a long time [212] and covers a variety of documents such as facsimiles [213], bank checks [214], business letters [215], forms [216], postal mail parcels [217] and technical articles [218, 219].

Classification of a document image into text and graphics is investigated in [220]. The approach is based on the different textural properties of graphics and non-graphics components in the document. Li and Gray [221] developed a method for segmenting document images into four classes: background, photograph, text, and graph. The distribution patterns of wavelet coefficients in high frequency bands are employed to extract features for the proposed classification.

Background thinning is used for page segmentation in [222]. The approach is effective but sensitive to content and computationally expensive. Wang and Chi [223] improved the approach by speeding up the process through applying a hierarchical content classification and script determination. They introduced a neural network based classifier to classify a sub-image into text or picture. They also developed an algorithm that can determine Chinese and Western scripts in the text region using a three-layer feed forward network.

One of the most important content components in official/business letters and bank checks is the signature. The effective extraction and verification of the signature play an important role in automatic processing of such documents. Moreover, paperless organizations are growing fast and the interaction with other paper-based organizations and individuals needs efficient methods for converting a paper-based document to an electronic version. Automated document and check signature processing tech-
niques consist of two main modules: a low-level processing for signature extraction and a high-level processing for signature verification.

English signature analysis, verification and recognition have been studied extensively. They could be divided into two broad areas: on-line and off-line. A comparison between wavelet-based and function-based on-line signature verification has been reported by Da Silva and De Freitas [224] whilst Justino et al. [225] have focused on off-line signature classification using Hidden Markov Models. The Persian/Arabic signatures however, have different characteristics. They usually are cursive sketches which are independent of the person’s name while English signatures are often reshaped handwritten names.

The Zernike moment invariants (ZMI) and projection methods are studied for off-line Chinese signature verification in [226]. Different orders of moment invariants were tested and the results show the better performance of the ZMI method compared to the projection method.

We have introduced the AP, ARP and CPCD techniques in Chapters 4, 5, and 6 of this dissertation, respectively. These methods were basically designed for sketch-based image and shape retrieval. They, meanwhile, can be applied for cursive signature recognition and retrieval. This is based on the fact that cursive signature is a special case of hand-drawn image and the above methods prove to have the ability of effective feature extraction for hand-drawn images and isolated shapes.

7.3 Document Image Retrieval Using Signature Image

We consider a framework, depicted in Figure 7.1, for signature-based document retrieval. Any document image in the database is processed off-line through a signature region extraction and a feature extraction processes. The first process extracts the signature region from the whole image and delivers it as an input to the second process. This process thereupon extracts a compact feature vector used for similarity measurement. The query signature is feature extracted on-line and the extracted fea-
tures are compared with all those extracted from the database. Documents are ranked based on the distance to the query signature and the top $N$ documents are retrieved. Different retrieval accuracy measures define $N$ in different ways. For example, for the ANMRR measure, $N$ is determined by the number of similar signatures exist in the database. In the following, we explain the signature region extraction process.

### 7.3.1 Signature Region Extraction

The main objective of this phase is to determine the smallest rectangle (bounding box) surrounding the cursive signature in the document. At the beginning, connected component analysis and labelling are employed to designate existing components in the document image. More precisely, let $I$ be the binary image of the original document. The connected components labelling [227] that performs the unit change from pixel to region is employed to label $I$ as follows: all pixels that have value binary 1 and are connected to each other in an 8-connectivity neighborhood are given the same identification label. The label is a unique index of the region to which the
pixels belong. For efficient implementation of the labelling algorithm the input image \( I \) is first run-length encoded and then assigned preliminary labels while scanning the runs and recording label equivalences in a local table. The equivalence classes are resolved next and the runs are relabelled based on the resolved equivalence classes.

We used the MATLAB function \( [J, n] = 	ext{bwlabel}(I, 8) \) for labelling. This function returns the matrix \( J \), of the same size as \( I \), containing labels of the connected objects in \( I \), where 8 specifies 8-connected objects. The elements of \( J \) are integer values greater than or equal 0, where pixels labelled 0 are the background. The pixels labelled 1 make up one isolated object, the pixels labelled 2 make up a second object, and so on. Number of connected objects found in \( I \) is returned in \( n \).

Next, geometric properties including area, circularity, eccentricity, size and position of different labelled objects are exploited to extract the signature region. The main idea is to employ unique characteristics of cursive Persian/Arabic signatures in the text to distinguish the signature region. Heuristic rules differing a cursive signature from other components are employed in this stage. For example, characters are distinguished by the size characteristic, and headlines and ruling lines by the circularity and eccentricity properties. Logos in the image are determined by the area and position characteristics. These heuristic rules are hypothesized first and then fine tuned empirically.

The output of the signature region extraction process is a bounding box within the document image which includes the cursive signature. If there is no cursive signature in the document the process returns nil. In most cases the region is correctly selected. However, there are some cases where the depicted region includes additive parts such as text or ruling lines. In some minor cases a portion of the signature is outside the box and missed out. The unwanted extra parts in the signature region are partially eliminated by applying the algorithm given in [228]. The adverse effects of remaining unwanted regions and missing parts are eliminated by choosing a proper feature extraction approach as discussed in the next section.

The bounding box of the signature region is finally normalized to \( J \times J \) pixels using nearest neighbor interpolation. Normalization is accomplished to gain the size-
invariance property. This is necessary since usually the person’s signature size is
differed in different instances. This region of interest called $B$ and is fed forward
to any arbitrary technique selected for signature feature extraction. In the next sec-
tion some alternative feature extraction techniques and justifications employed in this
study are discussed.

7.4 Feature Extraction and Comparative Results

To study the feature extraction stage, four different approaches were selected and sev-
eral experiments were conducted to select a reliable technique for signature feature
extraction. The approaches selected are the AP (Chapter 4), ARP (Chapter 5), CPCD
(Chapter 6), and ZMI [156, 226] methods. We used a document image database con-
taining 425 documents signed by 85 different persons who have Persian or Arabic
cursive signatures. The content of each document image includes a variety of mixed
text of Persian, Arabic, and English alphanumerics with different fonts and sizes, a
company logo, some horizontal and vertical lines and a cursive signature. Figure 7.2
shows two examples.

All documents are processed to generate a feature vector for the signature within
the image. The processing involves signature region detection using the procedure
described in Section 6.3. The signature region was found correctly in 418 cases
(98.35%) and the signature extracted completely in 413 cases (97.18%). This is due
to the fact that some cursive signatures have several disjoint parts while the algorithm
focuses on neighboring connected parts.

Subsequently, a compact feature vector, using the four above mentioned approaches,
is generated for the signature region of the document. Feature extraction is also
applied for the query signature $q$ presented by the user as a handwritten signature.

Using average size of signature regions in the database as a guideline, we select
$J = 129$ for size normalization. Thus, $B$ is chosen to be $129 \times 129$ pixels for the
region of interest. The ANMRR with $NG(q) = 5$ for all $q$’s, $L = 10$, and $Q_{total} = 85$
is used as the retrieval accuracy measure (see Section 2.7). The Manhattan distance
Figure 7.2 Examples of document images

($\ell_1$) is used as the similarity measure between corresponding N-dimensional vectors. Feature extraction time (FET) is employed to show the speed of each technique. The FET parameter is the average of feature extraction time for all signature images (database and query images) computed for each method using a Pentium-III, 1000 MHz machine.

For the AP method (Chapter 4), which partitions $B$ into slices and then extracts features using 1D Fourier transform, five alternative numbers of slices were examined (4, 8, 16, 32, and 64). For each, the ANMRR and FET parameters were obtained. Table 7.1 presents the results. The best ANMRR is achieved when 16 slices are used. The reason is that bigger slices lose capturing spatial characteristics while very fine slices are prone to noise and consequently unable to retrieve the corresponding images effectively.

For the ARP method (Chapter 5), which splits $B$ into sectors and then extracts fea-
TABLE 7.1: The ANMRR and FET of the AP method using different numbers of slices

<table>
<thead>
<tr>
<th>Number of Slices</th>
<th>4</th>
<th>8</th>
<th>16</th>
<th>32</th>
<th>64</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANMRR</td>
<td>0.5130</td>
<td>0.4212</td>
<td><strong>0.4034</strong></td>
<td>0.4108</td>
<td>0.4196</td>
</tr>
<tr>
<td>FET (Sec.)</td>
<td>0.40</td>
<td>0.44</td>
<td><strong>0.47</strong></td>
<td>0.52</td>
<td>0.56</td>
</tr>
</tbody>
</table>

TABLE 7.2: The ANMRR and FET of the ARP method using different numbers of angular radial partitions

<table>
<thead>
<tr>
<th>Partitions</th>
<th>$4 \times 4$</th>
<th>$8 \times 4$</th>
<th>$8 \times 8$</th>
<th>$8 \times 16$</th>
<th>$16 \times 4$</th>
<th>$16 \times 8$</th>
<th>$16 \times 16$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANMRR</td>
<td>0.3710</td>
<td>0.3314</td>
<td><strong>0.3017</strong></td>
<td>0.3219</td>
<td>0.3310</td>
<td>0.3418</td>
<td>0.3383</td>
</tr>
<tr>
<td>FET (Sec.)</td>
<td>0.43</td>
<td>0.46</td>
<td><strong>0.54</strong></td>
<td>0.59</td>
<td>0.57</td>
<td>0.60</td>
<td>0.66</td>
</tr>
</tbody>
</table>

atures employing 1D Fourier transforms for each group of concentric sectors, seven alternative partitions were examined ($4 \times 4$, $8 \times 4$, $8 \times 8$, $8 \times 16$, $16 \times 4$, $16 \times 8$, and $16 \times 16$). The ANMRR and FET criteria were computed for each partitioning scheme. The results are depicted in Table 7.2. The best ANMRR is obtained using the $8 \times 8$ scheme. The reason turns out to be related to equal diversity in radial and angular directions of the signature pixels. In other words, the importance of radial and angular partitions are the same for cursive signatures. Smaller and bigger schemes suffer from the same difficulties as described for the AP method. The exception of decreasing retrieval performance at $16 \times 16$ (last column) arises from the equal radial and angular resolutions, which cause better retrieval performance than the $16 \times 8$ divisions.

The FET parameter shows a direct correlation between the number of slices and the number of sectors for the AP and ARP methods respectively. This is obvious since as more sections are involved more computation power is required. Furthermore, comparing the FET parameter of the AP and the ARP methods indicates that the ARP methods requires more computation time for feature extraction. This is due to more complex data collection and transformation schemes involved in the ARP method.

To investigate significance of distributions of the line segment length and the angle between successive segments in the CPCD approach (Chapter 6), we conducted the
following experiments. First, the line segments’ set was obtained applying chain noise elimination, shifting and smoothing operations followed by first derivative and break points extraction as explained in 6.3. Next, alternative resolutions were chosen for the $\Psi$ and $\Phi$ vectors and the resulting hybrid features were used for comparison of the signature images. Table 7.3 depicts the ANMRR and the FET parameters for different resolutions. As can be seen, the best result is generated when the resolution of the length vector is 30 (the $\Psi$ vector) and the resolution of the angle vector is 45 (the $\Phi$ vector). This implies that the angle characteristic is more significant than the length characteristic for the cursive signatures. This relates to the existence of a higher curvature in a signature than a contour shape.

The FET parameter for the CPCD approach is slightly higher than for the case of shape retrieval (Chapter 6). The reason is the existence of a higher number of chains in a signature compared to a contour shape. Moreover, higher resolutions require more computation power and lead to a longer FET. An interesting characteristic of the CPCD method, which can be realized comparing Tables 7.1, 2, and 3 is its lower sensitivity to varying resolutions with respect to the AP and ARP methods. In other words, there are smaller variations in the ANMRR values in Table 7.3 than those in Tables 7.1 and 7.2 regarding the AP and ARP methods.

The ZMI method, the last alternative method, was implemented as described in the MPEG-7 standard [156] (we have mentioned that it is applied for the Chinese signature verification in [226]). Here, a 36-entry feature vector is generated based on Zernike moment invariants and used for similarity measurement. The ANMRR and the FET parameters for the ZMI method were computed to be 0.3835 and 0.95 Sec., respectively. Figure 7.3-a shows the best ANMRRs computed for each method and Figure 7.3-b depicts the associated FETs. The ARP method’s performance is the
best. The reason of why the ARP approach performs better than CPCD for signatures whilst the CPCD method is better for hand-drawn shapes (Chapter 6) is that the ARP method is more robust against extra and eroded parts since it considers a number of pixels in a relatively large area as the key concept for feature extraction. The signature region extraction technique, described in the last section, incorporates some extra and missing parts of the signature which cannot be easily tolerated by the CPCD method. The second reason is that the Persian/Arabic cursive signatures have a more dispersal pixel distribution in the region of interest than hand-drawn shapes. Therefore, the ARP can capture more detail information in the current application. Furthermore, the rotation invariance of the ARP technique is more robust than rotation invariance of the CPCD method since the ARP methods relies upon Fourier transform for this property.

The AP method performance is the poorest one. This is based on the fact that it looks only into angular slices for feature extraction. Although the technique shows good retrieval power for sketch-based image retrieval where a variety of complex multi-component images are involved (e.g. art work and photograph images), in signature images, there is not enough diversity in subsequent slices to be captured by the AP method as salient image features. Thus, it is hard to extract significant features from cursive signatures using the AP technique.

As stated before, the search time (SET) parameter for any feature vector-based tech-
nique is mostly dependent on the distance measure selected and the dimension of the associated features. In our study, the Manhattan distance was employed for all methods. Hence the SET parameter shows proportionality to the vector’s length.

7.5 Chapter Summary and Conclusion

A new approach for signature-based analysis and verification of document images has been proposed. In particular, cursive Persian/Arabic signatures were investigated. Connected component analysis and labelling along with geometric properties were used to determine the signature region. The signature image was then processed with four alternative methods including the AP, ARP, CPCD, and ZMI techniques to produce feature vectors. The extracted vectors serve as N-dimensional features for measuring the similarity between database and query signatures.

We have investigated several variations of the AP, ARP, and CPCD techniques to realize the best conditions for the case of signature-based document retrieval. The ANMRR and FET criteria were derived for all methods. Retrieval performance of the ARP method is the best while the CPCD method is ranked next. The speed of the AP method for feature extraction is the highest but its retrieval performance is the lowest. The CPCD method possesses the longest feature extraction time while the ARP method’s FET criterion is very close to the minimum (which belongs to the AP method).

In conclusion, the ARP method with $8 \times 8$ angular radial partitions could be considered as the best method in this application among those undertaken the tests. This arises from the fact that the retrieval performance of the ARP method is the best (i.e. the lowest ANMRR) while the associated FET measure is better than two other methods.
Chapter 8

Summary, Conclusion and Further Work

8.1 Introduction

This final chapter presents a thesis summary and conclusion followed by some new directions and improvements.

The thesis has considered content-based retrieval from image databases using sketched queries. The sketch-based image retrieval (SBIR) is a subbranch of the well known content-based image retrieval (CBIR). The main objective of the SBIR is to effectively facilitate the CBIR through sketch-based tools.

The sketch-based tools provide an easy and effective man-machine interface in multimedia access and results in more user satisfaction. In particular, searching through image databases becomes much easier and more reliable if a sketched image is applied as the initial query. To build such a user interface, sketch-based features should be efficiently determined and included in the image database management system.

The main goal of this work has been to develop efficient features for the SBIR. This problem was considered in two main categories: sketch-based image retrieval which deals with general multi-component images, and sketch-based shape retrieval which deals with individual objects. For the first category, the aim was to secure scale and rotation invariant features as well as segmentation-free processing. For the second
category, the features were expected to be not only scale and rotation but also translation invariant. This is due to the fact that the position of the input shape may vary compared to the position of the target shape(s) in the image plane. In both categories, the extracted features could act as an index for the retrieval task from large-scale image databases. The aims and objectives of this thesis have been successfully achieved.

Hereafter, we present a thesis summary and conclusion followed by discussion of some future works arising in this research area.

### 8.2 Thesis Summary and Conclusion

The thesis has addressed an important issue in the CBIR, which significantly facilitates image querying. It has considered the SBIR where the input image is a simple black and white hand-drawn image while the database images can be full-color and textured. This matter has been investigated both in multi-component and single-component images. The work has proposed several approaches mainly for sketch-based feature extraction, as summarized below.

**Chapter 1** provided a brief introduction to the CBIR and SBIR. A deeper insight into the motivations, problem statement, and the goals of this research were presented in this chapter. The major contributions of the thesis were outlined as well as a list of publications resulting from the research.

**Chapter 2** presented a more general literature survey on the related issues in the image retrieval and the CBIR. The chapter includes feature extraction techniques in the pixel and compressed domains. Different type of queries, variant similarity measures, and two major retrieval performance approaches were discussed. Semantic issues in image retrieval were briefly explained. Good standing CBIR systems both commercial and research products were also introduced in this chapter.

**Chapter 3** presented a specific study of the methods that can be employed directly or can be adapted accordingly for the SBIR. Six different approaches were described in detail in this chapter. The approaches were implemented and their results were
compared to the results from the proposed methods in the subsequent chapters.

**Chapter 4** devised a novel technique based on image abstraction and angular partitioning (AP). The image abstraction procedure has been implemented for full-color and textured images within the database using a statistical approach to find strong edges. Query images have been abstracted utilizing the morphological thinning. Angular partitioning of the scale-normalized abstract images into successive slices, and applying the 1D Fourier transform generated scale and rotation invariance features. The features have been used to build an index for the associated database. The proposed approach outperformed five out of six alternative methods in scale invariance test. It showed the best retrieval performance in rotation invariance test as well. The FET and SET parameters which exhibit the time-based performance were computed and compared to the alternative techniques.

**Chapter 5** developed a modified version of the partitioning scheme that is more precise than the AP technique. Here, instead of slices, several groups of concentric sectors were overlaid onto the abstract image. Magnitude of the coefficients of the 1D Fourier transform, applied to each group have been selected as the scale and rotation invariance features. The features were also robust against translations. The experimental results confirmed significant improvement in the retrieval performance with a slightly slower procedure. Moreover, edge image matching, using the angular radial partitioning (ARP) method in a movie search application has been studied. The experimental results showed the best retrieval performance of the proposed ARP method, compared to alternative approaches with a tantamount speed.

**Chapter 6** introduced a new category of sketch-based image retrieval. Here, the aim was to find suitable features which can be used for searching a database of isolated shapes using hand-drawn sketches. Due to scarceness of such images compared to general art works and photograph images, the previous approaches (AP and ARP) seemed to be inadequate. For this, a new line segment-based approach (CPCD) has been proposed. The approach utilizes the chain code differentiation to generate a line segment set which describes the contour shape geometrically. The set was used for feature extraction. The hybrid features have produced the best retrieval performance among all seven alternative approaches. The CPCD feature extraction speed was
computed to be lower than the AP and ARP approaches but higher than all other alternatives.

Chapter 7, finally, investigated a case study of signature-based document retrieval. In this case, we proposed a signature region detection technique which determines the region of interest. The technique is based on connected component analysis and labelling, and exerts heuristic rules to find a cursive signature within the given document. The signature region was fed to several feature extraction algorithm as the input. The retrieval performance of the ARP method was computed as the best.

In each chapter, we have provided all necessary mathematical basis, justifications and conditions of the experimental tests. We have always used the ANMRR criterion (which was developed in the MPEG-7 standard) for performance evaluation. This is based on the fact that the ANMRR measure considers not only the Recall and Precision information but also the rank information among the retrieved images. It is independent of the data size and normalized to [0,1]. The definition of the ANMRR has been given in Chapter 2.

In conclusion, this thesis has presented several novel techniques for invariance feature extraction used in the SBIR. They achieve significant improvement in retrieval accuracy. The proposed techniques are divided into two broad categories. The first was for segmentation-free SBIR for inhomogeneous background and multi-component images. The proposed methods (i.e. the AP and ARP methods) were based on the spatial distribution of strong edge pixels in normalized abstract images using the Fourier transform to achieve rotation invariance. This discipline has been showed to be suitable for general images and cursive signatures. The second discipline has targeted singular shapes in the sketch-based retrieval. The proposed method (i.e. CPCD) has used the geometric properties of the line segment distribution to extract invariant and effective features. The extracted features were confirmed to be more powerful for sketch-based shape retrieval than the alternative techniques.
8.3 Further Work

A number of significant issues related to the SBIR has been addressed in this work. However, there are still a number of possible improvements that require further investigation. There are also a number of new directions in which the presented work can be employed.

Possible improvements and further studies on the proposed methods are addressed below.

- In the AP method (Chapter 4), the number of slices could be assigned dynamically. This means, based on the existing density of the edge pixel distribution, an appropriate number of slices (associated with \( \varphi \)) can be chosen. This would improve the retrieval performance but might slow down the feature extraction process. Moreover, variation in feature vector dimension makes the image matching complex. One solution to this problem can be partitioning the database images according to the respective feature vector dimension. In the matching process, the features extracted from the query image need to be compared only with the similar dimension features of the database images.

- The dynamic selection of partitions, which is described above, can be applied to the ARP method (Chapter 5) as well. Here it indicates dynamically assigning an appropriate sector size for different images based on the existing density of edge pixels (or strong edge pixels). The same problem of complex matching will arise, which could be solved as described.

- The parameter \( \beta \) (Chapter 4) can also be computed to be different for different images. This would improve the retrieval power of the AP and the ARP methods (Chapter 4 and 5, respectively). For this, we need to preprocess all images (database and query) and apply for example a proper fuzzy rule, to determine the parameter. This technique can improve the retrieval process outcome without making the search procedure more complex. On the other hand, applying such an algorithm to determine \( \beta \) dynamically would significantly slow down the off-line feature extraction speed.
The feature extraction technique proposed in Chapter 6 (i.e. CPCD) could be improved by considering more geometric properties derived from the shape itself or from the line segments set. For example, normalized area, eccentricity, or the angle between non-adjacent segments could be extracted, quantized and used for similarity measure between shapes. This, indeed, requires more computational power.

To be able to search for partial matches, a refined version of the AP or ARP techniques, extracted from the query image could be slid over the database images. The existing local correlations between corresponding partitions in the query and the database images can be used for the retrieval task. There are some important issues in this case including: (a) sliding the partitioning scheme anywhere on the image prohibits having an index for the search. This is due to the fact that the correlation detection necessitates sweeping the whole area of all database images at the search time. This profoundly slows down the search since it needs to process each image in the database at the query time, and (b) scale and rotation invariance properties are sacrificed as long as the correlation is carried out in the pixel domain, and (c) as an alternative solution, choosing some predefined regions in the image e.g. $4 \times 4$ sub-images, and extracting different features (indexes) for each sub-image, will remedy the speed problem. Nevertheless, the query image may not reside fully in a sub-image, for example it may be in an area comprising adjacent sub-images.

A reasonable solution for the partial match improvement is to primarily apply a segmentation algorithm to extract main objects in the image. Then, each object needs to be appropriately used for feature extraction. The extracted features (i.e. shape index) will be assigned to the whole image and appropriately stored in a feature set. This implies the existence of several indexes to one image which should be managed in the indexing subsystem of the image retrieval system. In the query time, the features extracted from the query image (usually a single object) are compared with those in the feature set to find the best matches. Finally, all images containing the query image are retrieved and presented to the user. The approach we proposed in Chapter 6 for sketch-based shape retrieval can be employed for the feature extraction process of this im-
The following is a number of new directions that arise from the topic discussed in this thesis.

- An important application of the research conducted here is artistic drawing skill training. It refers to an environment for evaluation of artistic drawings. The features extracted by the AP, ARP, or CPCD methods can be used to evaluate the painting created by a novice painter. This is accomplished by comparing the user draft with the original one and give an appropriate feedback (or score) to the user. This approach can be developed for using in art education departments in courses such as “visual investigation”.

- The methods presented here can be developed for sketch-based animation detection. That is, instead of a sketch image as the query, an animated sketch may be used for search in a video for similar clips. This could be considered as a new and interesting tool in video-on-demand research area.

- Since most current digital images are in the compressed forms, including JPEG and JPEG2000, it should be computationally more effective to directly perform sketch-based image retrieval in the compressed domain than in the pixel domain after decomposition. Our proposed approaches can be developed in such a way that features are extracted from a compressed format first and then used for matching. Obviously, success in this area is highly dependant on the internal structure of the compressed format employed for image representation.

- Design and implementation of a proper sketch-based user interface which is integrated with an appropriate indexing structure for organizing the features extracted by the proposed methods can generate a complete sketch-based retrieval system. Such a system needs to include several approaches of low-level content-based features and high-level concepts to achieve a satisfactory outcome. This is due to the fact that shortcomings of low-level features are compensated using high-level features and vice versa.
High-level features can be, for example, a kind of meta-data associated with the query and with the database images. For instance, if we apply a sketched image as the query and say “Leonardo Da Vinci” as the creator, then the system can find more effectively and more efficiently similar images to the given query. This will be done by searching only a branch of the image database which consists of all images created by Leonardo Da Vinci.

Making an end-to-end system requires some more consideration including design and implementation of appropriate graphical user interfaces (GUI). Moreover, several other algorithms need to be included for completeness of the usage, which in fact, by and large, is a commercial task. The concepts and algorithms proposed in this thesis can be used in such a system.
Bibliography


